Learning Relative Similarity from Data Streams: Active Online Learning Approaches

Shuji Hao[†], Peilin Zhao[‡], Steven C.H. Hoi[§], Chunyan Miao[¶]
[†]Joint NTU-UBC Research Centre of Excellence in Active Living for the Elderly (LILY), IGS, NTU, Singapore
[‡]Data Analytics Department, Institute for Infocomm Research, A*STAR, Singapore
[§]School of Information Systems, Singapore Management University, Singapore
[¶]School of Computer Engineering, Nanyang Technological University, Singapore
haos0001@e.ntu.edu.sg, zhaop@i2r.a-star.edu.sg, chhoi@smu.edu.sg, ascymiao@ntu.edu.sg

ABSTRACT

Relative similarity learning, as an important learning scheme for information retrieval, aims to learn a bi-linear similarity function from a collection of labeled instance-pairs, and the learned function would assign a high similarity value for a similar instance-pair and a low value for a dissimilar pair. Existing algorithms usually assume the labels of all the pairs in data streams are always made available for learning. However, this is not always realistic in practice since the number of possible pairs is quadratic to the number of instances in the database, and manually labeling the pairs could be very costly and time consuming. To overcome the limitation, we propose a novel framework of active online similarity learning. Specifically, we propose two new algorithms: (i) PAAS: Passive-Aggressive Active Similarity learning; (ii) CWAS: Confidence-Weighted Active Similarity learning, and we will prove their mistake bounds in theory. We have conducted extensive experiments on a variety of real-world data sets, and we find encouraging results that validate the empirical effectiveness of the proposed algorithms.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; D.2.8 [**Information Retrieval**]: Online Learning—*Active Learning*

Keywords

Image Retrieval, Active Learning, Online Learning

1. INTRODUCTION

Similarity learning, also known as distance learning, has been widely studied in machine learning and data mining communities. It plays a critical role in a wide range of real-world applications, such as ranking[4], recommendation system[1], image retrieval [21, 26, 2, 30, 17] and text information retrieval [9], etc. In previous research, a variety of machine learning approaches have been proposed for learning distance or similarity functions from training data [31, 29, 14]. One of the most notable schemes is the Distance Metric Learning (DML) approach, which aims to learn a Mahalanobis distance [31] and requires the distance metric satisfying the Positive Semi-Definiteness (PSD) property. However, imposing the PSD requirement often makes the DML task computationally challenging, particularly when handling large-scale training data in high dimensional spaces, although various approximate techniques have been proposed to improve computational efficiency. Another strong assumption of DML approaches is that the exact distances of all pairs of instances should be obtained, which are usually costly and time consuming.

Instead of learning the Mahalanobis distance, another technique is to explore the relative similarity learning [4], which aims to learn a bilinear similarity model for measuring the similarity of any instance pair (two instances) by eliminating the PSD requirement, which is thus much more computationally efficient and scalable. More importantly, only relative similarities among any two pairs of instances are needed, which greatly alleviates the requirement of exact distances among all instances. In literature, several online learning algorithms have been proposed for learning relative similarity from data streams. Most existing studies generally assume that the data streams of instance pairs are fully labeled. However, we argue this may not be a realistic setting since the number of instance pairs is quadratic with respect to the number of instances, and labeling such a massive pool of instance pairs could be extremely costly and time consuming for large-scale applications, especially when the labeling task has to explicitly involve human in the loop in an early stage of deploying a new system.

Unlike the existing relative similarity learning approaches, in this paper, we investigate a novel scheme of active online learning of relative similarity from unlabeled data streams without requiring labeling every instance pair. Specifically, we propose two efficient and scalable online learning algorithms to tackle the new problem: (i) PAAS: Passive-Aggressive Active Similarity learning and (ii) CWAS: Confidence-Weighted Active Similarity learning. We then analyze the theoretical bounds of the proposed online learning algorithms, and further validate their empirical performances via an extensive set of experiments on various real datasets. The encouraging results show that the proposed online active relative similarity learning algorithms can achieve highly competitive performance while significantly reducing the amount of labeling costs.

The rest of this paper is organized as follows. We first review related work, and then present the formulation of the proposed two algorithms as well as their theoretical analysis. Further, we empirically validate the proposed algorithms in terms of effectiveness and efficiency, and finally conclude this paper and discuss the future work.

CIKM'15, October 19–23, 2015, Melbourne, Australia.

DOI: http://dx.doi.org/10.1145/2806416.2806464.

2. RELATED WORK

This paper is mainly related to two major categories of studies in literature: similarity learning, and active learning. We briefly review each of them below.

Similarity learning has been extensively studied in machine learning and data mining communities. Most existing works have been focused on DML [31] for learning a Mahalanobis distance through a PSD matrix [16, 22, 24]. Weinberger et al. [28] proposed a large margin nearest-neighbor classifier to address the DML problem in the context of ranking. Globerson and Roweis [11] proposed to address the positive constraints in a supervised setting. Based on LogDet-regularization with different loss functions, Davis et al. [8] proposed online metric learning algorithms. All these algorithms aim to learn a PSD matrix $\mathbf{M}$, based on which the dissimilarity of two images $\mathbf{x}_1$ and $\mathbf{x}_2$ is measured by computing $(\mathbf{x}_1 - \mathbf{x}_2)^\top \mathbf{M}(\mathbf{x}_1 - \mathbf{x}_2)$. However, it is computationally costly to impose the PSD constraint on $\mathbf{M}$, which makes these algorithms inappropriate for large-scale problems. To overcome this limitation, the bilinear form similarity function $\mathbf{x}_1^\top \mathbf{M}\mathbf{x}_2$ is proposed, where $\mathbf{M}$ is not required to be PSD. Along this direction, Chechik et al. [4] proposed the first-order relative similarity learning algorithm "OASIS" based on the first-order based online learning algorithm [6, 35, 15]. Recently, Crammer and Chechik [5] proposed a second-order relative similarity learning algorithm "AROMA" based on second-order based online learning algorithm [7, 33]. However, all these algorithms require a large set of labeled data to train the similarity models, which is not always realistic in many real-world applications due to the expensive labeling cost.

Active learning is a learning technique for reducing labeling cost by querying the most informative examples for labeling, which has been extensively studied in machine learning literature [3]. Existing active learning algorithms could be generally classified into four categories: uncertainty-based [18, 27, 34], searching through the hypothesis space [10], minimizing the expected error and variance on the pool of unlabeled instances [20, 13], and exploiting the structure information [25] among the instances. More can be found in the comprehensive survey [23]. Our strategy is more related to the work in the first two categories. Specifically, we adopt the typical active learning strategy in the work [3, 32, 19], where the algorithm maintains a Bernoulli random variable $Z_t \in \{0, 1\}$ with probability $\delta/(\delta + |p_t|)$, and p_t is the margin of the similarity function on the t-th triplet. The algorithm will query the label and update the model only when $Z_t = 1$.

Despite being extensively studied [23], most of the active strategies are proposed for the classification or regression tasks. To our knowledge, there is no previous work exploring the active strategy on the relative similarity learning problem. To alleviate the cost in labeling, we explore the idea of active learning algorithms for overcoming the limitation of conventional relative similarity learning approaches.

3. ACTIVE ONLINE LEARNING OF RELATIVE SIMILARITY FROM DATA STREAMS

In this section, we first introduce the problem setting for active online learning of relative similarity from data streams, and then present the details of the proposed Active Online Similarity Learning algorithms.

3.1 Problem Formulation

Following [4], we would study the problem of online learning a relative similarity function $S(\mathbf{x}, \mathbf{x}')$, $\forall \mathbf{x}, \mathbf{x}' \in \mathbb{R}^d$, which measures the similarity between $\mathbf{x}$ and $\mathbf{x}'$. Formally, let

$$\{(\mathbf{x}_t, \mathbf{x}_t^1, \mathbf{x}_t^2; y_t) \in \mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R}^d \times \{-1, +1\} | t \in [T]\}$$

be a sequence of triplets (where $[T] = \{1, \ldots, T\}$) . For the t-th triplet, $y_t = 1$ indicates $\mathbf{x}_t$ instance is more similar with $\mathbf{x}_t^1$ than $\mathbf{x}_t^2$, while $y_t = -1$ implies $\mathbf{x}_t$ is less similar with $\mathbf{x}_t^1$ than $\mathbf{x}_t^2$. Our goal is to learn a similarity function $S(\mathbf{x}, \mathbf{x}')$ that assigns a higher similarity score to the similar pair and a lower similarity score to the non-similar pair. Formally, given a triplet$(\mathbf{x}_t, \mathbf{x}_t^1, \mathbf{x}_t^2; y_t)$, the learned similar function should satisfy:

$$y_t[S(\mathbf{x}_t, \mathbf{x}_t^1) - S(\mathbf{x}_t, \mathbf{x}_t^2)] \geq 0, \quad \forall t \in [T].$$

For the similarity function, we specifically adopt a linear similarity function that has a bi-linear form,

$$S(\mathbf{x}, \mathbf{x}') = \mathbf{x}^\top \mathbf{M}\mathbf{x}', \tag{1}$$

where $\mathbf{M} \in \mathbb{R}^{d \times d}$. To note, it is possible to learn a $\mathbf{M} \in \mathbb{R}^{d \times d'}$, where $d \neq d'$, as similarity function between two different spaces, however we will assume $d = d'$ for simplicity.

