Online Asymmetric Active Learning with Imbalanced Data

Xiaoxuan Zhang, Tianbao Yang, Padmini Srinivasan
Department of Computer Science
The University of Iowa, IA 52242
{xiaoxuan-zhang, tianbao-yang, padmini-srinivasan}@uiowa.edu

ABSTRACT

This paper considers online learning with imbalanced streaming data under a query budget, where the act of querying for labels is constrained to a budget limit. We study different active querying strategies for classification. In particular, we propose an asymmetric active querying strategy that assigns different probabilities for query to examples predicted as positive and negative. To corroborate the proposed asymmetric query model, we provide a theoretical analysis on a weighted mistake bound. We conduct extensive evaluations of the proposed asymmetric active querying strategy in comparison with several baseline querying strategies and with previous online learning algorithms for imbalanced data. In particular, we perform two types of evaluations according to which examples appear as "positive"/ "negative". In push evaluation only the positive predictions given to the user are taken into account; in push and query evaluation the decision to query is also considered for evaluation. The push and query evaluation strategy is particularly suited for a recommendation setting because the items selected for querying for labels may go to the end-user to enable customization and personalization. These would not be shown any differently to the end-user compared to recommended content (i.e., the examples predicated as positive). Additionally, given our interest in imbalanced data we measure F-score instead of accuracy that is traditionally considered by online classification algorithms. We also compare the querying strategies on five classification tasks from different domains, and show that the probabilistic query strategy achieves higher F-scores on both types of evaluation than deterministic strategy, especially when the budget is small, and the asymmetric query model further improves performance. When compared to the state-of-the-art cost-sensitive online learning algorithm under a budget, our online classification algorithm with asymmetric querying achieves a higher F-score on four of the five tasks, especially on the push evaluation.

KDD '16, August 13-17, 2016, San Francisco, CA, USA

DOI: http://dx.doi.org/10.1145/2939672.2939854

Keywords

Online Learning; Query Budget; Imbalanced Data; F-score

1. INTRODUCTION

Traditional batch classification algorithms that have been broadly applied in various data mining domains, such as document filtering [29], news classification [23], spam detection [2], and opinion mining [25], are challenged by applications characterized by large-scale, streaming data as for instance in web mining applications. Large amounts of data are continually generated by the web such as through social media and news. These datasets are often also highly imbalanced with respect to classes of interest. A key consequence of scale is that getting adequate training data becomes non trivial in effort and costly. To address these problems, we study online classification algorithms where the act of querying for labels is constrained to a budget limit.

Lowering cost is always a goal in algorithms processing large-scale and real-time data. In classification, there are two major sources of cost. The first is the obvious cost incurred due to errors in performance (false positive and false negative decisions). The second is the cost of labeling the data used to initially build or retrain the model over time. Focusing on the second cost for the moment, labelled data may be collected in two general ways with cost differences that are both subtle and explicit. We may directly ask the client (end-user) to provide labels. This approach is particularly useful for personalized recommendation [8, 1]. As a consequence, in addition to giving the client the high confidence positive predictions made by the system we also show instances of low confidence to label. While risky these low confidence instances are likely to be the most useful for improving the performance of the classifier. However, over time the client may become disappointed if the system takes big risks with too many false positives shown. Therefore, it makes sense to limit the amount of queries to label sent to the end-user. In this paper, we tackle this issue in an online setting for streaming data. We aim to maximize the performance subject to a budget limit for querying the labels. There are several fold of entangled difficulties: (i) how to decide which examples to query for the true labels given a budget limit; (ii) which performance measure should we target? (iii) how to evaluate the performance of the system? These difficulties become severe in the presence of imbalanced data. For example, if there are many more negative examples than positive examples, treating them equally for making the query decisions can be sub-optimal. A 'bad' query may even harm performance. For example, Figure 2

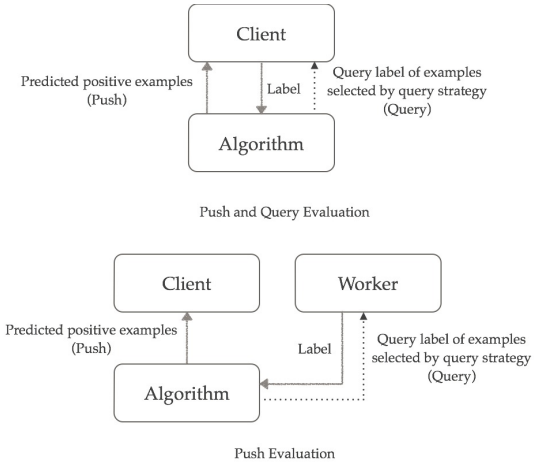


Figure 1: Illustration of two evaluation methods.

shows the results of two algorithms (the red and blue lines) on the OHSUMED dataset as discussed in the experiment section. The blue line queries the first 3,000 instances for labels then it stops querying. In contrast the red line algorithm keeps querying for labels of the examples that are predicted as positive. It shows that although the red line algorithm queries more labels it performs worse. The reason will be discussed in the experiment section. Moreover, the traditional performance measure by error rate is not appropriate for imbalanced data. Furthermore, the examples for label querying may also be pushed to the end-user in the same way as the recommended items and thus should be also taken into account in evaluation.

The contributions of the paper are three-fold. Firstly, we propose an asymmetric active querying strategy, which is randomized in nature and assigns different probabilities for querying to examples that are predicted as positive and negative. We also provide a rigorous theoretical analysis of the proposed asymmetric query strategy comparing with previous symmetric query strategy. Secondly, we evaluate the performance of different algorithms by two methods. In the first performance is assessed both in terms of instances predicted as positives (pushed to client) and instances shown for labeling (querying the client). I.e., both instances that are positive predictions by the classifier and instances selected for label querying are shown to the user as positive predictions. This evaluation is particularly suited when the client is the source of the labeling. In the second evaluation, performance is only based on the positive predictions made by the classifier. This traditional evaluation method is more suitable when the queried labels are from another source, e.g., internal workers (or crowd-workers). We refer to the first evaluation as 'push and query' and the second as 'push' evaluation. These two evaluations are in essence based upon what the user 'sees', and are illustrated in Figure 1. In our experiments, we find that the proposed algorithm performs well on both evaluations. Lastly, we evaluate the performance by F-1 score, which is more suited for imbalanced data. Most existing studies of online classification or learning mainly minimize error rate (or maximize accuracy). Lower error rate is always good. However, we are interested in working with highly imbalanced data where error rate is not meaningful. If 90% of the examples are negative, simply classifying all of the dataset as negative will give an error

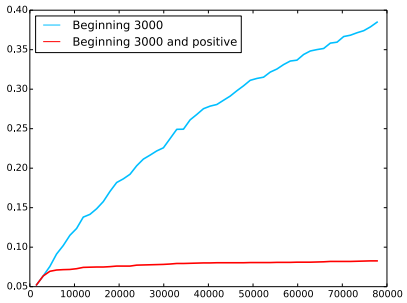


Figure 2: Bad queries harm the performance

rate of 0.1, which looks excellent but is actually meaningless. Therefore, we choose to use the F score as the performance metrics. In the paper, we will also show that the proposed asymmetric querying method also favors the F score.

The remainder of the paper is organized as follows. We first discuss the related work of online learning and online active classification in Section 2. Then we describe and analyze the algorithms in Section 3, including four algorithms with distinct query strategies and the variant with asymmetric query on positive and negative examples. The experimental design and results are discussed in Section 4 and the last section will be the discussion and conclusions.

2. RELATED WORK

Many online algorithms have been developed, e.g., the Perceptron algorithm [27], the exponentially weighted average algorithm [3, 22] and the online gradient descent [35, 16]. In the last ten years, we observe substantial applications of these algorithms in machine learning and data analytics, e.g., online classification [21, 15, 10, 19, 9]. Traditional online classification aims to minimize the number of mistakes by minimizing a cumulative convex loss function. However, most of the algorithms have innocently ignored both the imbalanced data distribution and the query budget constraint.

