Online Metric Learning for Multi-Label Classification

Xiuwen Gong, Dong Yuan, Wei Bao
Faculty of Engineering, The University of Sydney
{xiuwen.gong, dong.yuan, wei.bao}@sydney.edu.au

Abstract

Existing research into online multi-label classification, such as online sequential multi-label extreme learning machine (OSML-ELM) and stochastic gradient descent (SGD), has achieved promising performance. However, these works lack an analysis of loss function and do not consider label dependency. Accordingly, to fill the current research gap, we propose a novel online metric learning paradigm for multi-label classification. More specifically, we first project instances and labels into a lower dimension for comparison, then leverage the large margin principle to learn a metric with an efficient optimization algorithm. Moreover, we provide theoretical analysis on the upper bound of the cumulative loss for our method. Comprehensive experiments on a number of benchmark multi-label datasets validate our theoretical approach and illustrate that our proposed online metric learning (OML) algorithm outperforms state-of-the-art methods.

Introduction

Real-world applications often involve a large number of classes, each instance of which can be assigned multiple labels. For example, many web-related applications, such as Twitter, Facebook and Instagram posts and RSS feeds, are attached with multiple essential forms of categorization tags (Zhang, Graepel, and Herbrich 2012). In the search industry, revenue comes from clicks on ads embedded in the result pages. Ad selection and placement can be significantly improved if ads are tagged correctly. This scenario, referred to as 'online multi-label classification' in a machine learning context, is also useful in some other applications, such as object detection in video surveillance (Popovici, Weiler, and Grossniklaus 2014) and image retrieval in dynamic databases (Dong and Bhanu 2003).

In the development of multi-label classification (Tsoumakas, Zhang, and Zhou 2012; Gibaja and Ventura 2015), one challenge that remains unsolved is that most multi-label classification algorithms are developed in an off-line mode (Cheng and Hüllermeier 2009; Chen and Lin 2012; Babbar and Schölkopf 2017; Liu and Tsang 2017; Zhou et al. 2019b; Liu et al. 2019). These methods assume that all data are available in advance for learning. However,there are two major limitations of developing multi-label methods under such an assumption: firstly, these methods are impractical for large-scale datasets, since they require all datasets to be stored in memory; secondly, it is non-trivial to adapt off-line multi-label methods to the sequential data. In practice, data is collected sequentially, and data that is collected earlier in this process may expire as time passes. Therefore, it is important to develop new multi-label classification methods to deal with streaming data.

Several online multi-label classification studies have recently been developed to overcome the above-mentioned limitations. For example, online learning with accelerated nonsmooth stochastic gradient (OLANSGD) (Park and Choi 2013) was proposed to solve the online multi-label classification problem. Moreover, the online sequential multi-label extreme learning machine (OSML-ELM) (Venkatesan et al. 2017) is a single-hidden layer feed-forward neural network-based learning technique. OSML-ELM classifies the examples by their output weight and activation function. Unfortunately, all of these online multi-label classification methods lack an analysis of loss function and disregard label dependencies. Many studies (Dembczynski, Cheng, and Hüllermeier 2010; Read et al. 2011; Bhatia et al. 2015; Yen et al. 2016; Liu, Tsang, and Müller 2017) have shown that multi-label learning methods that do not capture label dependency usually achieve degraded prediction performance. This paper aims to fill these gaps.

k-nearest neighbour (kNN) algorithms have achieved superior performance in various applications (Deng et al. 2010). Moreover, experiments show that distance metric learning on single-label prediction can improve the prediction performance of kNN. Nevertheless, there are two problems associated with applying a kNN algorithm to an online multi-label setting. Firstly, naive kNN algorithms do not consider label dependencies. Secondly, it is non-trivial to learn an appropriate metric for online multi-label classification.

To break the bottleneck of kNN, we here propose a novel multi-label learning paradigm for multi-label classification.

More specifically, we project instances and labels into the same embedding space for comparison, after which we learn the distance metric by enforcing the constraint that the distance between embedded instance and its correct label must be smaller than the distance between the embedded instance and other labels. Thus, two nearby instances from different labels will be pushed further. Moreover, an efficient optimization algorithm is proposed for the online multi-label scenario. In theoretical terms, we analyze the upper bound of cumulative loss for our proposed model. A wide range of experiments on benchmark datasets corroborate our theoretical results and verify the improved accuracy of our method relative to state-of-the-art approaches.

The remainder of this paper is organized as follows. We first describe the related work, the online metric learning for multi-label classification and the optimization algorithm. Next, we introduce the upper bound of the loss function. Finally, we present the experimental results and conclude this paper.

Related Work

Existing multi-label classification methods can be grouped into two major categories: namely, *algorithm adaptation* (AA) and *problem transformation* (PT). AA extends specific learning algorithms to deal with multi-label classification problems. Typical AA methods include (Zhang and Zhou 2006; Brinker and Hullermeier 2007; Zhou et al. 2019a). Moreover, PT methods such as that developed by (Hsu et al. 2009a), transform the learning task into one or more single-label classification problems. However, all of these methods assume that all data are available for learning in advance. These methods thus incur prohibitive computational costs on large-scale datasets, and it is also non-trivial to apply them to sequential data.

The state-of-the-art approaches to online multi-label classification have been developed to handle sequential data. These approaches can be divided into two key categories: *Neural Network* and *Label Ranking*. Neural Network approaches are based on a collection of connected units or nodes, referred to as artificial neurons. Each connection between artificial neurons can transmit the signal from one neuron to another. The artificial neuron that receives the signal can process it and then transmit signal to other artificial neurons. Moreover, label ranking, another popular approach to multi-label learning, involves a set of ranking functions being learned to order all the labels such that relevant labels are ranked higher than irrelevant ones.