To learn the parameter $\mathbf{M}$, we could introduce some loss functions to measure its performance on the t-th triplet, for example, the hinge loss is defined as

$$\ell(M; (\mathbf{x}_t, \mathbf{x}_t^1, \mathbf{x}_t^2; y_t)) = \Big[1 - y_t[S(\mathbf{x}_t, \mathbf{x}_t^1) - S(\mathbf{x}_t, \mathbf{x}_t^2)]\Big]_+, \tag{2}$$

where the hinge loss $[\cdot]_+ = \max(0, \cdot)$ encourages $y_t[S(\mathbf{x}_t, \mathbf{x}_t^1) - S(\mathbf{x}_t, \mathbf{x}_t^2)] \geq 1$. It is also possible to define other loss functions, e.g. logistic loss. In this article, we adopt the hinge loss for simplicity.

To solve this online relative similarity learning task, we can adopt those existing methods including Perceptron, Online Gradient Descent, Online Passive Aggressive, etc. Although these online learning algorithms can achieve a cumulative loss comparable with the one attained by any fixed hypothesis, one major limitation is that they require all the labels of the instances while labeling instances might be very expensive or time consuming. To solve this problem, we would study active online similarity learning, which will try to only query the labels of a few informative instances so that the number of queried labels is reduced while the performance dose not degrade too much.

3.2 PAAS: Passive-Aggressive Active Similarity learning

To solve Active Online Similarity Learning, we propose to adopt the selective sampling technique used in [3]. Specifically we will use a stochastic sampling scheme to decide whether it is necessary to query the label of the current instance. This scheme maintains a Bernoulli random variable $Z_t \in \{0, 1\}$, where $Z_t = 1$ indicates the label should be queried at the t-th step. More specifically, this scheme employs the following sampling probability

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

where $\delta > 0$ is a parameter to tune the number of queried labels and

$$p_t = \mathbf{x}_t^\top \mathbf{M}_t(\mathbf{x}_t^1 - \mathbf{x}_t^2) \tag{4}$$

is the margin of similarities between $\mathbf{x}_t^\top \mathbf{M}\mathbf{x}_t^1$ and $\mathbf{x}_t^\top \mathbf{M}\mathbf{x}_t^2$. Intuitively, the smaller the value of $|p_t|$, the lower the confidence of the model on the current prediction; as a consequence, we should query the label with a higher probability of $\Pr(Z_t = 1)$. Once

Algorithm 1 PAAS: The proposed algorithm of Passive-Aggressive Active Similarity learning

Input: smooth parameters $\delta > 0, C > 0$
Initialize: $\mathbf{M}_1 = 0$
for $t = 1, 2, \ldots, T$ **do**
  Receive $(\mathbf{x}_t, \mathbf{x}_t^1, \mathbf{x}_t^2)$ and compute $X_t = \mathbf{x}_t(\mathbf{x}_t^1 - \mathbf{x}_t^2)^\top$;
  Compute $p_t = \text{Tr}(\mathbf{M}_t \mathbf{X}_t^\top)$ and $\hat{y}_t = \text{sign}(p_t)$;
  Sample $Z_t \in \{0, 1\}$ with $\Pr(Z_t = 1) = \frac{\delta}{\delta + |p_t|}$;
  if $Z_t = 1$ **then**
    Query y_t and compute $\ell_t(\mathbf{M}_t) = \max(0, 1 - y_t p_t)$;
    Update: $\mathbf{M}_{t+1} = \mathbf{M}_t + \tau_t y_t \mathbf{X}_t$;
    where $\tau_t = \min\left\{C, \frac{\ell_t(\mathbf{M}_t)}{\|\mathbf{X}_t\|_F^2}\right\}$.
  else
    $\mathbf{M}_{t+1} = \mathbf{M}_t$;
  end if
end for
Output: $\mathbf{M}_{T+1}$

the label is queried, the algorithm will update the model using the online Passive Aggressive strategy, i.e.,

$$\mathbf{M}_{t+1} = \arg\min_{\mathbf{M}} \left\{\frac{1}{2}\|\mathbf{M} - \mathbf{M}_t\|_F^2 + C\ell_t(M)\right\},$$

where $\ell_t(M) = \ell(M; (\mathbf{x}_t, \mathbf{x}_t^1, \mathbf{x}_t^2; y_t))$ and $C > 0$ is a trade-off between minimizing the adjustment of the model and minimizing the loss of the new model on the current example. This will produce a first-order update method [4, 6], i.e.,

$$\mathbf{M}_{t+1} = \mathbf{M}_t + \tau_t y_t \mathbf{X}_t,$$

where $\tau_t = \min\left\{C, \frac{\ell_t(\mathbf{M}_t)}{\|\mathbf{X}_t\|_F^2}\right\}$, and $\mathbf{X}_t = \mathbf{x}_t(\mathbf{x}_t^1 - \mathbf{x}_t^2)^\top$. Finally, the proposed Passive-Aggressive Active Similarity learning (PAAS) algorithm is summarized in Algorithm 1.

3.3 CWAS: Confidence-Weighted Active Similarity learning

To improve the learning performance, a second order similarity learning algorithm AROMA is proposed [5], which does not only use the first order information but also the second-order information, i.e., the covariance matrix for all the features, to update the model. Compared to the first-order algorithm OASIS, the effectiveness of AROMA has been theoretically and empirically verified. However, directly combining AROMA with query strategy in Equation (3) makes the theoretical analysis challenging.

Following the similar idea of AROMA, we assume the model maintains a Gaussian distribution, $\mathcal{N}(\mathbf{M}, \mathbf{\Sigma})$, where $\mathbf{M} \in \mathbb{R}^{d \times d}$ and $\mathbf{\Sigma} \in \mathbb{R}^{d^2 \times d^2}$ encode the model's knowledge of and confidence of the model. At the t-th round, when receiving $(\mathbf{x}_t, \mathbf{x}_t^1, \mathbf{x}_t^2)$, we firstly decide whether it is necessary to query the true label based on Equation (3). If $Z_t = 1$, we query the label and update the distribution by minimizing the following objective function

$$f_t(\mathbf{M}, \mathbf{\Sigma}) = D_{KL}\big(\mathcal{N}(\text{vec}(\mathbf{M}), \mathbf{\Sigma}) \| \mathcal{N}(\text{vec}(\mathbf{M}_t), \mathbf{\Sigma}_t)\big) + \eta \text{Tr}(\mathbf{G}_t^\top \mathbf{M}) + \frac{1}{2\gamma}\text{vec}(\mathbf{X}_t)^\top \mathbf{\Sigma} \text{vec}(\mathbf{X}_t),$$

where

$$D_{KL}(\mathcal{N}(\mu, \mathbf{\Sigma}) \| \mathcal{N}(\mu_t, \mathbf{\Sigma}_t)) = \frac{1}{2}\Big[\ln(\frac{|\mathbf{\Sigma}_t|}{|\mathbf{\Sigma}|}) + \text{Tr}(\mathbf{\Sigma}_t^{-1}\mathbf{\Sigma}) + (\mu_t - \mu)^\top \mathbf{\Sigma}_t^{-1}(\mu_t - \mu) - d\Big],$$

is the Kullback-Leibler divergence of two distributions, $\mathbf{G}_t = \partial \ell_t(\mathbf{M}_t) = -y_t \mathbf{X}_t$, where $\mathbf{X}_t = \mathbf{x}_t(\mathbf{x}_t^1 - \mathbf{x}_t^2)^\top$ and $\text{vec}(\mathbf{X}) = [\mathbf{X}_{11}, \ldots, \mathbf{X}_{1d}, \ldots, \mathbf{X}_{d1}, \ldots, \mathbf{X}_{dd}]^\top$. The first term is to keep the new distribution not far away from the old one. The second term is a first order approximation of the current loss function at the current model, which is used to minimize the loss of the new model on the current example. The final term is to update the confidence of the model, since a new triplet is observed. η and γ are two positive parameters to trade off these objectives.

