Efficient Multi-label Ranking for Multi-class Learning: Application to Object Recognition

Serhat S. Bucak, Pavan Kumar Mallapragada, Rong Jin and Anil K. Jain
Michigan State University
East Lansing, MI 48824, USA
{bucakser,pavanm,rongjin,jain}@cse.msu.edu

Abstract

Multi-label learning is useful in visual object recognition when several objects are present in an image. Conventional approaches implement multi-label learning as a set of binary classification problems, but they suffer from imbalanced data distributions when the number of classes is large. In this paper, we address multi-label learning with many classes via a ranking approach, termed multi-label ranking. *Given a test image, the proposed scheme aims to order all the object classes such that the relevant classes are ranked higher than the irrelevant ones. We present an efficient algorithm for multi-label ranking based on the idea of block coordinate descent. The proposed algorithm is applied to visual object recognition. Empirical results on the PASCAL VOC 2006 and 2007 data sets show promising results in comparison to the state-of-the-art algorithms for multi-label learning.*

1. Introduction

A number of problems in computer vision, such as visual object recognition, require an object to be assigned to a set of multiple classes, chosen from a large set of class labels. They are often cast into multi-label learning, in which each object can be simultaneously classified into more than one class. The most widely used approaches divide a multi-label learning task into multiple independent binary labeling tasks. The division usually follows one-vs-all (OvA), one-vs-one or the general error correction code framework [6, 13, 11]. Most of these approaches suffer from imbalanced data distributions when constructing binary classifiers to distinguish individual classes from the remaining classes. This problem becomes more severe when the number of classes is large. Another limitation of these approaches is that they are unable to capture the correlation among classes, which is known to be important in multi-label learning [22]. In this paper, we focus on the first problem of multi-label learning, namely imbalanced data distribution arising from dividing a multi-label learning task into a number of independent binary classification problems.

In this paper, we address multi-label learning with a large number of classes using a multi-label ranking approach. For a given example, multi-label ranking aims to order all the relevant classes at a higher rank than the irrelevant ones. By relaxing a classification problem into a ranking problem, multi-label ranking avoids constructing binary classifiers that distinguish individual classes from the other classes, thus alleviating the problem of imbalanced data distribution. In addition, by avoiding the binary decision about which subset of classes should be assigned to each example, multi-label ranking is usually more robust than the classification approaches, particularly when the number of classes is large.

Although several algorithms have been proposed for multi-label learning [22, 21, 8, 15], they are usually computationally expensive because the number of comparisons in multi-label ranking is $O(nK^2)$, where K is the number of classes and n is the number of training examples. The quadratic dependence on the number of classes makes it difficult to scale to a large number of classes. To this end, we present an efficient learning algorithm for multi-label ranking to handle a large number of classes. We apply the proposed algorithm to visual object recognition in which multiple object classes can be assigned to a single image. Our experiment with the PASCAL VOC 2006 dataset shows encouraging results in terms of both efficiency and efficacy.

2. Previous work

Ranking approach was first proposed in [9] for multi-label learning problems. Constraints derived from the multi-labeled instances were used in [9] to enforce that the ranking of relevant classes is higher than the irrelevant ones. [3] improves the computational efficiency of [9] by only considering the most violated constraints. Dekel *et al*. [5] and Shalev-Shwartz *et al*. [21] encode the ranking using

2009 IEEE 12th International Conference on Computer Vision (ICCV)
978-1-4244-4419-9/09/$25.00 ©2009 IEEE

a *preference graph*. In [5] a boosting based algorithm is used to learn the classifiers from a set of given instances and the corresponding preference graphs. In [21] a generalization of the hinge loss for the preference graphs is used for learning the ranking of classes. In [2], which presents a semi-supervised algorithm for multi-label learning by solving a Sylvester Equation (SMSE), a graph is constructed to capture the similarities between pair-wise categories. In [19] a vector function mapping is defined to get higher dimensional feature vectors that encode the model of individual categories as well as their correlations. A transductive multi-label classification approach, in which the multi-label interdependence is formulated as a pairwise Markov random field model, is proposed in [23]. In all these approaches, a ranking model is learned from the pairwise constraints between the relevant classes and the irrelevant classes. The number of pairwise constraints is square of the number of classes, which makes it computationally expensive when the number of classes is large. In contrast, the proposed framework for multi-label ranking that is computationally efficient and can handle a large number of classes (~ 100).