From the neural network perspective, Ding et al. (Ding et al. 2015) developed a single-hidden layer feedforward neural network-based learning technique named ELM. In this method, the initial weights and the hidden layer bias are selected at random, and the network is trained for the output weights to perform the classification. Moreover, Venkatesan et al. (Venkatesan et al. 2017) developed the OSML-ELM approach, which uses ELM to handle streaming data. OSML-ELM uses a sigmoid activation function and outputs weights to predict the labels. In each step, the output weight is learned from the specific equation. OSML-ELM converts the label set from bipolar to unipolar representation in order to solve multi-label classification problems.

Some other existing approaches are based on label ranking, such as OLANSGD (Park and Choi 2013). In the majority of cases, ranking functions are learned by minimizing the ranking loss in the max margin framework. However, the memory and computational costs of this process are expensive on large-scale datasets. Stochastic gradient decent (SGD) approaches update the model parameters using only the gradient information calculated from a single label at each iteration. OLANSGD minimizes the primal form using Nesterov's smoothing, which has recently been extended to the stochastic setting.

However, none of these methods analyze the loss function, and all of them fail to capture the interdependencies among labels; these issues have been proved to result in degraded prediction performance. Accordingly, this paper aims to address these issues.

Notation	Definition
t	the round of algorithm
x_t	an instance presented on round t
y_t	corresponding label vector to x_t
x	nearest neighbour instance to x_t
y	corresponding output of x
X	initialized input matrix
Y	corresponding output matrix
S	number of initialized instances
V_t, P_t	projection matrix on round t
m, M	lower bound and upper bound of λ_t
$\langle A, B\rangle_F$	Frobenius inner product of A and B
$\|\|\cdot\|\|_1$	l_1 norm
$\|\|\cdot\|\|_2$	l_2 norm
$\|\|\cdot\|\|_F$	Frobenius norm

Table 1: Summary of Notations

Our Proposed Method

Notations

We denote the instance presented to the algorithm on round t by $x_t \in \mathbb{R}^{p\times 1}$, and the label by $y_t \in \{0,1\}^{q\times 1}$, and refer each instance-label pair as an example. Suppose that we initially have S examples in memory, denoted by $D = \{(x_i, y_i)\}_{i=1}^{S}$. $(x, y) \in D$ is a nearest neighbour to x_t. The initialized instance matrix is denoted as $X \in \mathbb{R}^{S\times p}$ and the correspond output matrix is denoted as $Y \in \{0,1\}^{S\times q}$. t is a positive integer. $||\cdot||_F$ is Frobenius norm. $V_t = (v_1, v_2, ..., v_d) \in \mathbb{R}^{q\times d}(d < q)$ is projection matrix which maps each output vector y_t (q dimension) to $V^T y_t$ (d dimension). Let $P \in \mathbb{R}^{p\times q}$ also be the projection matrix. Each input vector x_t (p dimension) is projected to $V^T P^T x_t$ (d dimension). Then x_t and y_t can be compared in the projection space(d dimension). Notations are summarized in Table 1.

Online Metric Learning

Inspired by Hsu et al. (Hsu et al. 2009b),who stated that each label vector can be projected into a lower dimensional

label space, which is deemed as encoding, we propose the following large-margin metric learning approach with nearest neighbor constraints to learn projection. If the encoding scheme works well, the distance between the codeword of x_t, $(V^T P^T x_t)$, and y_t, $(V^T y_t)$, should tend to be 0 and less than the distance between codeword x_t and any other output $V^T y$. The following large margin formulation is then presented to learn the projection matrix V:

$$\begin{aligned} &argmin_{V\in\mathbb{R}^{q\times d}} \frac{1}{2}||V||_F^2 + \xi_t \\ &s.t. \quad ||V^T P^T x_t - V^T y_t||_2^2 + \Delta(y_t, y) - \xi_t \\ &\quad \leq ||V^T P^T x_t - V^T y||_2^2, \forall t \in \{1, 2, \cdots\} \end{aligned} \tag{1}$$

The constraints in Eq.(1) guarantee that the distance between the codeword of x_t and the codeword of y_t is less than the distance between the codeword of x_t and codeword of any other output. To give Eq.(1) more robustness, we add loss function $\Delta(y_t, y)$ as the margin. The loss function is defined as $\Delta(y_t, y) = ||y_t - y||_1$, where $||\cdot||_1$ is the l_1 norm. After that, we use Euclidean metric to measure the distances between instances x_t and x and then learn a new distance metric, which improves the performance of kNN and also captures label dependency.