Algorithm 2 CWAS: The proposed algorithm of Confidence-Weighted Active Similarity learning

Input: learning rate η; regularization parameter γ
Initialize: $\mathbf{\Sigma}_1 = I^{d^2 \times d^2}$, $\mathbf{M}_1 = 0^{d \times d}$
for $t = 1, 2, \ldots, T$ **do**
  Receive $(\mathbf{x}_t, \mathbf{x}_t^1, \mathbf{x}_t^2)$ and compute $\mathbf{X}_t = \mathbf{x}_t(\mathbf{x}_t^1 - \mathbf{x}_t^2)^\top$;
  $p_t = \text{Tr}(\mathbf{M}_t^\top \mathbf{X}_t)$, and $\hat{y}_t = \text{sign}(p_t)$;
  Sample $Z_t \in \{0, 1\}$ with $\Pr(Z_t = 1) = \frac{\delta}{\delta + |p_t|}$;
  if $Z_t = 1$ **then**
    Query y_t, and compute $\ell_t(\mathbf{M}_t) = [1 - y_t p_t]_+$;
    if $\ell_t(\mathbf{M}_t) > 0$ **then**
      $\mathbf{\Sigma}_{t+1} = \mathbf{\Sigma}_t - \frac{\mathbf{\Sigma}_t \text{vec}(\mathbf{X}_t)\text{vec}(\mathbf{X}_t)^\top \mathbf{\Sigma}_t}{\gamma + \text{vec}(\mathbf{X}_t)^\top \mathbf{\Sigma}_t \text{vec}(\mathbf{X}_t)}$;
      $\mathbf{G}_t = \partial \ell_t(\mathbf{M}_t) = -y_t \mathbf{X}_t$;
      $\mathbf{M}_{t+1} = \text{mat}\Big[\text{vec}(\mathbf{M}_t) - \eta \mathbf{\Sigma}_{t+1} \text{vec}(\mathbf{G}_t)\Big]$;
    else
      $\mathbf{\Sigma}_{t+1} = \mathbf{\Sigma}_t$, $\mathbf{M}_{t+1} = \mathbf{M}_t$;
    end if
  else
    $\mathbf{\Sigma}_{t+1} = \mathbf{\Sigma}_t$, $\mathbf{M}_{t+1} = \mathbf{M}_t$;
  end if
end for
Output: $\mathbf{M}_{T+1}$

To solve this optimization problem, we can set the derivatives $\partial_{\mathbf{M}} f_t(\mathbf{M}_{t+1}, \mathbf{\Sigma})$ and $\partial_{\mathbf{\Sigma}} f_t(\mu, \mathbf{\Sigma}_{t+1})$ as zeros, respectively to get the following updating rules:

$$\mathbf{M}_{t+1} = \text{mat}\Big[\text{vec}(\mathbf{M}_t) - \eta \mathbf{\Sigma}_t \text{vec}(\mathbf{G}_t)\Big], \quad (5)$$

$$\mathbf{\Sigma}_{t+1} = \mathbf{\Sigma}_t - \frac{\mathbf{\Sigma}_t \text{vec}(\mathbf{X}_t)\text{vec}(\mathbf{X}_t)^\top \mathbf{\Sigma}_t}{\gamma + \text{vec}(\mathbf{X}_t)^\top \mathbf{\Sigma}_t \text{vec}(\mathbf{X}_t)}, \quad (6)$$

where $\text{mat}(\cdot)$ is the inverse function of $\text{vec}(\cdot)$.

Finally, we summarize the proposed CWAS in Algorithm 2.

Remark One key drawback of this algorithm is that the dimension of $\mathbf{\Sigma}_t$ is $d^2 \times d^2$, which will result in very high memory and computational complexities. In practice, we can use diagonal $\mathbf{\Sigma}_t$ to reduce these complexities. Or equivalently, we can store a matrix $\mathbf{\Sigma}_t \in \mathbb{R}^{d \times d}$, and change the Equations (5) and (6) into

$$\mathbf{M}_{t+1} = \mathbf{M}_t - \eta \mathbf{\Sigma}_t \odot \mathbf{G}_t, \quad (7)$$

$$\mathbf{\Sigma}_{t+1} = \mathbf{\Sigma}_t - \frac{\mathbf{\Sigma}_t \odot \mathbf{X}_t \odot \mathbf{X}_t \odot \mathbf{\Sigma}_t}{\gamma + \sum_{ij}(\mathbf{X}_t \odot \mathbf{\Sigma}_t \odot \mathbf{X}_t)_{ij}}, \quad (8)$$

where $\odot$ is element-wise product.

3.4 Theoretical Analysis

Denote $m_t = \mathbb{I}(y_t \neq \hat{y}_t)$, we would analyze the performance of the proposed two algorithms in terms of expected mistake bounds $\mathbb{E}[\sum_{t=1}^T m_t]$. We will provide all the detailed proofs in the appendix. Firstly, we have the following theorem for the first-order algorithm PAAS.

THEOREM 1. *If the PAAS algorithm is run with a sequence of triplets* $(\mathbf{x}_t, \mathbf{x}_t^1, \mathbf{x}_t^2; y_t)$, $t \in [T]$, *with* $D_X = \max \|X_t\|_F$, *then*

for any $T > 0$, and $\mathbf{M} \in \mathbb{R}^{d\times d}$, we have

$$\mathbb{E}[\sum_{t=1}^{T} m_t] \leq \frac{\beta}{\delta}\Big\{(\frac{\delta+1}{2})^2\|\mathbf{M}\|_F^2 + (\delta+1)CL_T(\mathbf{M})\Big\},$$

where $\beta = \max(1/C, D_X^2)$, and $L_T(\mathbf{M}) = \sum_{t=1}^{T}\ell_t(\mathbf{M})$. In addition, the expected number of labels queried equals to $\sum_{t=1}^{T}\mathbb{E}[\frac{\delta}{\delta+|p_t|}]$.

Remark: The above bound depends on the choice of parameter δ. Generally, δ could be viewed as a parameter to rule the extent to which the learning model fits the present data [3]. Minimizing the right hand side over δ, we can observe that setting $\delta = \sqrt{1 + 4CL_T(\mathbf{M})/\|\mathbf{M}\|_F^2}$ in the above theorem gives the following upper bound for $\mathbb{E}[\sum_{t=1}^{T} m_t]$

$$\beta\{\frac{1}{2}\|\mathbf{M}\|_F^2 + CL_T(\mathbf{M}) + \frac{1}{2}\|\mathbf{M}\|_F\sqrt{\|\mathbf{M}\|_F^2 + 4CL_T(\mathbf{M})}\}.$$

Under the same assumptions as in the above theorem, we have the following theoretical guarantee for the second-order algorithm CWAS.

THEOREM 2. *If the algorithm CWAS is run with a sequence of examples $(\mathbf{x}_t, \mathbf{x}_t^1, \mathbf{x}_t^2; y_t)$, $t \in [T]$, then for any $T > 0$, and $\mathbf{M} \in \mathbb{R}^{d\times d}$, we have*

$$\mathbb{E}\sum_{t=1}^{T}[m_t] \leq \sum_{t=1}^{T} Z_t\ell_t(\mathbf{M}) + \frac{1}{\eta\delta}(D_M + |1-\delta|\|\mathbf{M}\|)^2 Tr(\mathbf{\Sigma}_{T+1}^{-1}) + \frac{\eta\gamma}{2\delta}\ln(|\mathbf{\Sigma}_{T+1}^{-1}|).$$

Setting $\eta = (D_M + |1-\delta|\|\mathbf{M}\|)\sqrt{\frac{2}{\gamma}Tr(\mathbf{\Sigma}_{T+1}^{-1})/\ln(|\mathbf{\Sigma}_{T+1}^{-1}|)}$, the following bound holds for CWAS

$$\mathbb{E}[\sum_{t=1}^{T} m_t] \leq L_T(\mathbf{M}) + \frac{|1-\delta|\|\mathbf{M}\|_F^2 + D_M}{\delta}\sqrt{2\gamma Tr(\mathbf{\Sigma}_{T+1}^{-1})\ln|\mathbf{\Sigma}_{T+1}^{-1}|},$$

where $D_M = \max_t \|\mathbf{M}_t - \mathbf{M}\|_F^2$.

Remark By optimizing over δ, and setting $\delta = 1$, we will have

$$\mathbb{E}[\sum_{t=1}^{T} m_t] \leq L_T(\mathbf{M}) + D_M\sqrt{2\gamma \mathrm{Tr}(\mathbf{\Sigma}_{T+1}^{-1})\ln|\mathbf{\Sigma}_{T+1}^{-1}|}.$$

4. EXPERIMENTS

In this section, we conduct experiments to evaluate the efficacy of the proposed algorithms for online similarity learning from data streams on several benchmark datasets.

4.1 Datasets

To examine the performance, we conduct extensive experiments on a variety of benchmark datasets from web machine learning repositories. Table 1 shows the details of five datasets used in our experiments. Caltech256 [12] is a standard dataset for the image classification and ranking problem [1]. We use the same classes with [4] to generate the *caltech50* and *caltech249*. The other datasets are standard machine learning datasets publicly available at LIBSVM [2] and UCI Machine Learning Repository [3].

[1] http://www.vision.caltech.edu/Image_Datasets/Caltech256/

[2] http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/

[3] https://archive.ics.uci.edu/ml/datasets.html

Table 1: Datasets used in the following experiments

Dataset	# Classes	# Instances	# Features	# Triplets
caltech249	249	16185	1000	1200000
caltech50	50	3250	1000	250000
covtype	7	455	54	10000
letter	26	1690	16	10140
pendigits	10	650	16	16848
satimage	6	390	36	18000
segment	7	455	19	25000
shuttle	7	458	9	4000

We evaluated the performance of all algorithms using precision-at-top-k, a standard ranking precision measure based on nearest neighbors. For each query instance in the test set, all other test instances were ranked according to their similarity to the query instance based on Equation (1), and the number of same-class instances among the top k instances (the k nearest neighbors) is computed, and then averaged across test instances. This measure is short named as AP. We also calculated the mean Average Precision (mAP), a measure that is widely used in the information retrieval community. For both of the AP and mAP measures, we set $k = 10$.