Learning with imbalanced data has attracted much attention from the machine learning and data mining community for many years. Most studies cast the problem into cost-sensitive learning that assigns different costs to mistakes of different classes [32, 14, 24, 28]. Batch learning with cost-sensitivity has been studied a lot, while few studies are devoted to online learning with imbalanced data [9, 31, 34]. These few studies either modify conventional loss functions to incorporate the given/defined cost matrix or use a different loss function to optimize a measure that is suited for imbalanced data. However, they also assume full knowledge of the label information for all received examples. This renders them unattractive for mining massive streaming data where querying for the labels is subject to a cost.

Online learning under a query budget has received little attention. Several papers have studied a similar problem in a context also known as label-efficient online learning or online active learning. Cesa-Bianchi, Lugosi, and Stoltz [6] studied the problem of learning from expert advice under a query budget. They proposed a simple strategy that uses an independent and identically distributed (i.i.d.) sequence $Z_1, \ldots, Z_T$ of Bernoulli random variables such that $\Pr(Z_t = 1) = \varepsilon$ and asks the label to be revealed whenever

$Z_t = 1$. The limitation with the pure random query strategy is that it does not differentiate between examples with high confidence score and low confidence score for classification. Later on, the same authors [7] designed a new strategy of query that makes the probability of querying dependent on the absolute value of the prediction score. This query algorithm has also been used in a recent work for cost-sensitive online learning [33]. Active learning for querying the labels has also been considered in different works for different algorithms from the perspective of sample complexity [11, 12, 4]. [11] analyzed the perceptron-like algorithm under an active setting and provided a complexity on the number of queried labels for achieving a certain generalization performance. [4, 12] studied online ridge-regression type of algorithms for classification for different querying strategies and established the sample complexity bound. However, these works have not consider the asymmetry between positive examples and negative examples for imbalanced data.

One of our goals in this paper is to compare different query strategies for learning with imbalanced data. Moreover, we note that the symmetric query model where the probability of querying is independent of the positive/negative decision is not well suited for imbalanced data. To understand this, consider that the number of positive examples is much smaller than the number of negative examples. As a result, there would be more false positives than false negatives. If we assign equal probabilities to these false predicted examples for querying the label, the algorithm would favor the negative class more than the positive class, consequentially harm the performance. Therefore, we propose a novel asymmetric query model that is demonstrated to be sound in theory and effective in practice as well. Moreover, the comparison of two different strategies to handle imbalanced data under a query budget, namely (i) an asymmetric query model plus a symmetric updating rule of the proposed algorithm, and (ii) an asymmetric updating rule plus a symmetric query model of a state-of-the-art algorithm [33], also demonstrate the proposed algorithm is very useful.

3. ONLINE ACTIVE LEARNING UNDER A QUERY BUDGET

We first introduce some notations. We denote by $\mathbf{x}_t \in \mathbb{R}^d$ the feature vector of the example received at the t-th round, and by $y_t \in \{1, -1\}$ the label of $\mathbf{x}_t$. Let $f(\mathbf{x})$ denote a prediction function. In the sequel, we will focus on the presentation using the linear model $f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x}$, while one can easily generalize it to a non-linear function in a reproducing kernel Hilbert space. Let $B > 0$ denote a budget limit on the number of queries.

The framework of online classification under a query budget is presented in Algorithm 1. We let $\mathbf{w}_t$ and B_t denote the model available before the $(t+1)$-th round and the budget used before the $(t+1)$-th round. Initially, the model is $\mathbf{w}_0 = 0$ and $B_0 = 0$. In line 5, the algorithm computes $p_t = \mathbf{w}_{t-1}^\top \mathbf{x}_t$ and in line 6 it makes a decision about the binary label of $\mathbf{x}_t$ by a function $\hat{y}_t = Predict(p_t)$ based on p_t. The simplest $Predict$ function is $\hat{y}_t = sign(p_t)$, i.e., using the sign of p_t to determine the label. More generally, we can use a threshold γ and and let $\hat{y}_t = sign(p_t - \gamma)$. For imbalanced data, we observe that using a threshold always yields better performance. We discuss how to set the value of γ in the experiment section.

After the binary decision is made, the algorithm enters the query stage, where it decides whether to query the label and update the model. When the algorithm reaches the budget limit, the model will not be updated because query is not allowed and there are two options for making future predictions. The Option I is to use the last updated model and the Option II is to use the averaged model before the iteration T_B when the budget is used up. Many previous studies have found that the averaged model might give more robust predictions than the last model [30].

If the remaining budget is not zero, i.e., $B_{t-1} < B$, we use the output of a query function $Z_t = Query(p_t)$, which might depend on p_t and some other parameters, to determine query ($Z_t = 1$) or not ($Z_t = 0$). If Z_t is 1, then the algorithm queries the label of current example denoted by $y_t \in \{1, -1\}$ from a source. Then the algorithm proceeds to update the model using $\mathbf{w}_t = Upate(\mathbf{w}_{t-1}, y_t, \mathbf{x}_t)$ in line 13. Function $Upate$ depends on what optimization method is employed and what surrogate loss function is assumed. Indeed, many different updating schemes can be used, including the margin-based updating rules (e.g., Perceptron [27], online passive-aggressive algorithm [9], confidence-weighted learning algorithm [13], etc) and online gradient descent updates for different types of loss functions [35]. Take the Perceptron for example, $\mathbf{w}_t$ is updated by

$$\mathbf{w}_t = \mathbf{w}_{t-1} + y_t \mathbf{x}_t \mathbb{I}(y_t f_t \leq 0) \quad (1)$$

where $\mathbb{I}(v)$ is an identity function that outputs 1 if v is true and otherwise outputs zero.

The query model is the key concern in this paper. Below we first discuss some baseline models that either arise straightforwardly or appear in previous work. Then we present the proposed query model for imbalanced data.

3.1 Baseline Query Models

We discuss several baseline query models below and comment on their deficiencies.

- **F**irst Come First Query (referred to as $\mathbf{Q_F}$). This strategy is simply to query the true labels of the first B examples and then uses the model learned from the first B examples to make predictions for all the following examples. This strategy is similar to the ϵ-greedy strategy used in bandit learning [20], where the first stage with B examples is devoted to exploration and the second stage is exploitation.

- **R**andom Query (referred to as $\mathbf{Q_R}$). This strategy is to use a Bernoulli random variable $Z_t = Bern(\epsilon)$ that equals 1 with a probability ϵ to determine whether to query or not. The random strategy has been used in predict with expert advices under budget feedback setting [5].

- **D**eterministic prediction dependent Query (referred to as $\mathbf{Q_D}$). Different from the first two query models where the decision to query does not depend on p_t, we can make the query function dependent on p_t. The motivation is that if p_t is closer to zero, meaning that the prediction is more uncertain, then we should query its label for updating the model. This strategy is very similar to active learning where uncertain examples should be queried first [18]. The deterministic predic-

Algorithm 1 Online Binary Classification Under A Query Budget

Input: budget B
Initialize $\mathbf{w}_0 = 0$, $B_0 = 0$
for $t = 1, \ldots, T$ **do**
  Receive an example $\mathbf{x}_t$
  Compute $p_t = \mathbf{w}_{t-1}^\top \mathbf{x}_t$
  Make a push decision $\hat{y}_t = Predict(p_t) \in \{1, -1\}$
  if $B_{t-1} \geq B$ **then**
    Set $\mathbf{w}_t$ as
      Option I: $\mathbf{w}_t = \mathbf{w}_{t-1}$
      Option II: $\mathbf{w}_t = \frac{1}{T_B} \sum_{t=0}^{T_B - 1} \mathbf{w}_t$
    where T_B is the smallest number that the budget is used up, i.e., $B_{T_B - 1} = B$.
  else
    Compute a variable $Z_t = Query(p_t) \in \{1, 0\}$
    if $Z_t = 1$ **then**
      Query the true label y_t from a source $\mathbf{S}$
      Update the model
        $\mathbf{w}_t = Upate(\mathbf{w}_{t-1}, y_t, \mathbf{x}_t)$
      Update $B_t = B_{t-1} + 1$
    end if
  end if
end for

tion dependent query function is given by

$$Query(p_t) = \mathbb{I}(|p_t| \leq c) \tag{2}$$

where $c > 0$ is a parameter that determines the threshold of uncertainty.