To retain the information learned on the round t, we apply above large margin formulation into online setting. Thus, we have to define the initialization of the projection matrix and the updating rule. We initialize the projection matrix V_1 to a non-zero matrix and set the new projection matrix V_{t+1} to be the solution of the following constrained optimization problem on round t.

$$\begin{aligned} &V_{t+1}^T = argmin_{V\in\mathbb{R}^{q\times d}} \frac{1}{2}||V^T - V_t^T||_F^2 \\ &\quad s.t. \quad l(V; (x_t, y_t)) = 0 \end{aligned} \tag{2}$$

The loss function is defined as following:

$$\begin{aligned} l(V; (x_t, y_t)) = \max\{0, \Delta(y_t, y) - (||V^T P^T x_t - V^T y||_2^2 \\ - ||V^T P^T x_t - V^T y_t||_2^2)\} \end{aligned} \tag{3}$$

where the matrix P is learned through the following formulation:

$$argmin_{P\in\mathbb{R}^{p\times q}} \frac{1}{2}||P^T X^T - Y^T||_F^2$$

Define the loss function on round t as

$$\begin{aligned} l_t(V_t; (x_t, y_t)) = \max\{0, \Delta(y_t, y) - (||V_t^T P^T x_t - V_t^T y||_2^2 \\ - ||V_t^T P^T x_t - V_t^T y_t||_2^2)\} \end{aligned} \tag{4}$$

When loss function is zero on round t, $V_{t+1} = V_t$. In contrast, on those rounds where the loss function is positive, the algorithm enforces V_{t+1} to satisfy the constraint $l_{t+1}(V_{t+1}; (x_{t+1}, y_{t+1})) = 0$ regardless of the step-size required. This update rule requires V_{t+1} to correctly classify the current example with a sufficient high margin and V_{t+1} have to stay as closed as V_t to retain the information learned on the previous round.

Optimization

The optimization of Eq.(2) can be shown by using standard tools from convex optimization (Boyd and Vandenberghe 2004). If $l_t = 0$ then V_t itself satisfies the constraint in Eq.(2) and is clearly the optimal solution. Therefore, we concentrate on the case where $l_t > 0$. Firstly, we define the Lagrangian of the optimization problem in Eq.(2) to be,

$$\begin{aligned} L = &\frac{1}{2}||V^T - V_t^T||_F^2 + \lambda(\Delta(y_t, y) \\ &-(||V^T P^T x_t - V^T y||_2^2 - ||V^T P^T x_t - V^T y_t||_2^2)) \end{aligned} \tag{5}$$

where the λ is a Lagrange multiplier.

Setting the partial derivatives of L with respect to the elements of V^T to zero gives

$$\begin{aligned} 0 = \frac{\partial L}{\partial V^T} = &V^T - V_t^T - 2V^T\lambda((P^T x_t - y)(P^T x_t - y)^T \\ &- (P^T x_t - y_t)(P^T x_t - y_t)(P^T x_t - y_t)^T) \end{aligned}$$

from this equation, we can get that

$$\begin{aligned} V^T = &V_t^T(I - 2\lambda((P^T x_t - y)(P^T x_t - y)^T \\ &- (P^T x_t - y_t)(P^T x_t - y_t)^T))^{-1} \end{aligned}$$

in which I stands for an identity matrix.

Inspired by (Petersen and Pedersen 2012), we use an approximation form of V^T to make it easier for following calculation.

$$\begin{aligned} \bar{V}^T = &V_t^T(I + 2\lambda((P^T x_t - y)(P^T x_t - y)^T \\ &- (P^T x_t - y_t)(P^T x_t - y_t)^T)) \end{aligned} \tag{6}$$

Define $Q = V_t V_t^T$, $A = (P^T x_t - y)(P^T x_t - y)^T - (P^T x_t - y_t)(P^T x_t - y_t)^T$. Plugging the approximation formula Eq.(6) back into Eq.(5), we get a cubic function $f(\lambda) = a\lambda^3 + b\lambda^2 + c\lambda$, $\lambda \in \mathbb{R}$, where

$$\begin{aligned} a = &4(P^T x_t - y_t)^T A^T Q A (P^T x_t - y_t) \\ &- (P^T x_t - y)^T A^T Q A (P^T x_t - y) \\ b = &2(||V_t^T A||_F^2 - (P^T x_t - y_t)^T Q A (P^T x_t - y_t) \\ &- (P^T x_t - y_t)^T A^T Q (P^T x_t - y_t) \\ &+ (P^T x_t - y)^T Q A (P^T x_t - y) \\ &+ (P^T x_t - y)^T A^T Q (P^T x_t - y)) \\ c = &(P^T x_t - y)^T Q (P^T x_t - y) - (P^T x_t - y)^T Q (P^T x_t - y) \\ &+ \Delta(y_t, y) \end{aligned}$$

If $f(\lambda)$ is non-monotonic function when $\lambda > 0$, let $\beta > 0$ to be the maximum point of $f(\lambda)$. We obtain,

$$\lambda_t = \begin{cases} m & if \quad f'(\lambda) < 0 \quad and \quad \lambda > 0, \quad \beta < m \\ \beta & if \quad m < \beta < M \\ M & if \quad f'(\lambda) > 0 \quad and \quad \lambda > 0, \quad \beta > M \end{cases} \tag{7}$$

where $m, M \in \mathbb{R}$, $0 < m < M$

Algorithm 1 provides detail of optimization. We denote the loss suffered by our algorithm on round t by l_t.

Algorithm 1 Online Metric Learning for Multi-Label Classification

Set V_1 to a non-zero matrix
Initialize $D = \{(x_i, y_i)\}_{i=1}^{S}$
for $t = 1, 2, \ldots,$ **do**
  Receive pairwise instances: (x_t, y_t)
  Find the Nearest Neighbour $(x, y) \in D$
  Compute loss l_t by Eq.(4)
  if $l_t > 0$ **then**
    Set λ_t as Eq.(7)
    Update $V^T = V_t^T(I - 2\lambda_t A)^{-1}$
  else
    $V_{t+1} = V_t$
  end if
  Append current instances into D
end for

We focus on the situation when $l_t > 0$. The optimal solution comes from the one satisfying $\partial L/\partial V = 0$, $\partial L/\partial \lambda = 0$. Based on the derivation, V_{t+1} can be update by $V_{t+1}^T = V_t^T(I - 2\lambda_t A)^{-1}$, where $A = (P^T x_t - y)(P^T x_t - y)^T - (P^T x_t - y_t)(P^T x_t - y_t)^T$.