On all datasets, we used standard 5-fold cross validation and report both the AP and the mAP on test set. In Table 1, *# Triplets* represents the number of triplets $(\mathbf{x}_t, \mathbf{x}_t^1, \mathbf{x}_t^2)$ constructed based on the training set in each fold.

4.2 Comparison Schemes

To examine the efficacy, we compare the proposed algorithms with the state-of-the-art relative similarity learning algorithms, where they have to query every triplet's label during training stage.

- **OASIS:** the state-of-the-art first-order online similarity learning algorithm [4]. It should be noted that this algorithm is also the passive version of our proposed PAAS algorithm;
- **AROMA:** the state-of-the-art second-order online similarity learning algorithm [5], and **AROMA-d** represents its diagonal version.

To our knowledge, there is no work on exploring active learning on the relative similarity learning, so we compare our proposed algorithms with their random versions, where they randomly decide whether to query the label of t-th triplet.

- **PAAS:** the proposed first-order Passive-Aggressive Active online Similarity Learning as shown in Algorithm 1;
- **PARS:** a variant of the proposed algorithm PAAS, where we use a uniform distribution to randomly query labels and keep the query ratio same as PAAS;
- **CWS:** the passive version of the algorithm CWAS, where we query each triplet in the training phrase.
- **CWAS:** the proposed Confidence-Weighted Active online Similarity Learning as shown in Algorithm 2. **CWAS-d** is the diagonal version with Equation (7) and (8) as updating rules.
- **CWRS:** a variant of the algorithm CWAS, where we use a uniform distribution to randomly query labels and keep the query ratio same as CWAS. **CWRS-d** is the diagonal version with Equation (7) and (8) as updating rules;

Table 2: Performance of algorithms on different datasets with the query ratio fixed to about 20%

	cotype				letter			
Alg.	Query(%)	AP@10	mAP@10	Time(s)	Query(%)	AP@10	mAP@10	Time(s)
PARS	20.620 ± 0.955	0.446 ± 0.036	0.334 ± 0.035	0.006 ± 0.008	20.6 ± 0.827	0.198 ± 0.024	0.127 ± 0.025	0.007 ± 0.000
PAAS	20.710 ± 0.336	0.512 ± 0.035	0.409 ± 0.027	0.006 ± 0.008	20.6 ± 0.667	0.233 ± 0.020	0.158 ± 0.023	0.007 ± 0.000
CWRS	20.960 ± 0.670	0.585 ± 0.027	0.476 ± 0.040	11.860 ± 1.741	19.4 ± 1.078	0.362 ± 0.019	0.276 ± 0.021	0.104 ± 0.005
CWAS	20.890 ± 0.677	$\mathbf{0.624 \pm 0.038}$	$\mathbf{0.515 \pm 0.043}$	22.390 ± 1.448	19.4 ± 0.507	$\mathbf{0.385 \pm 0.012}$	$\mathbf{0.298 \pm 0.012}$	0.134 ± 0.005
	pendigits				satimage			
Alg.	Query(%)	AP@10	mAP@10	Time(s)	Query(%)	AP@10	mAP@10	Time(s)
PARS	22.2 ± 0.378	0.527 ± 0.056	0.418 ± 0.063	0.010 ± 0.001	20.4 ± 0.387	0.493 ± 0.044	0.408 ± 0.040	0.046 ± 0.001
PAAS	22.2 ± 0.526	0.554 ± 0.027	0.452 ± 0.024	0.010 ± 0.001	20.5 ± 0.274	0.509 ± 0.031	0.423 ± 0.022	0.050 ± 0.001
CWRS	19.3 ± 1.068	0.673 ± 0.020	0.588 ± 0.032	0.181 ± 0.016	19.6 ± 0.860	0.643 ± 0.010	0.560 ± 0.013	4.476 ± 0.257
CWAS	19.3 ± 0.925	$\mathbf{0.700 \pm 0.010}$	$\mathbf{0.619 \pm 0.018}$	0.287 ± 0.021	19.5 ± 0.517	$\mathbf{0.657 \pm 0.008}$	$\mathbf{0.574 \pm 0.011}$	9.390 ± 0.446
	segment				shuttle			
Alg.	Query(%)	AP@10	mAP@10	Time(s)	Query(%)	AP@10	mAP@10	Time(s)
PARS	21.2 ± 0.366	0.361 ± 0.034	0.253 ± 0.041	0.020 ± 0.001	21.07 ± 0.898	0.435 ± 0.058	0.313 ± 0.054	0.001 ± 0.000
PAAS	21.2 ± 0.607	0.436 ± 0.029	0.316 ± 0.031	0.021 ± 0.000	20.64 ± 0.807	0.513 ± 0.027	0.403 ± 0.042	0.001 ± 0.000
CWRS	19.2 ± 0.817	0.690 ± 0.074	0.575 ± 0.103	0.297 ± 0.024	20.25 ± 1.117	0.502 ± 0.037	0.417 ± 0.059	0.006 ± 0.001
CWAS	19.2 ± 0.781	$\mathbf{0.787 \pm 0.019}$	$\mathbf{0.729 \pm 0.039}$	0.573 ± 0.036	20.84 ± 0.873	$\mathbf{0.561 \pm 0.024}$	$\mathbf{0.493 \pm 0.017}$	0.009 ± 0.001

For all algorithms, the parameters are searched within the space $10^{[-5:1:5]}$ using cross validation. For both PAAS and CWAS algorithms, we evaluate their performances on 10 different query ratios which are achieved by setting the sampling threshold $\delta = 2^{[-10:2:10]}$. The query ratio represents the ratio of queried labels over the total number of triplets in the data stream. PARS and CWRS are also evaluated on 10 query ratios by setting the random sampling parameters according to the query ratios in PAAS and CWAS, respectively.

4.3 Evaluation of Fixed Query Ratio

In this experiment, we evaluate the performance of the proposed algorithms with some fixed query ratios. Table 2 shows the experimental results on different datasets.

First of all, with the same query ratio, the proposed active similarity learning algorithms (CWAS and PAAS) consistently outperform their random versions (CWRS and PARS), respectively. Besides, CWAS greatly outperforms PAAS over all the datasets. Moreover, both CWAS and PAAS also achieve smaller variances as compared to their random algorithms (PARS and CWRS), respectively. These observations further confirm the effectiveness and robustness of the proposed algorithms.

Secondly, on all the datasets, the proposed second-order active online similarity learning algorithm CWAS always achieves the best performance and the smallest variance. This is consistent with the previous results as shown in Figure 1 and 3, which further confirms the effectiveness of exploiting the second-order information.

Thirdly, by examining the running time cost, PAAS and CWAS spent slightly more time cost as compared to their random variants, respectively, mainly due to the cost of computing the query strategies.

In addition, CWAS in general spends more time than PAAS due to the computation of the second order information. However, the extra time cost could almost be ignored considering the high efficiency of the online learning scheme. In practice, for the higher dimensional datasets, we could adopt the diagonal version algorithm "CWAS-d" to further reduce the running time cost, as shown in subsequent experiments.

4.4 Evaluation of Varied Query Ratios

In this section, we evaluate the performance of the proposed algorithms with varied query ratios as shown in Figure 1. Several observations could be drawn from the results.

Firstly, we can observe that the proposed active online learning algorithms can consistently outperform their random versions, respectively. This observation is consistent with the one in Table 2 and confirms the effectiveness of the proposed algorithms on selecting informative instances. More importantly, with around 30% query ratio, the proposed algorithm CWAS could achieve the similar performance as CWS which queries all. This means that the proposed second-order active algorithm CWAS could save us around 70% effort compared to the traditional passive algorithms. For the proposed first-order active algorithm, PAAS could save us around 50% effort compared to the passive algorithm OASIS.

Secondly, the proposed second-order algorithm CWAS can outperform the first-order algorithm PAAS. From Figure1, we can observe CWAS could outperform PAAS on most of the datasets from the very small query ratio. The same result also could be observed on the CWRS compared to the PARS. These findings confirmed the effectiveness of introducing the second-order information. From the figure, we also could observe that the baseline second-order algorithm AROMA could achieve a little higher performance than our proposed algorithm CWS, on some datasets. One possible reason may be that the learning rate η is fixed in our algorithm in the Equation (5), while it is adaptive in the algorithm AROMA [7]. However, CWS in general is comparable with AROMA, while its active version is much easier to be theoretically analyzed.