- Randomized **S**ymmetric prediction dependent Query (referred to as $\mathbf{Q_S}$). This strategy was proposed as selective sampling in previous work, where the output Z_t of the query function is a Bernoulli random variable with the sampling probability dependent on the prediction, i.e,,

$$\Pr(Z_t = 1) = \frac{c}{|p_t| + c} \tag{3}$$

It can be seen that the smaller p_t, the more uncertain the prediction and the higher probability to query for the label. In contrast to the asymmetric query model presented below, the above query model is symmetric for $p_t > 0$ and $p_t < 0$.

Comparing the above query models, we can see that Q_F and Q_R are prediction independent, and therefore will waste many budget on those easy examples with the model intact. Moreover, if the data is imbalanced and the first B examples are negative, then the model learned by using Q_F will predict all the following examples to be negative. Similarly, the Q_R model will also query more negative examples. In contrast, Q_D and Q_S are prediction dependent and therefore will likely query more uncertain examples facilitating the learning of the model $\mathbf{w}_t$. Q_S uses randomization in query that tends to be more robust. More importantly, it has been analyzed theoretically in [5] about the mistake bound. To facilitate the comparison between the symmetric query model and the proposed asymmetric query model in subsection 3.2, we present the mistake bound of Algorithm 1 using the symmetric query model Q_S in the following theorem.

THEOREM 1. *Let T_B be the smallest number that $B_{T_B - 1} = B$. If we run Algorithm 1 using Eqn. (3) as the query model, then for all $\mathbf{u} \in \mathbb{R}^d$ and for all $\xi > 0$, the expected number of mistakes up to T_B satisfies*

$$\mathrm{E}\left[\sum_{t=1}^{T_B} \mathbb{I}_{sign(p_t) \neq y_t}\right] \leq \frac{\alpha}{c} \sum_{t=1}^{T_B} \ell_\xi(y_t \mathbf{u}^\top \mathbf{x}_t) + \frac{\alpha^2}{2c} \|\mathbf{u}\|_2^2$$

where $\ell_\xi(z) = \max(0, \xi - z)$ is a hinge loss parameterized by a margin parameter $\xi > 0$, and $\alpha = \frac{c + R^2/2}{\xi}$ with R being the upper bound of data norm, i.e., $\max_t \|\mathbf{x}_t\|_2 \leq R$.

Remark: Since the upper bound holds for any $\mathbf{u}$, we can minimize the upper bound by choosing the best $\mathbf{u}$. Note that we only establish the number of mistake bound up to T_B since the model will keep the same after that and its performance is determined by examples received before iteration T_B. The proof can be found in [5]. For completeness, we include a proof in the appendix.

3.2 Asymmetric Query Model

The issue of the randomized symmetric query model is to treat the positive examples and the negative examples equally. For imbalanced data, this will be vulnerable to the majority class (e.g., the negative class). If the negative class is the majority class, positive examples will be more likely to be predicted as negative. Therefore, intuitively for the sake of learning the model, it is better to query more false negative examples than false positive examples, i.e., making the query asymmetric. To quantify this, we propose the following asymmetric query model, referred to as $\mathbf{Q}_A$:

$$\Pr(Z_t = 1) = \begin{cases} \frac{c_+}{|p_t| + c_+} & \text{if } p_t \geq 0 \\ \frac{c_-}{|p_t| + c_-} & \text{otherwise} \end{cases} \tag{4}$$

We establish below the weighted mistake bound of Algorithm 1 using the asymmetric query model. The proof is deferred to the supplement due to the limits of space.

THEOREM 2. *Let T_B be the smallest number that $B_{T_B - 1} = B$. If we run Algorithm 1 using Eqn. (4) as the query model, then for all $\mathbf{u} \in \mathbb{R}^d$ and for all $\xi_+, \xi_- > 0$, the expected weighted number of mistakes up to T_B satisfies*

$$\mathrm{E}\left[\sum_{y_t=1} c_- \mathbb{I}_{sign(p_t) \neq y_t} + \sum_{y_t=-1} c_+ \mathbb{I}_{sign(p_t) \neq y_t}\right]$$
$$\leq \alpha \left[\sum_{y_t=1} \ell_{\xi_-}(y_t \mathbf{u}^\top \mathbf{x}_t) + \sum_{y_t=-1} \ell_{\xi_+}(y_t \mathbf{u}^\top \mathbf{x}_t)\right] + \frac{\alpha^2}{2} \|\mathbf{u}\|_2^2$$

where $\alpha = \max\{\frac{c_+ + R^2/2}{\xi_+}, \frac{c_- + R^2/2}{\xi_-}\}$ with R being the upper bound of data norm, i.e., $\max_t \|\mathbf{x}_t\|_2 \leq R$.

Remark: Compared to Theorem 1, there are two key differences: (i) Theorem 2 is bounding the weighted number of mistakes, where the false negative is weighted by c_- and false positive is weighted by c_+; (ii) the mistake bound in Theorem 2 is compared to the optimal loss that is defined using different margin ξ_+ and ξ_- for positive and negative examples, respectively. It is these differences that render the flexibility of Algorithm 1 in balancing between false negative

and false positive for imbalanced data. Hence, it achieves the similar affect as using different costs for false negative and false positive, which has been widely adopted in previous studies on learning from imbalanced data. In particular, if the negative class is the dominant class, then it is expected that c_- should be set to a larger value than c_+. This phenomenon has been observed in our experiments, which validates the result in Theorem 2.

PROOF. We denote by $\hat{y}_t = sign(p_t)$ and introduce the Bernoulli random variable $M_t = \mathbb{I}_{\hat{y}_t \neq y_t}$. Consider now a round t where the algorithms queries a label and makes a mistake, i.e., $M_t Z_t = 1$. We consider two scenarios. First if $p_t \geq 0$, then we have for any $\mathbf{u} \in \mathbb{R}^d$ and $\xi_- > 0$,

$$\begin{aligned}
&\xi_+ - \ell_{\xi_+}(y_t \mathbf{u}^\top \mathbf{x}_t) = \xi_+ - \max(0, \xi_+ - y_t \mathbf{u}^\top \mathbf{x}_t) \\
&\leq y_t \mathbf{u}^\top \mathbf{x}_t = y_t (\mathbf{u} - \mathbf{w}_{t-1} + \mathbf{w}_{t+1})^\top \mathbf{x}_t \\
&= y_t \mathbf{w}_{t-1}^\top \mathbf{x}_t + \frac{1}{2}\|\mathbf{u} - \mathbf{w}_{t-1}\|_2^2 - \frac{1}{2}\|\mathbf{u} - \mathbf{w}_t\|_2^2 \\
&+ \frac{1}{2}\|\mathbf{w}_{t-1} - \mathbf{w}_t\|_2^2
\end{aligned}$$

where we use the fact $\mathbf{w}_t = \mathbf{w}_{t-1} + y_t \mathbf{x}_t$ for $M_t Z_t = 1$. Since $y_t \neq \hat{y}_t$, then $y_t \mathbf{w}_{t-1}^\top \mathbf{x}_t \leq 0$. Replacing $\mathbf{u}$ by $\alpha \mathbf{u}$ with $\alpha > 0$ and reorganize the inequality we have

$$\begin{aligned}
(\alpha \xi_+ + |p_t|) M_t Z_t \leq \alpha \ell_{\xi_+}(y_t \mathbf{u}^\top \mathbf{x}_t) + \frac{1}{2}\|\alpha \mathbf{u} - \mathbf{w}_{t-1}\|_2^2 \\
- \frac{1}{2}\|\alpha \mathbf{u} - \mathbf{w}_t\|_2^2 + \frac{1}{2}\|\mathbf{w}_{t-1} - \mathbf{w}_t\|_2^2
\end{aligned}$$

We note that the above inequality holds for all rounds such that $\hat{y}_t = 1$. If $M_t Z_t = 0$, then $\mathbf{w}_t = \mathbf{w}_{t-1}$, and the above inequality holds because $\alpha \ell_{\xi_+}(y_t \mathbf{u}^\top \mathbf{x}_t) \geq 0$.