Inspired by metric learning (Kulis 2013), we use the learned metric to select k nearest neighbours from D for each testing instance, and conduct the predictions based on these k nearest neighbours. The equation of the distance between codeword x_j and x_t in the embedding space can be computed as $(P^T x_j - P^T x_t)^T Q (P^T x_j - P^T x_t)$.

Loss Bound

Following the analysis in (Crammer et al. 2006), we state the upper bounds for our online metric learning algorithm. Let $U = (u_1, u_2, ..., u_d) \in \mathbb{R}^{q\times d}(d < q)$ be an arbitrary matrix. We use the approximate form given in Eq.(6) to replace V^T.

Lemma 1. *Let λ_t as defined in Eq.(7), $V = (v_1, v_2, ..., v_d) \in \mathbb{R}^{q\times d}(d < q)$, V_1 is a non-zero matrix. The following bound holds for any $U \in \mathbb{R}^{q\times d}(d < q)$*

$$||V_1 - U||_F^2 - ||V_{T+1} - U||_F^2 \le ||V_1 - U||_F^2$$

Proof. Define $\Psi_t = ||V_t - U||_F^2 - ||V_{t+1} - U||_F^2$, this lemma is proved by summing Ψ_t over all t in $1, \ldots, T$ and the bounding of this sum is obviously as followed,

$$\sum_{t=1}^{T} \Psi_t = ||V_1 - U||_F^2 - ||V_{T+1} - U||_F^2 \le ||V_1 - U||_F^2$$

□

Lemma 2. *Assume there exists some U such that $4\lambda_t \langle U, A^T V_t\rangle_F - 4\lambda_t^2 ||A^T V_t|| \ge 5\lambda_t \langle V_t, A^T V_t\rangle_F + \frac{qc^2\lambda_t}{(||P||_F^2 r + q)}$, $\forall t \in \{1, 2, ..., T\}$. Let $V = (v_1, v_2, ..., v_d) \in \mathbb{R}^{q\times d}(d < q)$. λ_t as defined as in Eq.(7). V_1 is a non-zero matrix. c is defined in the proof Eq.(4). We bound cumulative $||V_t||_F^2$ as follows,*

$$\sum_{t=1}^{T} ||V_t||_F^2 \le \frac{||V_1 - U||_F^2}{m \cdot c^2} - \frac{q \cdot T}{(||P||_F^2 r + q)}$$

Proof. By using the operation of Frobenius norm,

$$||A + B||_F^2 = ||A||_F^2 + ||B||_F^2 + 2\langle A, B\rangle_F$$

where $\langle\cdot\rangle_F$ is the Frobenius inner product, we can get

$$\begin{aligned}
\Psi_t &= ||V_t - U||_F^2 - ||V_{t+1} - U||_F^2 \\
&= ||V_t - U||_F^2 - ||(I + 2\lambda_t A)^T V_t - U||_F^2 \\
&= ||V_t - U||_F^2 - ||V_t - U||_F^2 - 4\lambda_t^2 ||A^T V_t||_F^2 \\
&\quad - 4\lambda_t \langle V_t - U, A^T V_t\rangle_F \\
&= -4\lambda_t \langle V_t, A^T V_t\rangle_F + 4\lambda_t \langle U, A^T V_t\rangle_F \\
&\quad - 4\lambda_t^2 ||A^T V_t||_F^2
\end{aligned}$$

Using the assumption in Lemma 2, we can get that $\Psi_t \ge \lambda_t \langle V_t, A^T V_t\rangle_F + \frac{qc^2\lambda_t}{(||P||_F^2 r + q)}$. where $A = (P^T x_t - y)(P^T x_t - y)^T - (P^T x_t - y_t)(P^T x_t - y_t)^T$. It is clearly that A is a symmetric matrix. We take the SVD of A as $A = \bar{U}\bar{A}\bar{U}^T$, then using the minimum non-negative singular value of A to replace the non-positive element in matrix $\bar{A}$, and denote approximation form of matrix A as $\hat{A}$. Apparently, $\hat{A}$ is a non-negative symmetric matrix. Furthermore, by using definition of Frobenius inner product $\langle A, B\rangle_F = Trace(A^T B)$, where $Trace(A) = \sum_{i=1}^{n} a_{ii}$, we can get that

$$\begin{aligned}
\langle V_t, \hat{A}^T V_t\rangle_F &= Trace(V_t^T \hat{A}^T V_t) \\
&= Trace(V_t^T (\hat{A}^{\frac{1}{2}T}) \hat{A}^{\frac{1}{2}} V_t) \\
&= Trace((\hat{A}^{\frac{1}{2}} V_t)^T \hat{A}^{\frac{1}{2}} V_t) \\
&= ||\hat{A}^{\frac{1}{2}} V_t||_F^2
\end{aligned}$$