4.5 Evaluation of Parameter Sensitivity

In this section, we evaluate the sensitivity of the parameters both in Algorithm 1 and 2, respectively. However, it is difficult to evaluate the algorithms with active query strategies. Thus, we evaluate their passive versions OASIS and CWS, respectively. Figure 2 shows the varied performances corresponding to different parameters on four of the datasets. For each dataset, the left figure shows the performance (Y-axis) corresponding to the parameter C (X-axis) in OASIS, and the right figure shows the performance (different color) corresponding to parameter γ (X-axis) and η (Y-axis)

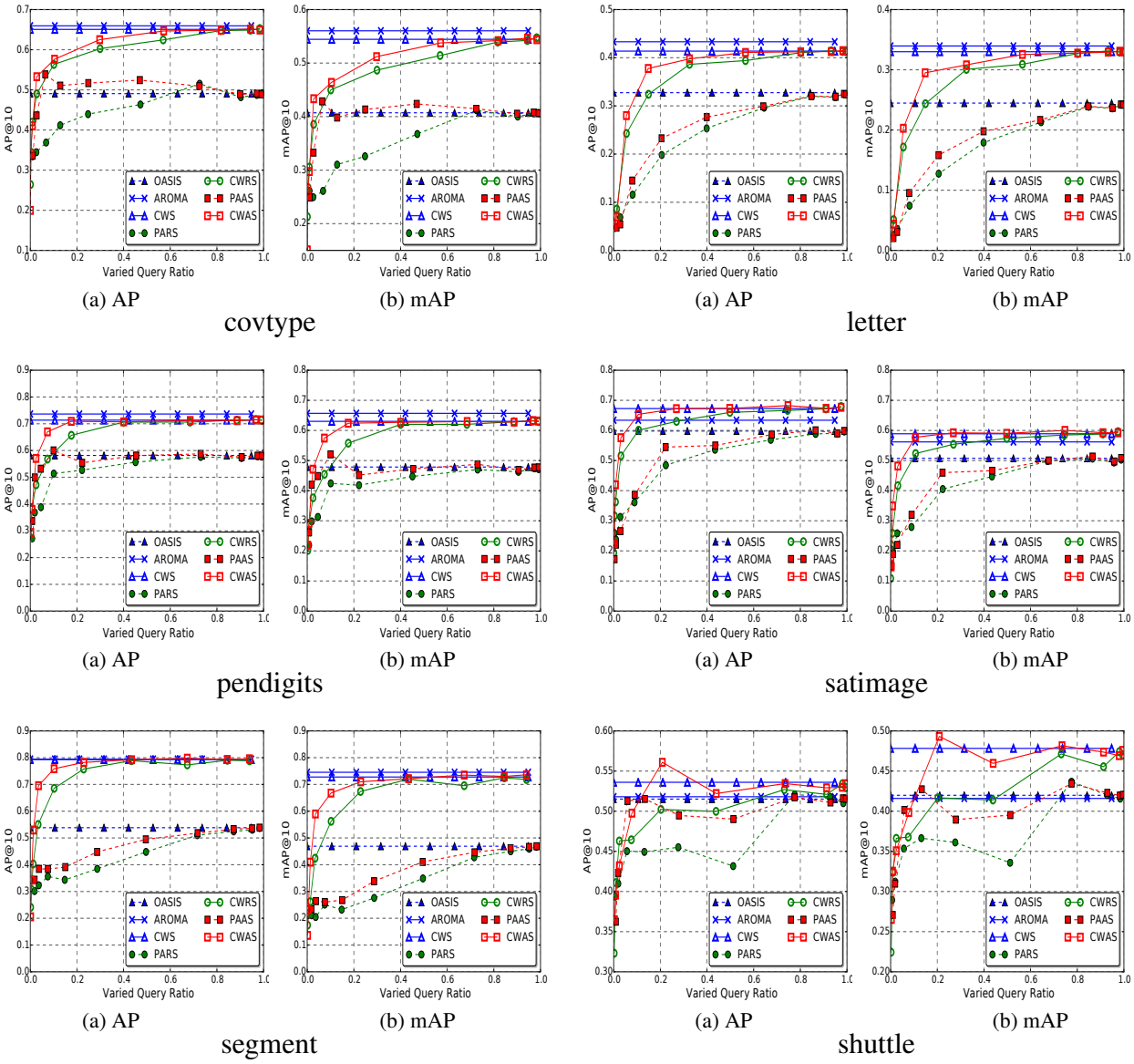

Figure 1: Performance with varied query ratios on the test datasets, where AP@10 denotes Average Precision at 10, and mAP@10 denotes mean Average Precision at 10.

in CWS, where bright color corresponds to higher performance in terms of Average Precision at 10 than the dark color.

For the parameter C, we can see it greatly affects the results of OASIS, on all of the datasets, which could be explained using our mistake bound in Theorem 1. Specifically, the mistake bound could be divided into two terms, and the parameter C divides the first term and multiplies the second term. When C is too small, the first term will dominate the mistake bound, and the second term will dominate the mistake bound when C is too large. This is consistent with the results shown in Figure 2. For most of the datasets, $C = 1$ would present a promising result.

For algorithm CWS, the relationship between the performance and the parameters is relatively complex. From Figure 2, we can observe that the learning rate parameter η should be not too small in general. And the regularization parameter γ should be search around 1.

4.6 Evaluation of Efficiency and Scalability

Although the second-order algorithm CWAS greatly outperforms the first-order algorithm PAAS, it is costly to maintain the covariance matrix when the number of features is large. To reduce the complexity, we can use their diagonal versions by Equations (7) and (8). To test the diagonal algorithm CWAS-d, we carried out experiments on large-scale datasets *caltech50* and *caltech249* shown in Figure 3, where we can observe similar results as the one in Figure 1. In addition, the increase of performance of CWAS-d over PAAS is not as large as CWAS in Figure 1 as expected, since only part of the second-order information is used.

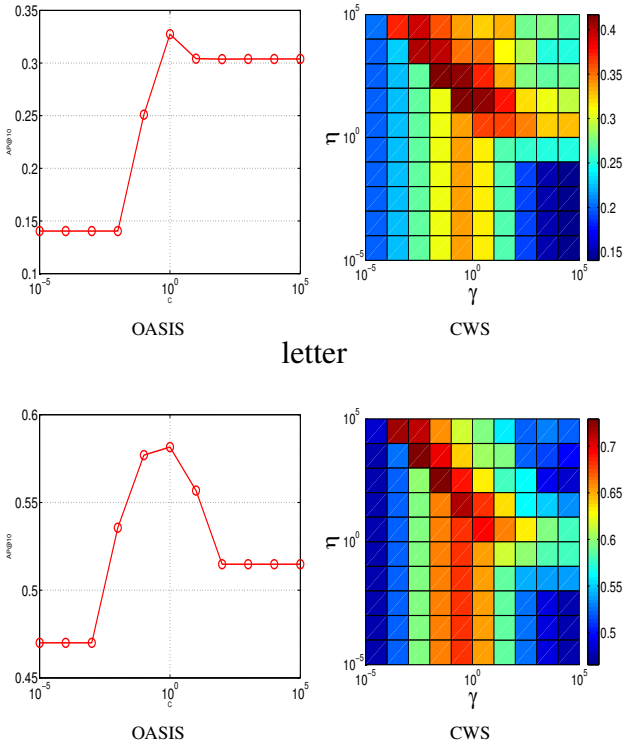

Figure 2: Sensitivity of the parameters

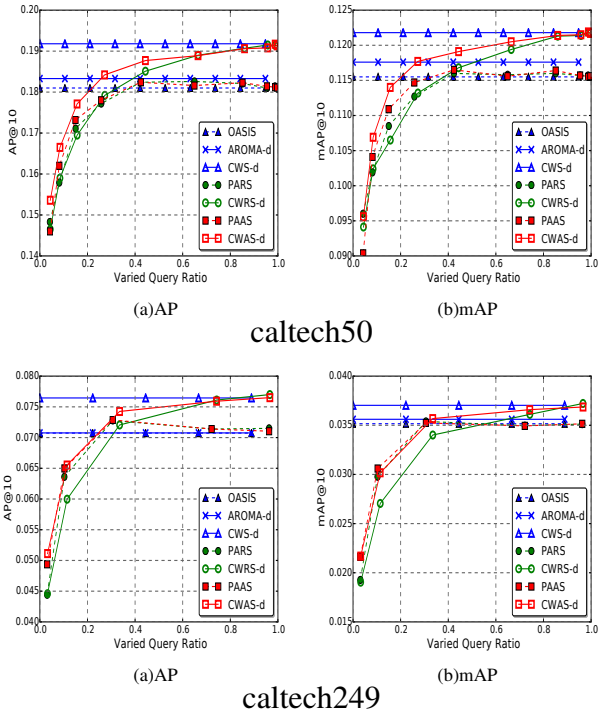

Figure 3: Performance on the large-scale datasets

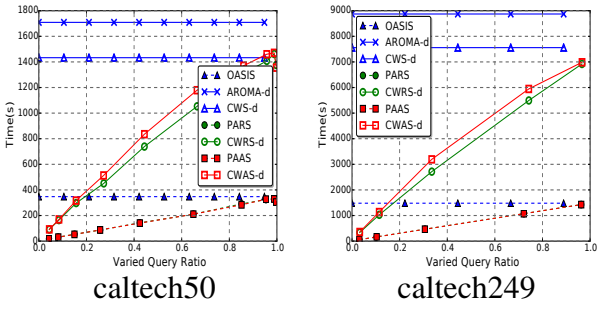

Figure 4: Time cost corresponding to varied querying ratio on large-scale datasets

In Figure 4, we evaluate the relationship between the querying ratio and the time-cost of our proposed algorithms. From the figure, we can see that the time cost of the proposed algorithms is lineally increasing with respect to the increasing querying ratio. Besides, CWAS-d cost a little extra time than its random version CWRS-d due to the computation of the query strategy, and PAAS costs almost the same as its random version PARS. What's more, the second-order algorithm CWAS-d cost more than the first-order algorithm PAAS due to the computation of the second-order information, this is consistent with the finding in Table 2. More importantly, in Figure 3, we can observe that with around 30% query ratio, the proposed algorithm PAAS and CWAS-d could obtain similar performances as their passive versions OASIS and CWS-d, respectively. Meanwhile, when the query ratio is around 30%, in Figure 4, PAAS and CWAS-d only spends around 30% of the time spent by OASIS and CWS-d , respectively. These observations il-

lustrate that our proposed algorithms can save a lot effort in labeling and training the model without sacrificing performance compared to the passive algorithms which query the labels of all the triplets.