Similarly, if $p_t < 0$, then for any $\mathbf{u} \in \mathbb{R}^d$ and $\xi_- > 0$,

$$\begin{aligned}
(\alpha \xi_- + |p_t|) M_t Z_t \leq \alpha \ell_{\xi_-}(y_t \mathbf{u}^\top \mathbf{x}_t) + \frac{1}{2}\|\alpha \mathbf{u} - \mathbf{w}_{t-1}\|_2^2 \\
- \frac{1}{2}\|\alpha \mathbf{u} - \mathbf{w}_t\|_2^2 + \frac{1}{2}\|\mathbf{w}_{t-1} - \mathbf{w}_t\|_2^2
\end{aligned}$$

The above inequality holds for all rounds such that $\hat{y}_t = -1$. Summing the above inequality over $t = 1, \ldots, T_B$, we have

$$\begin{aligned}
&\sum_{\hat{y}_t=1} (\alpha \xi_+ + |p_t|) M_t Z_t + \sum_{\hat{y}_t=-1} (\alpha \xi_- + |p_t|) M_t Z_t \\
&\leq \alpha \left[\sum_{y_t=-1} \ell_{\xi_+}(y_t \mathbf{u}^\top \mathbf{x}_t) + \sum_{y_t=1} \ell_{\xi_-}(y_t \mathbf{u}^\top \mathbf{x}_t) \right] \\
&+ \frac{1}{2}\|\alpha \mathbf{u} - \mathbf{w}_0\|_2^2 + \frac{1}{2}\sum_{t=1}^{T_B} \|\mathbf{w}_{t-1} - \mathbf{w}_t\|_2^2 \\
&\leq \alpha \left[\sum_{y_t=-1} \ell_{\xi_+}(y_t \mathbf{u}^\top \mathbf{x}_t) + \sum_{y_t=1} \ell_{\xi_-}(y_t \mathbf{u}^\top \mathbf{x}_t) \right] \\
&+ \frac{1}{2}\|\alpha \mathbf{u} - \mathbf{w}_0\|_2^2 + \sum_{t=1}^{T_B} \frac{\|\mathbf{x}_t\|_2^2}{2} M_t Z_t
\end{aligned}$$

In the first inequality, on the right hand side, we use the summation over $y_t = 1$ and $y_t = -1$. The inequality holds because when $M_t Z_t = 1$, $\hat{y}_t = 1$ indicates $y_t = -1$, and $\hat{y}_t = -1$ indicates $y_t = 1$. Then,

$$\begin{aligned}
&\sum_{\hat{y}_t=1} (\alpha \xi_+ + |p_t| - \frac{R^2}{2}) M_t Z_t + \sum_{\hat{y}_t=-1} (\alpha \xi_- + |p_t| - \frac{R^2}{2}) M_t Z_t \\
&\leq \alpha \left[\sum_{y_t=-1} \ell_{\xi_+}(y_t \mathbf{u}^\top \mathbf{x}_t) + \sum_{y_t=1} \ell_{\xi_-}(y_t \mathbf{u}^\top \mathbf{x}_t) \right] + \frac{\alpha^2}{2}\|\mathbf{u}\|_2^2
\end{aligned}$$

Since $\alpha = \max\{\frac{c_+ + R^2/2}{\xi_+}, \frac{c_- + R^2/2}{\xi_-}\}$, therefore $\alpha \xi_+ \geq c_+ + R^2/2$ and $\alpha \xi_- \geq c_- + R^2/2$, then have

$$\begin{aligned}
&\sum_{\hat{y}_t=1} (c_+ + |p_t|) M_t Z_t + \sum_{\hat{y}_t=-1} (c_- + |p_t|) M_t Z_t \\
&\leq \alpha \left[\sum_{y_t=-1} \ell_{\xi_+}(y_t \mathbf{u}^\top \mathbf{x}_t) + \sum_{y_t=1} \ell_{\xi_-}(y_t \mathbf{u}^\top \mathbf{x}_t) \right] + \frac{\alpha^2}{2}\|\mathbf{u}\|_2^2
\end{aligned}$$

By taking expectation over randomness in Z_t and noting that $\mathrm{E}[Z_t] = \frac{c_+}{c_+ + |p_t|}$ for $\hat{y}_t = 1$ and $\mathrm{E}[Z_t] = \frac{c_-}{c_- + |p_t|}$ for $\hat{y}_t = -1$, we have

$$\begin{aligned}
&\mathrm{E}\left[\sum_{\hat{y}_t=1} c_+ \mathbb{I}_{\hat{y}_t \neq y_t} + \sum_{\hat{y}_t=-1} c_- \mathbb{I}_{\hat{y}_t \neq y_t} \right] \\
&\leq \alpha \left[\sum_{y_t=-1} \ell_{\xi_+}(y_t \mathbf{u}^\top \mathbf{x}_t) + \sum_{y_t=1} \ell_{\xi_-}(y_t \mathbf{u}^\top \mathbf{x}_t) \right] + \frac{\alpha^2}{2}\|\mathbf{u}\|_2^2
\end{aligned}$$

i.e.,

$$\begin{aligned}
&\mathrm{E}\left[\sum_{y_t=-1} c_+ \mathbb{I}_{\hat{y}_t \neq y_t} + \sum_{y_t=1} c_- \mathbb{I}_{\hat{y}_t \neq y_t} \right] \\
&\leq \alpha \left[\sum_{y_t=-1} \ell_{\xi_+}(y_t \mathbf{u}^\top \mathbf{x}_t) + \sum_{y_t=1} \ell_{\xi_-}(y_t \mathbf{u}^\top \mathbf{x}_t) \right] + \frac{\alpha^2}{2}\|\mathbf{u}\|_2^2
\end{aligned}$$

□

From the result in Theorem 2, we can see the proposed asymmetric query strategy aims to minimize the cost-sensitive error. Next, we leverage the previous results to show that minimizing the cost-sensitive error with appropriate costs is equivalent to the F-measure maximization. To present the results, we first give some notations. Let $h(\mathbf{x}) \in \mathcal{H} : \mathbb{R}^d \to \{1, -1\}$ denote a classifier and $\mathbf{e}(h) = (e_1(h), e_2(h))^\top$ denote the false negative (FN) error and false positive (FP) error of $h(\mathbf{x})$ on the population level, respectively, i.e.,

$$\begin{aligned}
e_1(h) &= \Pr(y = 1, h(\mathbf{x}) = -1) \\
e_2(h) &= \Pr(y = -1, h(\mathbf{x}) = 1)
\end{aligned}$$

where $\Pr(\cdot)$ denotes the probability over $(\mathbf{x}, y)$. When it is clear from the context, we write $\mathbf{e} = \mathbf{e}(h)$ for short. Let P_1 denote the marginal probability of the positive class, i.e., $P_1 = \Pr(y = 1)$. Then the F-measure (i.e., F-1 score) of $h(\cdot)$ on the population level can by computed by [26]

$$F(h) \triangleq F(\mathbf{e}) = \frac{2(P_1 - e_1)}{2P_1 - e_1 + e_2}$$

Let $\mathbf{c}(t) = (1 - \frac{\tau}{2}, \frac{\tau}{2})^\top$. The following proposition exhibits that maximizing the F-measure is equivalent to minimizing a cost-sensitive error.