Taking the SVD of $\hat{A}^{\frac{1}{2}}$ as $\hat{A}^{\frac{1}{2}} = U^* A^* U^{*T}$. Since matrix U^* is a unitary matrix, then $||U^* B||_F^2 = ||B||_F^2$, $\forall B \in \mathbb{R}^{q\times q}$. Let c be the minimum singular value of A^*, getting that

$$\begin{aligned}
||\hat{A}^{\frac{1}{2}} V_t||_F^2 &= ||U^* A^* U^{*T} V_t||_F^2 \\
&= ||A^* U^{*T} V_t||_F^2 \\
&\ge ||cIU^{*T} V_t||_F^2 \\
&\ge c^2 ||V_t||_F^2
\end{aligned} \tag{8}$$

where I is an identity matrix. Now, we get that,

$$\Psi_t \ge c^2 \lambda_t ||V_t||_F^2 + \frac{qc^2\lambda_t}{(||P||_F^2 r + q)}$$

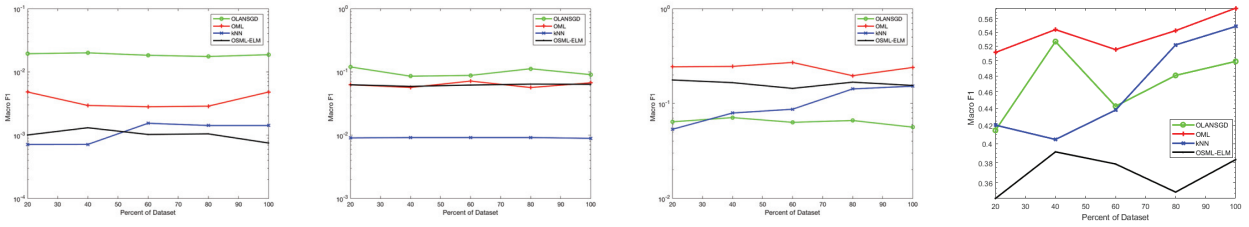
By summing both side of inequality on t over all t in $1, \ldots, T$, and using that $m \le \lambda_t \le M$, gives that

$$\sum_{t=1}^{T} c^2 \cdot m ||V_t||_F^2 + \frac{T \cdot qc^2 m}{(||P||_F^2 r + q)} \le ||V_1 - U||_F^2$$

Then, we can get that

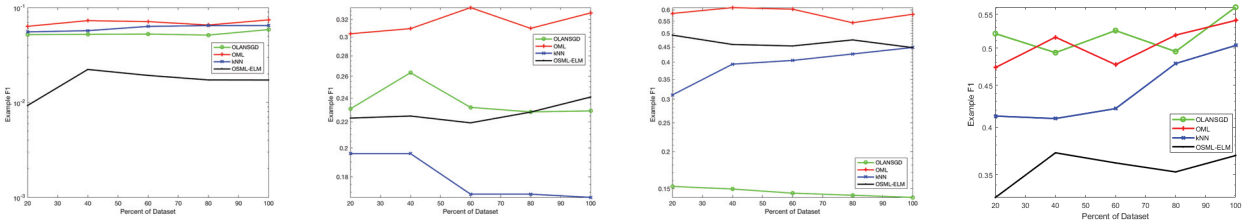
$$\sum_{t=1}^{T} ||V_t||_F^2 \le \frac{||V_1 - U||_F^2}{m \cdot c^2} - \frac{q \cdot T}{(||P||_F^2 r + q)}$$

Lemma 2 has been proved. □



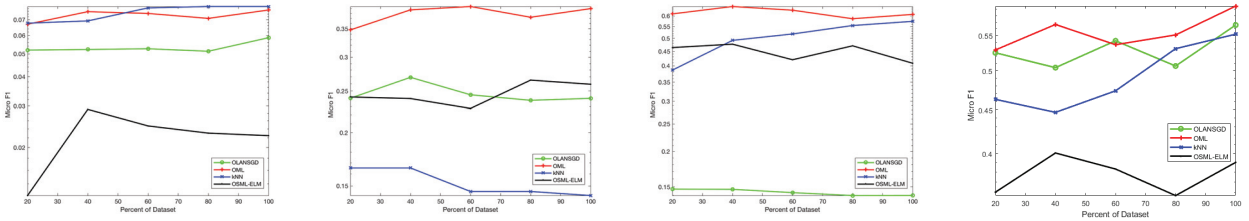
(a) CoreI5k (b) Enron (c) FullMedical (d) Emotions

Figure 1: Macro F1 of various methods on Corel5k, Enron, Medical and Emotions datasets.



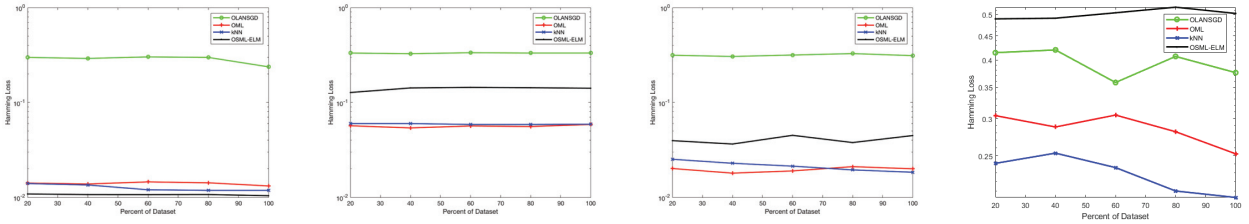
(a) CoreI5k (b) Enron (c) FullMedical (d) Emotions

Figure 2: Example F1 of various methods on Corel5k, Enron, Medical and Emotions datasets.