5. CONCLUSIONS

To overcome the critical limitation of traditional passive online similarity learning from data streams, in this paper, we proposed a novel framework of active online learning for relative similarity learning. Specifically, we proposed two active online similarity learning algorithms for reducing the number of queried labels in the learning process. We theoretically analyzed the bounds of the proposed algorithms and conducted extensive experiments to examine the effectiveness of their empirical performance. The encouraging empirical results validate the effectiveness and efficiency of the proposed algorithms.

There are several aspects we are interested to explore in future. Firstly, it would be more interesting to design an automatic method to assign the parameter δ to control the query ratio, currently, it is manually assigned.

Secondly, for the proposed second-order based algorithm CWAS, it would be more effective to consider the second-order information when designing the query strategy, such as the covariance information contained in $\mathbf{\Sigma}$.

Thirdly, we are interested to design a real online active relative similarity learning where the labels of instance come from several noisy online workers, such as from the crowdsourcing platforms.

6. ACKNOWLEDGEMENT

This research is supported by the National Research Foundation, Prime Minister's Office, Singapore under its IDM Futures Funding Initiative and administered by the Interactive and Digital Media Programme Office. This research is also partially supported by Singapore MOE tier 1 research grant (C220/MSS14C003).

Appendix

In this appendix, we provide the proofs of the theorems in the section 3.4.

Theoretical Analysis on PAAS

To prove the Theorem 1, we need the following lemma.

LEMMA 1. *Let $(\mathbf{x}_t, \mathbf{x}_t^1, \mathbf{x}_t^2; y_t)$, $t \in [T]$ be a sequence of input triplets, where $\mathbf{x}_t, \mathbf{x}_t^1, \mathbf{x}_t^2 \in \mathbb{R}^d$ and $y_t \in \{-1, +1\}$ for all t. Let τ_t be the step size parameter for PAAS as given in the algorithm. Then the following bound holds for any $\mathbf{M} \in \mathbb{R}^{d\times d}$:*

$$\sum_{t=1}^{T} 2\tau_t Z_t \left[l_t(\alpha - |p_t|) + m_t(\alpha + |p_t|)\right] \leq \alpha^2\|\mathbf{M}\|_F^2 + \sum_{t=1}^{T}\tau_t^2\|\mathbf{X}_t\|_F^2 + \sum_{t=1}^{T} 2\alpha\tau_t\ell_t(\mathbf{M}),$$

where $l_t = \mathbb{I}(\ell_t(\mathbf{M}_t) > 0 \text{ and } \mathrm{sign}(\mathrm{p_t}) = \mathrm{y_t})$, $m_t = \mathbb{I}(\mathrm{sign}(p_t) \neq y_t)$, $\mathbb{I}$ is the indicator function, $\alpha > 0$ and $\mathbf{X}_t = \mathbf{x}_t(\mathbf{x}_t^1 - \mathbf{x}_t^2)^\top$.

Lemma 1 can be proved using similar techniques in [3].Given Lemma 1, Theorem 1 can be proven as follows:

PROOF. According to Lemma 1, we have

$$\begin{aligned}
&\alpha^2\|\mathbf{M}\|_F^2 + \sum_{t=1}^{T} 2\alpha\tau_t\ell_t(\mathbf{M}_t)\\
&\geq \sum_{t=1}^{T} 2\tau_t Z_t[l_t(\alpha - |p_t|) + m_t(\alpha + |p_t|)] - \sum_{t=1}^{T}\tau_t^2\|\mathbf{X}_t\|_F^2\\
&= \sum_{t=1}^{T} 2\tau_t Z_t[l_t(\alpha - |p_t| - \frac{\tau_t}{2}\|\mathbf{X}_t\|_F^2) + m_t(\alpha + |p_t| - \frac{\tau_t}{2}\|\mathbf{X}_t\|_F^2)]\\
&\geq \sum_{t=1}^{T} 2\tau_t Z_t[l_t(\alpha - |p_t| - \frac{\ell_t(\mathbf{M}_t)}{2}) + m_t(\alpha + |p_t| - \frac{\ell_t(\mathbf{M}_t)}{2})]\\
&= \sum_{t=1}^{T} l_t Z_t 2\tau_t(\alpha - \frac{1 + |p_t|}{2}) + \sum_{t=1}^{T} m_t Z_t 2\tau_t(\alpha - \frac{1 - |p_t|}{2}).
\end{aligned}$$

Plugging $\alpha = \frac{\delta+1}{2}$, $\delta \geq 1$ into the above inequality will result in

$$\begin{aligned}
&(\frac{\delta+1}{2})^2\|\mathbf{M}\|_F^2 + \sum_{t=1}^{T}(\delta+1)\tau_t\ell_t(M)\\
&\geq \sum_{t=1}^{T} m_t Z_t\tau_t(\delta + |p_t|).
\end{aligned}$$

Since $\tau_t \geq \min(C, 1/D_X^2)$, the above inequality implies:

$$\begin{aligned}
&(\frac{\delta+1}{2})^2\|\mathbf{M}\|_F^2 + \sum_{t=1}^{T}(\delta+1)\tau_t\ell_t(M)\\
&\geq \min(C, 1/D_X^2)\sum_{t=1}^{T} m_t Z_t(\delta + |p_t|).
\end{aligned}$$

Taking expectation with the above inequality, plugging the equality $\mathbb{E}Z_t = \delta/(\delta + |p_t|)$ and re-arranging the result conclude this theorem. □

Theoretical Analysis on CWAS

In this subsection we will abuse $\mathbf{M}_t$, G_t, $\mathbf{X}_t$ to denote $\mathrm{vec}(M_t)$, $\mathrm{vec}(\mathbf{G}_t)$, and $\mathrm{vec}(X_t)$, respectively. To prove Theorem 2, we need the following lemma.

LEMMA 2. *Let $(\mathbf{x}_t, \mathbf{x}_t^1, \mathbf{x}_t^2; y_t)$, $t \in [T]$ be a sequence of input triplets, where $\mathbf{x}_t, \mathbf{x}_t^1, \mathbf{x}_t^2 \in \mathbb{R}^d$ and $y_t \in \{-1, +1\}$ for all t. If the CWAS is run on this sequence of triplets, then the following bound holds for any $\mathbf{M} \in \mathbb{R}^{d\times d}$,*

$$\begin{aligned}
&Z_t[m_t(\delta + |p_t|) + l_t(\delta - |p_t|)]\\
&\leq \frac{Z_t}{2\eta}\Big[\|\mathbf{M}_t - \delta\mathbf{M}\|_{\mathbf{\Sigma}_{t+1}^{-1}}^2 - \|\mathbf{M}_{t+1} - \delta\mathbf{M}\|_{\mathbf{\Sigma}_{t+1}^{-1}}^2\\
&\quad + \|\mathbf{M}_t - \mathbf{M}_{t+1}\|_{\mathbf{\Sigma}_{t+1}^{-1}}^2\Big] + \delta Z_t\ell_t(\mathbf{M}_t),
\end{aligned}$$

where $l_t = \mathbb{I}(\ell_t(\mathbf{M}_t) > 0 \text{ and } \mathrm{sign}(\mathrm{p_t}) = \mathrm{y_t})$, $m_t = \mathbb{I}(\mathrm{sign}(p_t) \neq y_t)$, $\mathbb{I}$ is the indicator function, $\delta > 0$ and $\|\mathbf{M}_t - \delta\mathbf{M}\|_{\mathbf{\Sigma}_{t+1}^{-1}}^2$ actually denotes $\|vec(\mathbf{M}_t) - vec(\delta\mathbf{M})\|_{\mathbf{\Sigma}_{t+1}^{-1}}^2$.

PROOF. When $Z_t = 0$, it is easy to verify the inequality in the theorem.