PROPOSITION 1. *(Proposition 4 [26]) Let $F_* = \max_{\mathbf{e}} F(\mathbf{e})$. Then we have $\mathbf{e}_* = \arg\min_{\mathbf{e}} \mathbf{c}(F_*)^\top \mathbf{e} \Leftrightarrow F(\mathbf{e}_*) = F_*$.*

The above proposition indicates that one can optimize the following cost-sensitive error

$$\mathbf{c}(F_*)^\top \mathbf{e} = \left(1 - \frac{F_*}{2}\right) e_1 + \frac{F_*}{2} e_2 \tag{5}$$

to obtain an optimal classifier $h^*(\mathbf{x})$, which will give the optimal F-measure, i.e., $F(h^*) = F_*$. However, the cost-sensitive error in (5) requires knowing the exact value of the optimal F-measure. To address this issue, we discretize $(0, 1)$ to have a set of evenly distributed values $\{\theta_1, \ldots, \theta_K\}$ such that $\theta_{j+1} - \theta_j = \epsilon_0/2$. Then we can solve for a series of K classifiers to minimize the cost-senstive error

$$h_j^* = \arg\min_{h \in \mathcal{H}} \left(1 - \frac{\theta_j}{2}\right) e_1 + \frac{\theta_j}{2} e_2 = \mathbf{c}(\theta_j)^\top \mathbf{e}, j = 1, \ldots, K \tag{6}$$

The following proposition shows that there exists one classifier among $\{h_j^*, \cdots, h_K^*\}$ that can achieve a close-to-optimal F-measure as long as ϵ_0 is small enough.

PROPOSITION 2. *Let $\{\theta_1, \ldots, \theta_K\}$ be a set of values evenly distributed in $(0, 1)$ such that $\theta_{j+1} - \theta_j = \epsilon_0/2$. Then there exists $h_j^* \in \{h_j^*, \cdots, h_K^*\}$ such that*

$$F(h_j^*) \geq F_* - \frac{2\epsilon_0 B}{P_1}$$

where $B = \max_{\mathbf{e}} \|\mathbf{e}\|_2$.

Remark: The above proposition is a corollary of Proposition 5 in [26]. Note that the cost-sensitive error in (6) is just the population level counterpart of the cost-sensitive error in Theorem 2, which further justifies the proposed asymmetric query model.

The above analysis implies that we can try different values for the costs associated with the false negative error and false positive error, and use the cross-validation approach to choose the best setting.

4. EXPERIMENT AND RESULTS

In order to investigate the algorithms on data of different types, dimensionality, and proportion of positive examples, we conduct the experiment on 5 binary classification tasks from 3 datasets. All datasets are split as 2:1 for validation and testing respectively, with more information listed in Table 1. The validation data is used to tune the parameters in the compared algorithms. The testing data is used to evaluate the performance of different algorithms. On each collection, we evaluate the performance mainly through F_1 score across the number of received examples. Also, we provide both "query evaluation" and "query and push evaluation" according to the potential two types of label sources.

4.1 Data

One collection is cover type dataset from the UCI repository of Machine Learning databases. It contains 581,012 examples and 7 classes of forest type, namely, Spruce-Fir, Lodgepole Pine, Ponderosa Pine, Cottonwood/Willow, Aspen, Douglas-fir, and Krummholz. Since the 7 classes have distinct rates of positive examples, we conduct the experiment on three of them with different level of imbalance to investigate the performance of algorithms. The three binary classification problems are Ponderosa Pine vs Non Ponderosa Pine, referred to as "cov1", Spruce-Fir vs Non Spruce-Fir, referred to as "cov2", and Lodgepole Pine vs None Lodgepole Pine, referred to as "cov3". From cov1, cov2 to cov3, the level of imbalance decreases. Detailed information about the three tasks can be seen in Table 1.

Table 1: Statistics of five classification tasks

	# of examples		Percentage	# of
	Validation	Testing	of Positive	features
cov1			6.15	
cov2	387,341	19,3671	36.46	54
cov3			48.76	
hd	155,630	77815	6.25	313,539
2days	666,667	333,333	1.25	450,065

Table 2: The best parameter values of different algorithms using PE

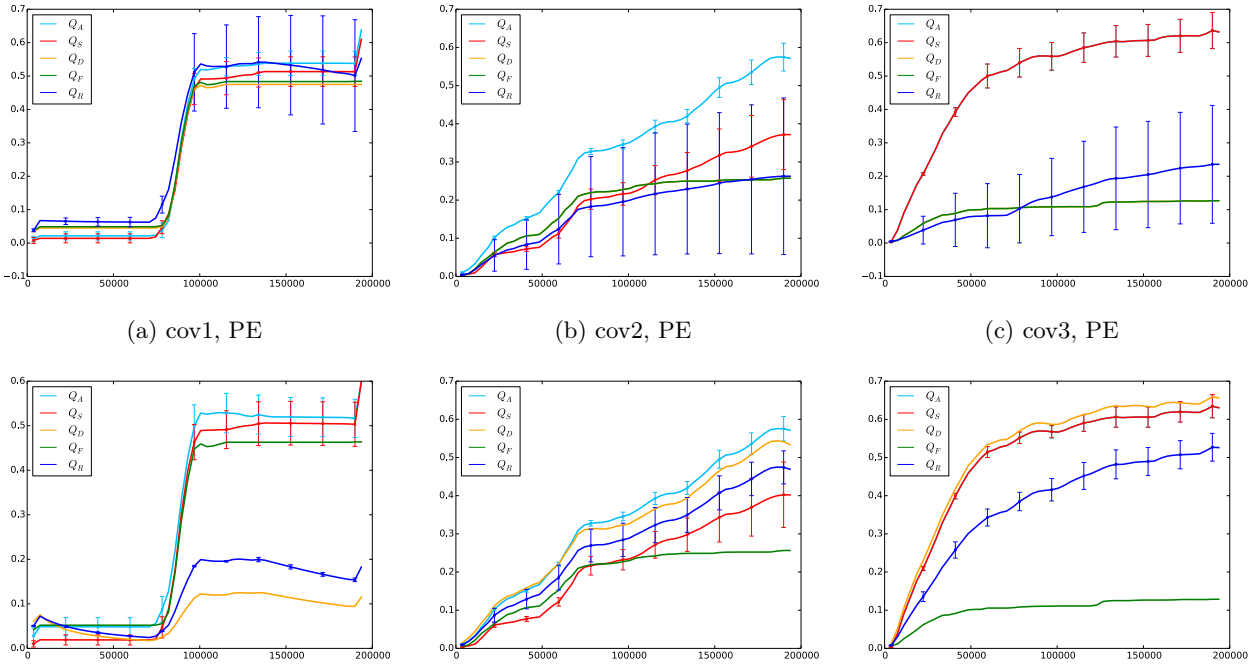
		cov1	cov2	cov3	hd	2days
$\mathbf{Q_A}$	γ	0.1	0.1	0	10	0.1
	c_+	0.005	0.01	0.1	1	0.01
	c_-	0.01	0.1	0.1	10	1
$\mathbf{Q_S}$	γ	0.1	0	0	10	1
	c	0.01	0.1	0.1	10	0.1
$\mathbf{Q_D}$	γ	1	0	1	10	0
	c	0.01	0.01	0.01	100	0.01
$\mathbf{Q_F}$	γ	0	0	1	1	0.1
$\mathbf{Q_R}$	γ	0	0.1	0.1	0	0
CSOAL	δ	0.01	0.01	0.1	0.1	0.1

We also evaluate our algorithm on two more datasets from different domains, OHSUMED - a dataset of biomedical publications, and 2 days' tweets collected from Twitter. OHSUMED [17] is a well-known dataset, collecting 348, 543 medical documents from MEDLINE from the year 1987 to 1991. Each document consist of all or some of following fields: MEDLINE identifier, MeSH terms, title, publication type, abstract, author, and source. Each document is associated with one or more MeSH terms, the medical subject heading assigned by human. Since the MeSH term is organized in a tree structure, we pick a subtree rooted at the MeSH term "Heart Disease" to be the positive class, and leave all the terms not in the subtree as negative. Specifically, a document is a positive example if and only if it contains at least one MeSH term in the "Heart Disease" subtree. The task is referred to as hd.