(a) CoreI5k (b) Enron (c) FullMedical (d) Emotions

Figure 3: Micro F1 of various methods on Corel5k, Enron, Medical and Emotions datasets.



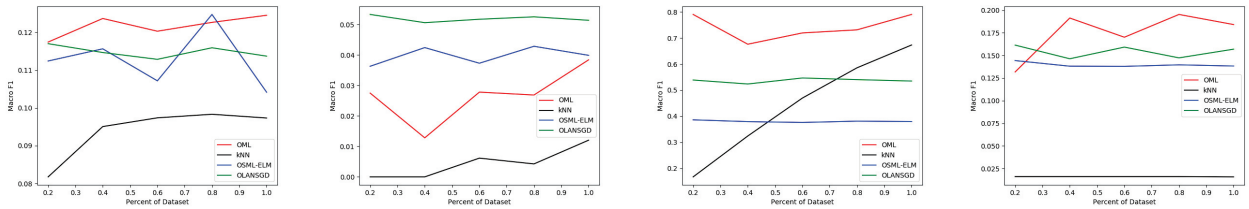
(a) CoreI5k (b) Enron (c) FullMedical (d) Emotions

Figure 4: Hamming Loss of various methods on Corel5k, Enron, Medical and Emotions datasets.

Based on the Lemma 2, we provide following theorem.

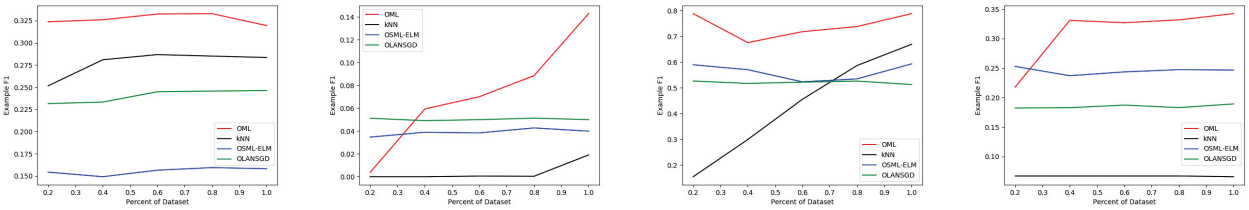
Theorem 1. *Let* $(x_1, y_1), \ldots, (x_T, y_T)$ *be a sequence of examples where* $x_t \in \mathbb{R}^{p\times 1}$ *and* $y_t \in \{0,1\}^{q\times 1}$. $V_t = (v_1, v_2, ..., v_d) \in \mathbb{R}^{q\times d}(d < q)$ *is projection matrix,* q *is in* $\mathbb{R}^n$. V_1 *is a non-zero matrix .*$U \in \mathbb{R}^{q\times d}(d < q)$. *Let* r *be the upper bound of* $||x_t||_2^2$. *Under the assumption of Lemma 2, the cumulative loss suffered on the sequence is bounded as follow,*

$$\sum_{t=1}^{T} l_t \leq \frac{||V_1 - U||_F^2(||P||_F^2 \cdot r + q)}{m \cdot c^2}$$



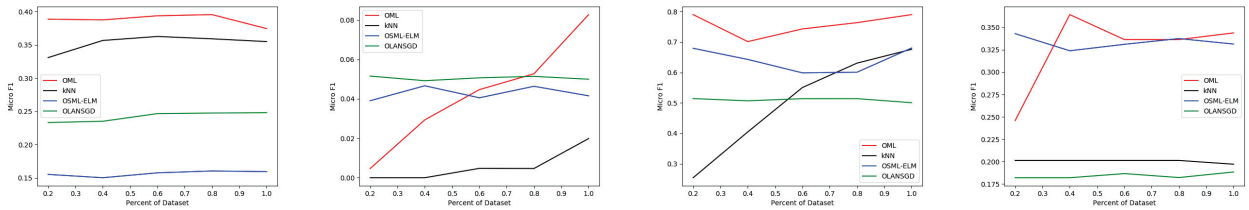
(a) Cal500 (b) Image (c) scene (d) slashdot

Figure 5: Macro F1 of various methods on Cal500, Image, scene and slashdot datasets.



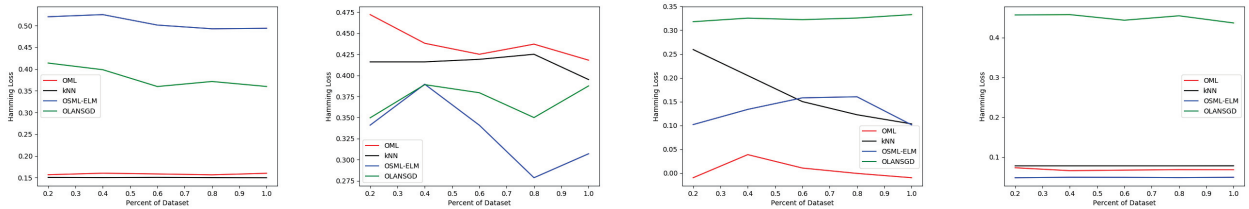
(a) Cal500 (b) Image (c) scene (d) slashdot

Figure 6: Example F1 of various methods on Cal500, Image, scene and slashdot datasets.



(a) Cal500 (b) Image (c) scene (d) slashdot

Figure 7: Micro F1 of various methods on Cal500, Image, scene and slashdot datasets.



(a) Cal500 (b) Image (c) scene (d) slashdot

Figure 8: Hamming Loss of various methods on Cal500, Image, scene and slashdot datasets.