When $Z_t = 1$, it is easy to observe that

$$\mathbf{M}_{t+1} = \arg\min_{\mathbf{M}} f_t(\mathbf{M}),$$

where

$$f_t(\mathbf{M}) = \frac{1}{2}\|\mathbf{M}_t - \mathbf{M}\|_{\mathbf{\Sigma}_{t+1}^{-1}}^2 + \eta\mathbf{G}_t^\top\mathbf{M}.$$

Because f_t is convex, we have the following inequality $\forall \mathbf{M}$,

$$\begin{aligned} 0 &\leq \partial f_t(\mathbf{M}_{t+1})^\top(\mathbf{M}-\mathbf{M}_{t+1}) \\ &= [\boldsymbol{\Sigma}_{t+1}^{-1}(\mathbf{M}_{t+1}-\mathbf{M}_t)+\eta\mathbf{G}_t]^\top(\mathbf{M}-\mathbf{M}_{t+1}). \end{aligned}$$

Re-arranging the above inequality will result in

$$\begin{aligned} &\eta\mathbf{G}_t^\top(\mathbf{M}_{t+1}-\mathbf{M}) \\ &\leq (\mathbf{M}_{t+1}-\mathbf{M}_t)^\top\boldsymbol{\Sigma}_{t+1}^{-1}(\mathbf{M}-\mathbf{M}_{t+1}) \\ &= \frac{1}{2}[\|\mathbf{M}_t-\mathbf{M}\|^2_{\boldsymbol{\Sigma}_{t+1}^{-1}} - \|\mathbf{M}_{t+1}-\mathbf{M}\|^2_{\boldsymbol{\Sigma}_{t+1}^{-1}} \\ &\quad -\|\mathbf{M}_t-\mathbf{M}_{t+1}\|^2_{\boldsymbol{\Sigma}_{t+1}^{-1}}]. \end{aligned}$$

Now, we would provide a lower bound for $\mathbf{G}_t^\top(\mathbf{M}_{t+1}-\mathbf{M})$,

$$\begin{aligned} &\mathbf{G}_t^\top(\mathbf{M}_{t+1}-\mathbf{M}) - \mathbf{G}_t^\top(\mathbf{M}_t-\mathbf{M}) + \mathbf{G}_t^\top(\mathbf{M}_{t+1}-\mathbf{M}_t) \\ &= (l_t+m_t)(-y_tX_t^\top\mathbf{M}_t) + (l_t+m_t)y_tX_t^\top\mathbf{M} \\ &-\frac{1}{\eta}\|\mathbf{M}_t-\mathbf{M}_{t+1}\|^2_{\boldsymbol{\Sigma}_{t+1}^{-1}}, \end{aligned}$$

where the second inequality used the facts $\mathbf{G}_t = (l_t + m_t)(-y_t\mathbf{X}_t)$ and $\partial f_t(\mathbf{M}_{t+1}) = 0$, i.e.,

$$\boldsymbol{\Sigma}_{t+1}^{-1}(\boldsymbol{\mu}_{t+1}-\mu_t)+\eta\mathbf{G}_t = 0.$$

Combining the above equality with the facts

$$m_t(-y_t\mathbf{X}_t^\top\mathbf{M}_t) = M_t|p_t|, l_t(-y_t\mathbf{X}_t^\top\mathbf{M}_t) = -l_t|p_t|$$

and

$$y_t\mathbf{X}_t^\top\mathbf{M}_t + \delta\ell_t(\mathbf{M}/\delta) \geq y_tp_t + \delta(1-y_tp_t/\delta) = \delta,$$

we get the following bound for $\mathbf{G}_t^\top(\mathbf{M}_{t+1}-\mathbf{M})$,

$$\begin{aligned} &\mathbf{G}_t^\top(\mathbf{M}_{t+1}-\mathbf{M}) \\ &\geq (m_t|p_t| - l_t|p_t|) + (l_t+m_t)[\delta-\delta\ell_t(\mathbf{M}/\delta)] \\ &\quad -\frac{1}{\eta}\|\mathbf{M}_t-\mathbf{M}_{t+1}\|^2_{\boldsymbol{\Sigma}_{t+1}^{-1}} \\ &= [m_t(\delta+|p_t|)+l_t(\delta-|p_t|)] - (l_t+m_t)\delta\ell_t(\mathbf{M}/\delta) \\ &-\frac{1}{\eta}\|\mathbf{M}_t-\mathbf{M}_{t+1}\|^2_{\boldsymbol{\Sigma}_{t+1}^{-1}}. \end{aligned}$$

Combining the previous inequalities, will give the following important inequality

$$\begin{aligned} &[m_t(\delta+|p_t|)+l_t(\delta-|p_t|)] \\ &\leq \frac{1}{2\eta}[\|\mathbf{M}_t-\delta\mathbf{M}\|^2_{\boldsymbol{\Sigma}_{t+1}^{-1}} - \|\mathbf{M}_{t+1}-\delta\mathbf{M}\|^2_{\boldsymbol{\Sigma}_{t+1}^{-1}} \\ &\quad -\|\mathbf{M}_t-\mathbf{M}_{t+1}\|^2_{\boldsymbol{\Sigma}_{t+1}^{-1}}] + \frac{1}{\eta}\|\mathbf{M}_t-\mathbf{M}_{t+1}\|^2_{\boldsymbol{\Sigma}_{t+1}^{-1}} + \delta\ell_t(\mathbf{M}/\delta) \\ &= \frac{1}{2\eta}[\|\mathbf{M}_t-\delta\mathbf{M}\|^2_{\boldsymbol{\Sigma}_{t+1}^{-1}} - \|\mathbf{M}_{t+1}-\delta\mathbf{M}\|^2_{\boldsymbol{\Sigma}_{t+1}^{-1}} \\ &\quad +\|\mathbf{M}_t-\mathbf{M}_{t+1}\|^2_{\boldsymbol{\Sigma}_{t+1}^{-1}}] + \delta\ell_t(\mathbf{M}/\delta). \end{aligned}$$

Replacing $\mathbf{M}$ with $\delta\mathbf{M}$ concludes the proof. □

Given Lemma 2, the theorem 2 can be proven as follows:

PROOF. Firstly, according to the update rule

$$\mathbf{M}_{t+1} = \text{mat}[\text{vec}(\mathbf{M}_t) - \eta\boldsymbol{\Sigma}_t\text{vec}(G_t)],$$

we can derive the following equality

$$\begin{aligned} &\|\mathbf{M}_t-\mathbf{M}_{t+1}\|^2_{\boldsymbol{\Sigma}_{t+1}^{-1}} = \eta^2\mathbf{G}_t^\top\boldsymbol{\Sigma}_{t+1}\mathbf{G}_t \\ &= \eta^2(m_t+l_t)\mathbf{X}_t^\top\boldsymbol{\Sigma}_{t+1}\mathbf{X}_t = \eta^2\gamma(1-\frac{|\boldsymbol{\Sigma}_t^{-1}|}{|\boldsymbol{\Sigma}_{t+1}^{-1}|}), \end{aligned}$$

where, we used the fact $A = B+\mathbf{x}\mathbf{x}^\top$ implies $\mathbf{x}^\top A^{-1}\mathbf{x} = 1-\frac{|B|}{|A|}$. Plugging the above equality into the inequality in the Lemma 2, and re-arranging it will gives

$$\begin{aligned} &Z_t[m_t(\delta+|p_t|)+l_t(\delta-|p_t|)] \\ &\leq \frac{Z_t}{2\eta}\Big[\|\mathbf{M}_t-\delta\mathbf{M}\|^2_{\boldsymbol{\Sigma}_{t+1}^{-1}} - \|\mathbf{M}_{t+1}-\delta\mathbf{M}\|^2_{\boldsymbol{\Sigma}_{t+1}^{-1}}\Big] \\ &\quad +\frac{Z_t\eta\gamma}{2}(1-\frac{|\boldsymbol{\Sigma}_t^{-1}|}{|\boldsymbol{\Sigma}_{t+1}^{-1}|}) + \delta Z_t\ell_t(\mathbf{M}_t). \end{aligned}$$

Summing the above inequality over $t = 1, 2, \ldots, T$ can give

$$\begin{aligned} \sum_{t=1}^T Z_t[m_t(\delta+|p_t|)+l_t(\delta-|p_t|)] &\leq \sum_{t=1}^T \delta Z_t\ell_t(\mathbf{M}_t) \\ &+\sum_{t=1}^T \frac{Z_t}{2\eta}\Big[\|\mathbf{M}_t-\delta\mathbf{M}\|^2_{\boldsymbol{\Sigma}_{t+1}^{-1}} - \|\mathbf{M}_{t+1}-\delta\mathbf{M}\|^2_{\boldsymbol{\Sigma}_{t+1}^{-1}}\Big] \\ &+\sum_{t=1}^T \frac{Z_t\eta\gamma}{2}(1-\frac{|\boldsymbol{\Sigma}_t^{-1}|}{|\boldsymbol{\Sigma}_{t+1}^{-1}|}). \end{aligned}$$