The two days' tweets, referred as 2days in the following discussion, is a collection related to life satisfaction of the author of the tweets. It is collected by keywords such as "I", "my", etc. And each tweet is manually labeled as satisfy, dissatisfy, or irrelevant. We consider the binary classification problem of "related to the topic of life satisfaction" (positive example) or not relevant (negative example). Due to the nature of the data (e.g., tweet is a short text). this is a difficult task.

4.2 Evaluation

As we mentioned in the introduction section, we mainly evaluate the algorithms by F_1 score. Specifically, we plot the accumulative F_1 score along with the increase of the iteration. We also conduct two types of evaluation according to two types of label source. A push evaluation (PE) means that the labeling is independent from the use of the application, while push and query evaluation (PQE) means the labels are completely obtained from the user feedback. Specifically, in PE, only the examples such that $\hat{y}_t = 1$ are counted as positive predication, and in PQE the queried ex-



(a) cov1, PE (b) cov2, PE (c) cov3, PE

(d) cov1, PQE (e) cov2, PQE (f) cov3, PQE

Figure 3: Comparison of different query models on covtype dataset with budget $B = 1,250$.

amples are also considered as positive predication. Thus, PQE may lead to a lower precision and higher recall, and it is related to the budget and data as well. We compare the proposed online classification algorithm with the asymmetric query model $\mathbf{Q_A}$ to (i) different query strategies as discussed in subsection 3.1; and (ii) a state-of-the-art cost-sensitive online active learning algorithm under a query budget, referred to as **CSOAL** [33]. Different from our algorithm $\mathbf{Q_A}$, CSOAL uses an asymmetric updating rule instead of an asymmetric query model. In experiments for the proposed framework (with different querying strategies), we use the Perceptron update in (1) for updating the model. The comparison between $\mathbf{Q_A}$, CSOAL and $\mathbf{Q_S}$ can help demonstrate which method is more effective in the context of online learning with imbalanced data under a query budget. For all randomized algorithms, we repeat 10 times and show the average and the error bar in all figures. For the presented algorithms with different query strategies, we use the same option I in Algorithm 1.

4.3 Results

As mentioned above, each collection is split into a validation and a testing subset. To be fair, we tune the parameters on the validation set for all the algorithms. Due to limits of space, we only report the best parameter values using PE in Table 2. For the three tasks on the covtype data, we fix the budget to $B = 1250$ and mainly investigate how the performance changes as the ratio of the number of positive examples change. For hd and 2days, we try two different values for the budget, a lower budget and a higher budget.

We show the results on covtype data in Figure 3 that compares different query strategies and Figure 4 that compares $\mathbf{Q_A}$ to CSOAL. We only report the results with a lower budget value $B = 1,250$, and the results with a larger budget value $B = 5000$ are included in the supplementary material due to limits of space. The results on hd are shown in Figure 5 for two different values of budget. Figure 6 shows the results on 2days data for two different values of budget.

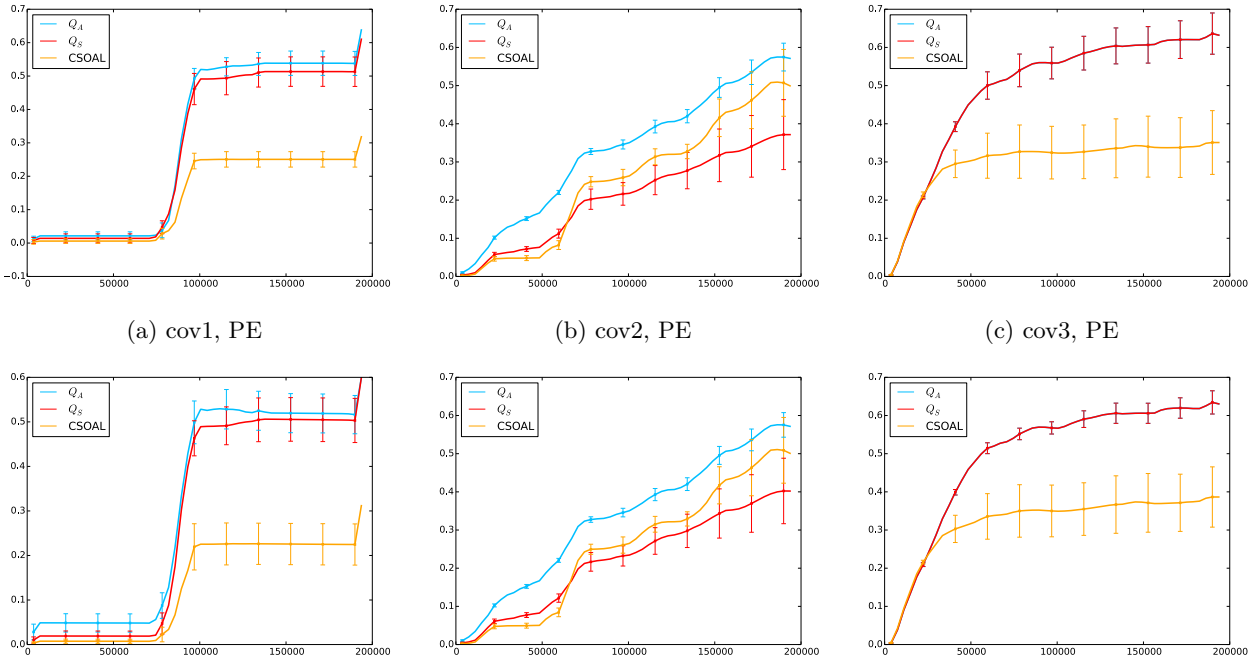
4.4 Discussion of the results

We first discuss the best parameter values of the proposed algorithm $\mathbf{Q_A}$. It can be seen from Table 2 the value of c_- is larger than c_+ on imbalanced data (cov1, cov2, hd, and 2days), and they are the same for the balanced data cov3. This is consistent with our theoretical findings.

The comparison between different query strategies on covtype data (Figure 3) clearly demonstrate the effectiveness of the proposed asymmetric query model. In particular, when the data is imbalanced (cov1, cov2), the asymmetric query model is better than other baseline query models. When the data is balanced, the asymmetric query model reduces to the symmetric query model. Therefore, the line of $\mathbf{Q_S}$ overlaps with the line of $\mathbf{Q_A}$ in Figure 3(e), (f) and the line of $\mathbf{Q_D}$ in Figure 3(e) as well. Moreover in this case, the probabilistic query model does not have any advantage over deterministic query strategy. On 2days data, we have similar observations. On hd data, the comparison between different query strategies (Figure 5 (a)$\sim$(d)) shows that using a smaller query budget favors the asymmetric query model more than using a larger value of budget. From the results, we also observe that using PE favors $\mathbf{Q_A}$ more than using PQE.

Next, we discuss the comparison to CSOAL. On cover type dataset, $\mathbf{Q_A}$ performs better than CSOAL on all the classes and both evaluations. On the hd data, the CSOAL performs better when $B = 5,000$ especially on PQE, however, it loses to $\mathbf{Q_A}$ when the budget is low ($B = 100$). On 2days data, $\mathbf{Q_A}$ performs consistently better than CSOAL.

Finally, it is worth mentioning that when only querying those examples predicted as positive the algorithm performs extremely bad as seen in Figure 2. We conjecture that the



(a) cov1, PE (b) cov2, PE (c) cov3, PE

(d) cov1, PQE (e) cov2, PQE (f) cov3, PQE

Figure 4: Comparison with CSOAL on covtype dataset with budget $B = 1,250$.

reason is that many of the positive predictions are difficult, i.e. their features look like positive examples but they are actually negative. There can be many such examples as the dataset is imbalanced. And querying those examples will push the decision boundary to that side that includes fewer positive examples and consequentially harm performance. Therefore, using a probabilistic query model with a smaller value of c_+ will reduce querying for such examples, which is consistent with observation both in theory and practice.