Proof. By using Eq.(4), we get that

$$l_t \leq \Delta(y_t, y) + ||V_t^T P^T x_t - V_t^T y_t||_2^2$$

and,

$$||P^T x_t - y_t||_2^2 \leq ||P^T x_t||_2^2 + ||y_t||_2^2 \leq ||P^T||_F^2 \cdot r + q$$

Since $\Delta(y_t, y)$ is defined as l_1 norm, therefore $\Delta(y_t, y)$ is bounded by q. we can get y is bounded by q as well. By

using Lemma 2, we can get,

$$\begin{aligned}\sum_{t=1}^{T} l_t &\leq T \cdot q + \sum_{t=1}^{T} ||V_t||_F^2 \cdot ||P^T x_t - y_t||_2^2 \\ &\leq T \cdot q + \sum_{t=1}^{T} ||V_t||_F^2 \cdot (||P||_F^2 \cdot r + q) \\ &\leq T \cdot q + (\frac{||V_1 - U||_F^2}{m \cdot c^2} - \frac{q \cdot T}{(||P||_F^2 r + q)})(||P||_F^2 \cdot r + q) \\ &\leq \frac{||V_1 - U||_F^2 (||P||_F^2 \cdot r + q)}{m \cdot c^2}\end{aligned}$$

□

Therefore, the cumulative loss is bounded by $\frac{||V_1-U||_F^2(||P||_F^2 \cdot r+q)}{m \cdot c^2}$. As l_t is bounded, it guarantees the performance of our proposed model for unseen data.

Experiments

To evaluate the performance of our proposed online metric learning algorithm, we conduct experiments on four benchmark datasets: Corel5k, Enron, Medical and Emotions. The statistics of these datasets can be found in website[1]. All experiments are conducted on a workstation with 3.20GHz Intel CPU and 16GB main memory, running the Windows 10 platform.

Experiment Setup

Baseline Methods We compare our OML method with several state-of-the-art online multi-label prediction methods:

- OSML-ELM (Venkatesan et al. 2017): OSML-ELM uses a sigmoid activation function and outputs weights to predict the labels. In each step, output weight is learned from specific equation. OSML-ELM converts the label set from bipolar to unipolar representation in order to solve multi-label classification problems.
- OLANSGD (Park and Choi 2013): Based on Nesterov's smooth method, OLANSGD proposes to use accelerated nonsmooth stochastic gradient descent to solve the online multi-label classification problem. It updates the model parameters using only the gradient information calculated from a single label at each iteration. It then implements a ranking function that ranks relevant and irrelevant labels.
- kNN: We adapt the k nearest neighbor(kNN) algorithm to solve online multi-label classification problems. A Euclidean metric is used to measure the distances between instances.

In our experiment, the matrix V_1 is initialized as a normal distributed random matrix. Initially, we keep 20% of data for nearest neighbor searching. In our experiment, M is set to 100000 and m is set to 0.00001, while k is set to 10. The codes are provided by the respective authors. Parameter λ in OLANSGD is chosen from among $\{10^{-6}, 10^{-5}, \cdots, 10^{0}\}$ using five-fold cross validation. We use the default parameter for OSML-ELM.

[1]http://mulan.sourceforge.net

Performance Measurements To fairly measure the performance of our method and baseline methods, we consider the following evaluation measurements (Mao, Tsang, and Gao 2013; Zhang et al. 2015):

- Micro-F1: computes true positives, true negatives, false positives and false negatives over all labels, then calculates an overall F-1 score.
- Macro-F1: calculates the F-1 score for each label, then takes the average of the F-1 score.
- Example-F1: computes the F-1 score for all labels of each testing sample, then takes the average of the F-1 score.
- Hamming Loss: computes the average zero-one score for all labels and instances.

The smaller the Hamming Loss value, the better the performance; moreover, the larger the value of the other three measurements, the better the performance.

Prediction Performance

Figures 1 to 8 present the four measurement results for our method and baseline approaches in respect of various datasets. From these figures, we can see that:

- OML outperforms OSML-ELM and OLANSGD on various datasets, this is because neither of the latter approaches consider the label dependency.
- kNN is comparable to OSML-ELM and OLANSGD on most datasets, which demonstrates the competitive performance of kNN.
- OML achieves better performance than kNN on all datasets. This result illustrates that our proposed method is able to learn an appropriate metric for online multi-label classification.

Our experiments verify our theoretical studies and the motivation of this work: in short, our method is able to capture the interdependencies among labels, while also overcoming the bottleneck of kNN.

Conclusion

Current multi-label classification methods assume that all data are available in advance for leaning. Unfortunately, this assumption hinders off-line multi-label methods from handling sequential data. OLANSGD and OSML-ELM have overcome this limitation and achieved promising results in online multi-label classification; however, these methods lack a theoretical analysis for their loss functions, and also do not consider the label dependency, which has been proven to lead to degraded performance. Accordingly, to fill the current research gap on streaming data, we here propose a novel online metric learning method for multi-label classification based on the large margin principle. We first project instances and labels into the same embedding space for comparison, then learn the distance metric by enforcing the constraint that the distance between an embedded instance and its correct label must be smaller than the distance between the embedded instance and other labels. Thus, two nearby

instances from different labels will be pushed further. Moreover, we develop an efficient online algorithm for our proposed model. Finally, we also provide the upper bound of cumulative loss for our proposed model, which guarantees the performance of our method on unseen data. Extensive experiments corroborate our theoretical results and demonstrate the superiority of our method.