Now, we would like to bound the right hand side of the above inequality. Firstly, we bound the first term as

$$\begin{aligned} &\sum_{t=1}^T Z_t\Big[\|\mathbf{M}_t-\delta\mathbf{M}\|^2_{\boldsymbol{\Sigma}_{t+1}^{-1}} - \|\mathbf{M}_{t+1}-\delta\mathbf{M}\|^2_{\boldsymbol{\Sigma}_{t+1}^{-1}}\Big] \\ &\leq \|\mathbf{M}_1-\delta\mathbf{M}\|^2_{\boldsymbol{\Sigma}_2^{-1}} + \sum_{t=2}^T[\|\mathbf{M}_t-\delta\mathbf{M}\|^2_{\boldsymbol{\Sigma}_{t+1}^{-1}} - \|\mathbf{M}_t-\delta\mathbf{M}\|^2_{\boldsymbol{\Sigma}_t^{-1}}] \\ &\leq \|\mathbf{M}_1-\delta\mathbf{M}\|^2\text{Tr}(\boldsymbol{\Sigma}_2^{-1}) + \sum_{t=2}^T\|\mathbf{M}_t-\delta\mathbf{M}\|^2\text{Tr}(\boldsymbol{\Sigma}_{t+1}^{-1}-\boldsymbol{\Sigma}_t^{-1}) \\ &= \max_{t\leq T}\|\mathbf{M}_t-\delta\mathbf{M}\|^2\text{Tr}(\boldsymbol{\Sigma}_{T+1}^{-1}) \\ &\leq 2(D_M+|1-\delta|\|\mathbf{M}\|)^2\text{Tr}(\boldsymbol{\Sigma}_{T+1}^{-1}), \end{aligned}$$

where $D_M = \max_{t\leq T}\|\mathbf{M}_t-\mathbf{M}\|^2$. Combining the above two inequalities, result in

$$\begin{aligned} &\sum_{t=1}^T Z_t[m_t(\delta+|p_t|)] \leq \delta\sum_{t=1}^T Z_t\ell_t(\mathbf{M}) \\ &+\frac{1}{\eta}(D_M+(1-\delta)^2\|\mathbf{M}\|^2)\text{Tr}(\boldsymbol{\Sigma}_{T+1}^{-1}) + \sum_{t=1}^T\frac{Z_t\eta\gamma}{2}(1-\frac{|\boldsymbol{\Sigma}_t^{-1}|}{|\boldsymbol{\Sigma}_{t+1}^{-1}|}) \\ &\leq \delta\sum_{t=1}^T Z_t\ell_t(\mathbf{M}) + \frac{1}{\eta}(D_M+|1-\delta|\|\mathbf{M}\|)^2\text{Tr}(\boldsymbol{\Sigma}_{T+1}^{-1}) \\ &+\frac{\eta\gamma}{2}\ln(|\boldsymbol{\Sigma}_{T+1}^{-1}|), \end{aligned}$$

where we used the fact $(1-\frac{|A|}{|B|}) \leq \ln\frac{|B|}{|A|}$, since $1-x \leq -\ln x$ for all $x > 0$.

Taking expectation with the above inequality and using the fact $\mathbb{E}[Z_t] = \frac{\delta}{\delta+|p_t|}$ gives

$$\begin{aligned} &\delta\mathbb{E}\sum_{t=1}^T[m_t] \leq \delta\sum_{t=1}^T Z_t\ell_t(\mathbf{M}) \\ &+\frac{1}{\eta}(D_M+|1-\delta|\|\mathbf{M}\|)^2\text{Tr}(\boldsymbol{\Sigma}_{T+1}^{-1}) + \frac{\eta\gamma}{2}\ln(|\boldsymbol{\Sigma}_{T+1}^{-1}|). \end{aligned}$$

Dividing the above inequality with δ concludes the proof. □

7. REFERENCES

[1] G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *Knowledge and Data Engineering, IEEE Transactions on*, 17(6):734–749, 2005.

[2] Q. Cao, Y. Ying, and P. Li. Similarity metric learning for face recognition. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 2408–2415. IEEE, 2013.

[3] N. Cesa-Bianchi, C. Gentile, and L. Zaniboni. Worst-case analysis of selective sampling for linear classification. *The Journal of Machine Learning Research*, 7:1205–1230, Dec. 2006.

[4] G. Chechik, V. Sharma, U. Shalit, and S. Bengio. Large scale online learning of image similarity through ranking. *Journal of Machine Learning Research*, 11:1109–1135, 2010.

[5] K. Crammer and G. Chechik. Adaptive regularization for similarity measures. In *Proceedings of the 29th International Conference on Machine Learning, ICML, Edinburgh, Scotland, UK, June 26 - July 1*, 2012.

[6] K. Crammer, O. Dekel, J. Keshet, and S. Shalev-shwartz. Online passive-aggressive algorithms. *The Journal of Machine ...*, 7:551–585, 2006.

[7] K. Crammer, A. Kulesza, and M. Dredze. Adaptive regularization of weight vectors. In *Advances in Neural Information Processing Systems*, pages 414–422, 2009.

[8] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon. Information-theoretic metric learning. In *Proceedings of the 24th international conference on Machine learning*, pages 209–216. ACM, 2007.

[9] W. B. Frakes and R. Baeza-Yates, editors. *Information Retrieval: Data Structures and Algorithms*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1992.

[10] Y. Freund and Y. Mansour. Learning under persistent drift. In *Computational Learning Theory*, pages 109–118, 1997.

[11] A. Globerson and S. T. Roweis. Metric learning by collapsing classes. In *Advances in neural information processing systems*, pages 451–458, 2005.

[12] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. 2007.

[13] Y. Guo and D. Schuurmans. Discriminative batch mode active learning. In *Advances in neural information processing systems*, pages 593–600, 2008.

[14] S. C. Hoi, W. Liu, M. R. Lyu, and W.-Y. Ma. Learning distance metrics with contextual constraints for image retrieval. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR2006)*, New York, US, 2006.

[15] S. C. Hoi, J. Wang, and P. Zhao. Libol: A library for online learning algorithms. *The Journal of Machine Learning Research*, 15(1):495–499, 2014.

[16] R. Jin, S. Wang, and Y. Zhou. Regularized distance metric learning: Theory and algorithm. In *Advances in neural information processing systems*, pages 862–870, 2009.

[17] D. D. Lewis. Learning in intelligent information retrieval. In *Machine Learning: Proceedings of the Eighth International Workshop*, pages 235–239, 2014.

[18] D. D. Lewis and J. Catlett. Heterogenous uncertainty sampling for supervised learning. In *ICML*, volume 94, pages 148–156, 1994.

[19] J. Lu, P. Zhao, and S. C. Hoi. Online passive aggressive active learning and its applications. In *Journal of Machine Learning Research - Proceedings Track (ACML2014)*, 2014.

[20] N. Roy, A. Mccallum, and M. W. Com. Toward optimal active learning through sampling estimation of error reduction. In *ICML*, Williamstown, 2001.

[21] Y. Rui, T. S. Huang, and S.-F. Chang. Image retrieval: Current techniques, promising directions, and open issues. *Journal of visual communication and image representation*, 10(1):39–62, 1999.

[22] M. Schultz and T. Joachims. Learning a distance metric from relative comparisons. *Advances in neural information processing systems (NIPS)*, page 41, 2004.

[23] B. Settles. Active learning literature survey. *University of Wisconsin, Madison*, 52:55–66, 2010.

[24] S. Shalev-Shwartz, Y. Singer, and A. Y. Ng. Online and batch learning of pseudo-metrics. In *Proceedings of the twenty-first international conference on Machine learning*, page 94. ACM, 2004.

[25] V. S. Sheng, F. Provost, and P. G. Ipeirotis. Get another label? improving data quality and data mining using multiple, noisy labelers. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 614–622. ACM, 2008.

[26] L. Si, R. Jin, S. C. Hoi, and M. R. Lyu. Collaborative image retrieval via regularized metric learning. *ACM Multimedia Systems Journal (MMSJ)*, 12(1):34–44, 2006.

[27] S. Tong and D. Koller. Support vector machine active learning with applications to text classification. *The Journal of Machine Learning Research*, pages 45–66, 2002.

[28] K. Q. Weinberger, J. Blitzer, and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. In *Advances in neural information processing systems*, pages 1473–1480, 2005.

[29] L. Wu, R. Jin, S. C. Hoi, J. Zhu, and N. Yu. Learning bregman distance functions and its application for semi-supervised clustering. In Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems 22*, pages 2089–2097, 2009.

[30] W. Wu, H. Li, and J. Xu. Learning query and document similarities from click-through bipartite graph with metadata. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 687–696. ACM, 2013.

[31] L. Yang and R. Jin. Distance metric learning: A comprehensive survey. *Michigan State Universiy*, 2, 2006.

[32] P. Zhao and S. C. Hoi. Cost-sensitive online active learning with application to malicious url detection. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '13*, page 919, New York, New York, USA, 2013. ACM Press.

[33] P. Zhao, S. C. Hoi, and R. Jin. Double updating online learning. *The Journal of Machine Learning Research*, 12:1587–1615, 2011.

[34] P. Zhao, S. C. Hoi, and J. Zhuang. Active learning with expert advice. In *UAI2013*. AUAI Press, Corvallis, Oregon, 2013.

[35] P. Zhao, R. Jin, T. Yang, and S. C. Hoi. Online auc maximization. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 233–240, 2011.