5. CONCLUSIONS

In this paper, we have considered online classification with imbalanced data under a query budget. We compare and investigate different query strategies in an online classification algorithm. We also propose a novel asymmetric query model and provide a theoretical analysis of the weighted mistake bound. We conducted extensive experiments on five classification tasks from three real datasets. The experimental results demonstrate the usefulness of the proposed online classification algorithm with an asymmetric query model.

6. ACKNOWLEDGMENTS

This work is supported by the National Science Foundation under grant 1545995. T. Yang was also partially supported by NSF (1463988). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the National Science Foundation.

7. REFERENCES

[1] X. Amatriain. Mining large streams of user data for personalized recommendations. *SIGKDD Explor. Newsl.*, 14:37–48, 2013.

[2] C. Castillo, D. Donato, A. Gionis, V. Murdock, and F. Silvestri. Know your neighbors: Web spam detection using the web topology. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 423–430. ACM, 2007.

[3] N. Cesa-Bianchi, Y. Freund, D. Haussler, D. P. Helmbold, R. E. Schapire, and M. K. Warmuth. How to use expert advice. *Journal of ACM*, 44:427–485, 1997.

[4] N. Cesa-Bianchi, C. Gentile, and F. Orabona. Robust bounds for classification via selective sampling. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 121–128, 2009.

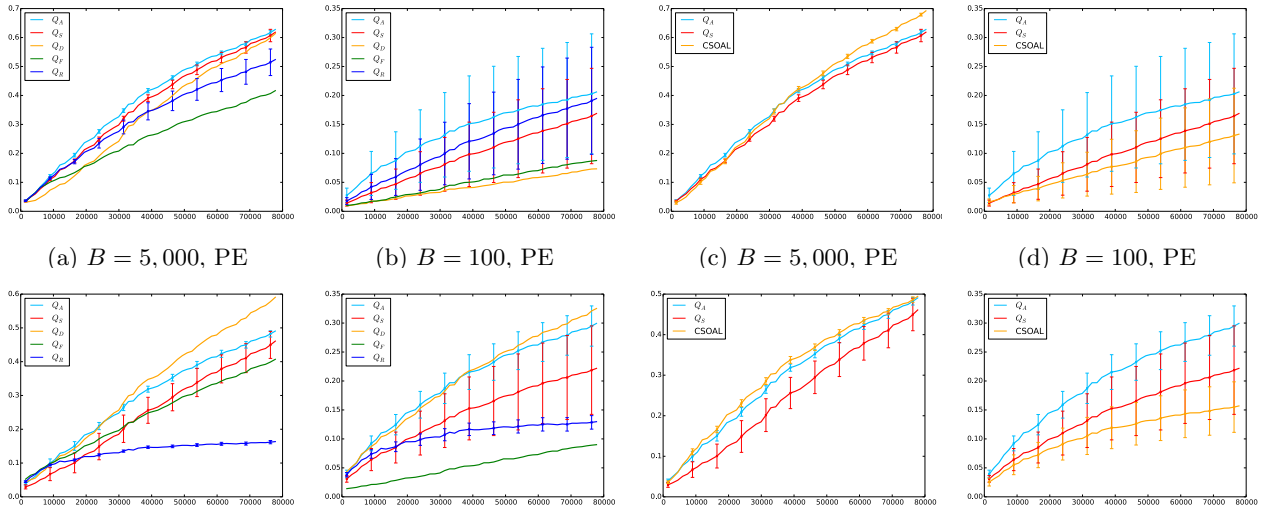
[5] N. Cesa-Bianchi and G. Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006.

[6] N. Cesa-Bianchi, G. Lugosi, and G. Stoltz. Minimizing regret with label efficient prediction. In *Proceedings of Conference of Learning Theory*, volume 3120 of *Lecture Notes in Computer Science*, pages 77–92. Springer, 2004.

[7] N. Cesa-Bianchi, G. Lugosi, and G. Stoltz. Minimizing regret with label efficient prediction. *IEEE Transactions on Information Theory*, 51(6):2152–2162, 2005.

[8] W. Chu and S.-T. Park. Personalized recommendation on dynamic content using predictive bilinear models. In *Proceedings of the 18th International Conference on World Wide Web*, pages 691–700, 2009.

[9] K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, and Y. Singer. Online passive-aggressive algorithms. *Journal of Machine Learning Research*, 7, 2006.



(a) $B = 5,000$, PE (b) $B = 100$, PE (c) $B = 5,000$, PE (d) $B = 100$, PE

(e) $B = 5,000$, PQE (f) $B = 100$, PQE (g) $B = 5,000$, PQE (h) $B = 100$, PQE

Figure 5: Results on the hd data with different budget

[10] K. Crammer and Y. Singer. Ultraconservative online algorithms for multiclass problems. *Journal of Machine Learning Research*, 3:951–991, 2003.

[11] S. Dasgupta, A. T. Kalai, and C. Monteleoni. Analysis of perceptron-based active learning. *J. Mach. Learn. Res.*, pages 281–299, 2009.

[12] O. Dekel, C. Gentile, and K. Sridharan. Selective sampling and active learning from single and multiple teachers. *J. Mach. Learn. Res.*, pages 2655–2697, 2012.

[13] M. Dredze, K. Crammer, and F. Pereira. Confidence-weighted linear classification. In *Proceedings of the 25th International Conference on Machine Learning*, pages 264–271, 2008.

[14] C. Elkan. The foundations of cost-sensitive learning. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence - Volume 2*, pages 973–978, 2001.

[15] C. Gentile. A new approximate maximal margin classification algorithm. *Journal of Machine Learning Research*, 2:213–242, Mar. 2002.

[16] E. Hazan, A. Agarwal, and S. Kale. Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69(2-3):169–192, 2007.

[17] W. Hersh, C. Buckley, T. Leone, and D. Hickam. Ohsumed: An interactive retrieval evaluation and new large test collection for research. In *SIGIR 94*, pages 192–201. Springer, 1994.

[18] S. Huang, R. Jin, and Z. Zhou. Active learning by querying informative and representative examples. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36(10):1936–1949, 2014.

[19] J. Kivinen, A. Smola, and R. Williamson. Online learning with kernels. *IEEE Transactions on Signal Processing*, 52(8), 2004.

[20] J. Langford and T. Zhang. The epoch-greedy algorithm for multi-armed bandits with side information. In *Advances in Neural Information Processing Systems (NIPS)*, pages 817–824, 2007.

[21] Y. Li and P. M. Long. The relaxed online maximum margin algorithm. *Mach. Learn.*, 46(1-3):361–387, 2002.

[22] N. Littlestone and M. K. Warmuth. The weighted majority algorithm. *Information and Computation*, pages 212–261, 1994.

[23] J. Liu, P. Dolan, and E. R. Pedersen. Personalized news recommendation based on click behavior. In *Proceedings of the 15th international conference on Intelligent user interfaces*, pages 31–40. ACM, 2010.

[24] H. Masnadi-Shirazi and N. Vasconcelos. Risk minimization, probability elicitation, and cost-sensitive svms. In *Proceedings of the 27th International Conference on Machine Learning (ICML)*, pages 759–766, 2010.

[25] A. Pak and P. Paroubek. Twitter as a corpus for sentiment analysis and opinion mining. In *LREC*, volume 10, pages 1320–1326, 2010.