Acknowledgments

The author would like to thank all the anonymous reviewers for their useful comments.

References

Babbar, R., and Schölkopf, B. 2017. Dismec: Distributed sparse machines for extreme multi-label classification. In *WSDM*, 721–729.

Bhatia, K.; Jain, H.; Kar, P.; Varma, M.; and Jain, P. 2015. Sparse local embeddings for extreme multi-label classification. In *NIPS*, 730–738.

Boyd, S., and Vandenberghe, L. 2004. *Convex Optimization*. New York, NY, USA: Cambridge University Press.

Brinker, K., and Hullermeier, E. 2007. Case-based multilabel ranking. In *IJCAI*, 702–707.

Chen, Y., and Lin, H. 2012. Feature-aware label space dimension reduction for multi-label classification. In *NIPS*, 1538–1546.

Cheng, W., and Hüllermeier, E. 2009. Combining instance-based learning and logistic regression for multilabel classification. *Machine Learning* 76(2-3):211–225.

Crammer, K.; Dekel, O.; Keshet, J.; Shalev-Shwartz, S.; and Singer, Y. 2006. Online passive-aggressive algorithms. *Journal of Machine Learning Research* 7:551–585.

Dembczynski, K.; Cheng, W.; and Hüllermeier, E. 2010. Bayes optimal multilabel classification via probabilistic classifier chains. In *ICML*, 279–286.

Deng, J.; Berg, A. C.; Li, K.; and Li, F. 2010. What does classifying more than 10, 000 image categories tell us? In *ECCV*, 71–84.

Ding, S.; Zhao, H.; Zhang, Y.; Xu, X.; and Nie, R. 2015. Extreme learning machine: algorithm, theory and applications. *Artif. Intell. Rev.* 44(1):103–115.

Dong, A., and Bhanu, B. 2003. Concept learning and transplantation for dynamic image databases. In *ICME*, 765–768.

Gibaja, E., and Ventura, S. 2015. A tutorial on multilabel learning. *ACM Comput. Surv.* 47(3):52:1–52:38.

Hsu, D.; Kakade, S.; Langford, J.; and Zhang, T. 2009a. Multi-label prediction via compressed sensing. In *NIPS*, 772–780.

Hsu, D. J.; Kakade, S.; Langford, J.; and Zhang, T. 2009b. Multi-label prediction via compressed sensing. In *NIPS*, 772–780.

Kulis, B. 2013. Metric learning: A survey. *Foundations and Trends in Machine Learning* 5(4):287–364.

Liu, W., and Tsang, I. W. 2017. Making decision trees feasible in ultrahigh feature and label dimensions. *Journal of Machine Learning Research* 18(81):1–36.

Liu, W.; Xu, D.; Tsang, I. W.; and Zhang, W. 2019. Metric learning for multi-output tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41(2):408–422.

Liu, W.; Tsang, I. W.; and Müller, K. 2017. An easy-to-hard learning paradigm for multiple classes and multiple labels. *Journal of Machine Learning Research* 18(94):1–38.

Mao, Q.; Tsang, I. W.-H.; and Gao, S. 2013. Objective-guided image annotation. *IEEE Transactions on Image Processing* 22(4):1585–1597.

Park, S., and Choi, S. 2013. Online multi-label learning with accelerated nonsmooth stochastic gradient descent. In *ICASSP*, 3322–3326.

Petersen, K. B., and Pedersen, M. S. 2012. *The Matrix Cookbook*. Technical University of Denmark.

Popovici, R.; Weiler, A.; and Grossniklaus, M. 2014. Online clustering for real-time topic detection in social media streaming data. In *WWW*, 57–63.

Read, J.; Pfahringer, B.; Holmes, G.; and Frank, E. 2011. Classifier chains for multi-label classification. *Machine Learning* 85(3):333–359.

Tsoumakas, G.; Zhang, M.; and Zhou, Z. 2012. Introduction to the special issue on learning from multi-label data. *Machine Learning* 88(1-2):1–4.

Venkatesan, R.; Er, M. J.; Dave, M.; Pratama, M.; and Wu, S. 2017. A novel online multi-label classifier for high-speed streaming data applications. *Evolving Systems* 8(4):303–315.

Yen, I. E.; Huang, X.; Ravikumar, P.; Zhong, K.; and Dhillon, I. S. 2016. Pd-sparse : A primal and dual sparse approach to extreme multiclass and multilabel classification. In *ICML*, 3069–3077.

Zhang, M., and Zhou, Z. 2006. Multilabel neural networks with applications to functional genomics and text categorization. *IEEE Transactions on Knowledge and Data Engineering* 18(10):1338–1351.

Zhang, L.; Zhang, Q.; Zhang, L.; Tao, D.; Huang, X.; and Du, B. 2015. Ensemble manifold regularized sparse low-rank approximation for multiview feature embedding. *Pattern Recognit.* 48(10):3102–3112.

Zhang, X.; Graepel, T.; and Herbrich, R. 2012. Bayesian online learning for multi-label and multi-variate performance measures. In *AISTATS*, 956–963.

Zhou, J. T.; Tsang, I. W.; Ho, S.; and Müller, K. 2019a. N-ary decomposition for multi-class classification. *Machine Learning* 108(5):809–830.

Zhou, J. T.; Tsang, I. W.; Pan, S. J.; and Tan, M. 2019b. Multi-class heterogeneous domain adaptation. *Journal of Machine Learning Research* 20:57:1–57:31.