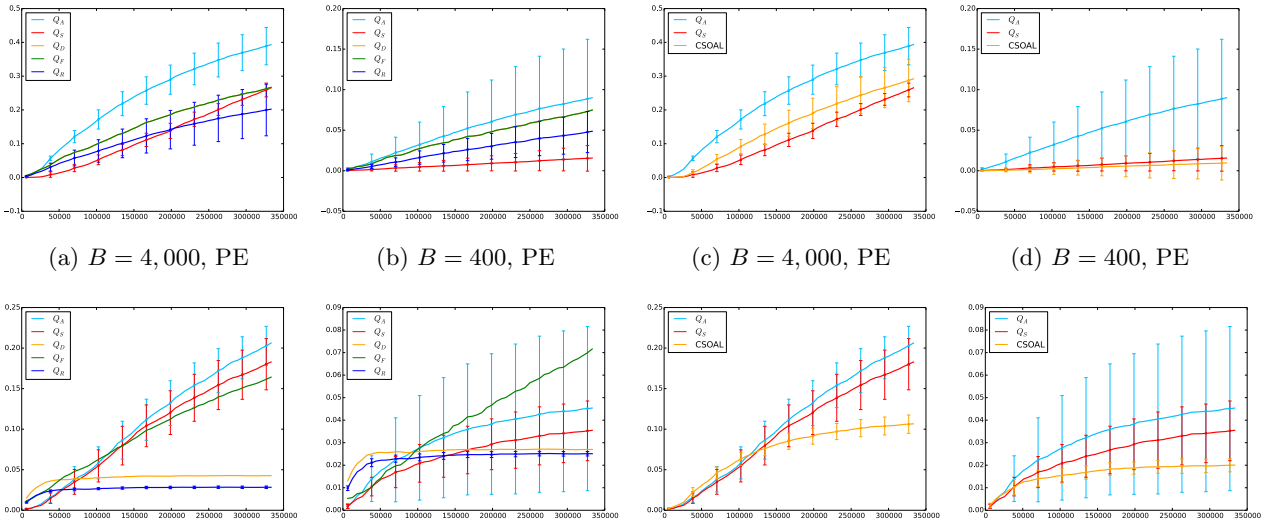
[26] S. Puthiya Parambath, N. Usunier, and Y. Grandvalet. Optimizing f-measures by cost-sensitive classification. In *Advances in Neural Information Processing Systems 27*, pages 2123–2131, 2014.

[27] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65, 1958.

[28] C. Scott. Surrogate losses and regret bounds for cost-sensitive classification with example-dependent costs. In *Proceedings of International Conference of Machine Learning (ICML)*, pages 153–160, 2011.

[29] F. Sebastiani. Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, 34(1):1–47, 2002.

[30] O. Shamir and T. Zhang. Stochastic gradient descent for non-smooth optimization: Convergence results and optimal averaging schemes. In *Proceedings of the 30th International Conference on Machine Learning, (ICML)*, pages 71–79, 2013.



(a) $B = 4,000$, PE (b) $B = 400$, PE (c) $B = 4,000$, PE (d) $B = 400$, PE

(e) $B = 4,000$, PQE (f) $B = 400$, PQE (g) $B = 4,000$, PQE (h) $B = 400$, PQE

Figure 6: Results on the 2days data with different budget

[31] J. Wang, P. Zhao, and S. C. H. Hoi. Cost-sensitive online classification. In *Proceedings of IEEE International Conference on Data Mining (ICDM)*, pages 1140–1145, 2012.

[32] G. M. Weiss, K. McCarthy, and B. Zabar. Cost-sensitive learning vs. sampling: Which is best for handling unbalanced classes with unequal error costs? In *Proceedings of the 2007 International Conference on Data Mining (DMIN)*, pages 35–41, 2007.

[33] P. Zhao and S. C. H. Hoi. Cost-sensitive online active learning with application to malicious URL detection. In *The 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 919–927, 2013.

[34] P. Zhao, S. C. H. Hoi, R. Jin, and T. Yang. Online auc maximization. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 233–240, 2011.

[35] M. Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 928–936, 2003.

APPENDIX

A. PROOF OF THEOREM 1

PROOF. We denote by $\hat{y}_t = sign(p_t)$ and introduce the Bernoulli random variable $M_t = \mathbb{I}_{\hat{y}_t \neq y_t}$. Consider a round t where the algorithms queries a label and makes a mistake. Then $Z_t = 1$ and $M_t = 1$. Then we have for any $\mathbf{u} \in \mathbb{R}^d$

$$\begin{aligned}
&\xi - \ell_\xi(y_t \mathbf{u}^\top \mathbf{x}_t) = \xi - \max(0, \xi - y_t \mathbf{u}^\top \mathbf{x}_t) \leq y_t \mathbf{u}^\top \mathbf{x}_t \\
&= y_t(\mathbf{u} - \mathbf{w}_{t-1} + \mathbf{w}_{t+1})^\top \mathbf{x}_t \\
&= y_t \mathbf{w}_{t-1}^\top \mathbf{x}_t + \frac{1}{2}\|\mathbf{u} - \mathbf{w}_{t-1}\|_2^2 - \frac{1}{2}\|\mathbf{u} - \mathbf{w}_t\|_2^2 \\
&+ \frac{1}{2}\|\mathbf{w}_{t-1} - \mathbf{w}_t\|_2^2
\end{aligned}$$

where we use the fact $\mathbf{w}_t = \mathbf{w}_{t-1} + y_t \mathbf{x}_t$ for $Z_t = 1$ and $M_t = 1$. Since $y_t \neq \hat{y}_t$, then $y_t \mathbf{w}_{t-1}^\top \mathbf{x}_t \leq 0$. Replacing $\mathbf{u}$ by $\alpha \mathbf{u}$ with $\alpha > 0$ and reorganize the inequality we have

$$\begin{aligned}
(\alpha\xi + |p_t|) M_t Z_t \leq \alpha \ell_\xi(y_t \mathbf{u}^\top \mathbf{x}_t) + \frac{1}{2}\|\alpha\mathbf{u} - \mathbf{w}_{t-1}\|_2^2 \\
- \frac{1}{2}\|\alpha\mathbf{u} - \mathbf{w}_t\|_2^2 + \frac{1}{2}\|\mathbf{w}_{t-1} - \mathbf{w}_t\|_2^2
\end{aligned}$$

We note that the above inequality holds for all $t = 1, \ldots, T$. If $M_t Z_t = 0$, then $\mathbf{w}_t = \mathbf{w}_{t-1}$, and the above inequality holds because $\alpha \ell_\xi(y_t \mathbf{u}^\top \mathbf{x}_t) \geq 0$. Summing the above inequality over $t = 1, \ldots, T_B$, we have

$$\begin{aligned}
&\sum_{t=1}^{T_B} (\alpha\xi + |p_t|) M_t Z_t \leq \alpha \sum_{t=1}^{T_B} \ell_\xi(y_t \mathbf{u}^\top \mathbf{x}_t) + \frac{1}{2}\|\alpha\mathbf{u} - \mathbf{w}_0\|_2^2 \\
&+ \frac{1}{2}\sum_{t=1}^{T_B} \|\mathbf{w}_{t-1} - \mathbf{w}_t\|_2^2 \\
&\leq \alpha \sum_{t=1}^{T_B} \ell_\xi(y_t \mathbf{u}^\top \mathbf{x}_t) + \frac{1}{2}\|\alpha\mathbf{u} - \mathbf{w}_0\|_2^2 + \sum_{t=1}^{T_B} \frac{\|\mathbf{x}_t\|_2^2}{2} M_t Z_t
\end{aligned}$$

Therefore

$$\sum_{t=1}^{T_B} (\alpha\xi + |p_t| - \frac{R^2}{2}) M_t Z_t \leq \alpha \sum_{t=1}^{T_B} \ell_\xi(y_t \mathbf{u}^\top \mathbf{x}_t) + \frac{\alpha^2}{2}\|\mathbf{u}\|_2^2$$

Due to $\alpha = \frac{c + R^2/2}{\xi}$, we then have

$$\sum_{t=1}^{T_B} (c + |p_t|) M_t Z_t \leq \alpha \sum_{t=1}^{T_B} \ell_\xi(y_t \mathbf{u}^\top \mathbf{x}_t) + \frac{\alpha^2}{2}\|\mathbf{u}\|_2^2$$

By taking expectation over randomness in Z_t and noting that $\mathrm{E}[Z_t] = \frac{c}{c + |p_t|}$, we have

$$\mathrm{E}\left[\sum_{t=1}^{T_B} c M_t\right] \leq \alpha \sum_{t=1}^{T_B} \ell_\xi(y_t \mathbf{u}^\top \mathbf{x}_t) + \frac{\alpha^2}{2}\|\mathbf{u}\|_2^2$$

□