A number of approaches have been developed for multi-label learning that aim to capture the dependency among classes. In [22], the authors proposed to model the dependencies among the classes using a generative model. Ghamrawi et al. [8] try to capture the dependencies by defining a conditional random field over all possible combinations of the labels. In [15], a matrix factorization approach is used for multi-label learning that captures the class correlation via a class co-occurrence matrix. Hierarchical Bayesian approach is used in [24] to capture the dependency among classes. Overall, these approaches are computationally expensive when the number of classes is large. There are several approaches [17, 12, 25, 20, 16] for multi-label learning which encode the class dependence by assuming the sharing of important features among classes. [12] showed that a shared subspace model outperforms a number of state-of-the-art approaches for multi-label learning in terms of capturing the class correlation. We emphasize that our work does not focus on exploring the class correlation. It can be combined with these approaches to further improve the efficacy of multi-label learning.

3. Maximum margin framework for multi-label ranking

Let $x_i, i = 1, \dots, n$ be the collection of training examples where each example $x_i \in \mathbb{R}^d$ is a vector of d dimensions. Each training example x_i is annotated by a set of class labels, denoted by a binary vector $y_i = (y_i^1, \dots, y_i^K) \in \{-1, 1\}^K$, where K is the total number of classes, and $y_i^k = 1$ when x_i is assigned to class c_k and -1 otherwise. In multi-label ranking, we aim to learn K classification functions $f_k(x) : \mathbb{R}^d \mapsto \mathbb{R}, k = 1, \dots, K$, one for each class, such that for any example x, $f_k(x)$ is larger than $f_l(x)$ when x belongs to class c_k and does not belong to class c_l. We define the classification error $\varepsilon_i^{k,l}$ for an example x_i with respect to any two classes c_k and c_l, as follows

$$\varepsilon_i^{k,l} = I(y_i^k \neq y_i^l)\ell\left(\frac{y_i^k - y_i^l}{2}\left(f_k(x_i) - f_l(x_i)\right)\right), \qquad (1)$$

where $I(z)$ is an indicator function that outputs 1 when z is true and zero, otherwise. The loss $\ell(z)$ is defined to be the hinge loss, where $\ell(z) = \max(0, 1 - z)$. Note that the above error function outputs 0 when $y_i^k = y_i^l$, namely when no classification error is counted, i.e. x_i either belongs to both c_k and c_l or x_i does not belong to neither of the two classes.

Following the maximum margin framework for classification, we aim to search for the classification functions $f_k(x), k = 1, \dots, K$ that simultaneously minimize the overall classification error. This is summarized into the following optimization problem.

$$\min_{\{f_k \in \mathcal{H}_\kappa\}_{k=1}^K} \frac{1}{2}\sum_{k=1}^K |f_k|_{\mathcal{H}_\kappa}^2 + C\sum_{i=1}^n \sum_{k,l=1}^K \varepsilon_i^{k,l}, \qquad (2)$$

where $\kappa(x, x') : \mathbb{R}^d \times \mathbb{R} \mapsto \mathbb{R}$ is a kernel function, $\mathcal{H}_\kappa$ is a Hilbert space endowed with a kernel function $\kappa(\cdot, \cdot)$ and C is a constant parameter. Theorem 1 provides the representer theorem for $f_k(\cdot), k = 1, \dots, K$.

Theorem 1. *Classification functions $f_k(x)$, $k = 1, \dots, K$ that optimize (2) are represented in the following form*

$$f_k(x) = \sum_{i=1}^n y_i^k [\Gamma_i]^k \kappa(x_i, x), \qquad (3)$$

where $[\Gamma_i]^k = \sum_{l=1}^K \Gamma_i^{k,l}$. Note that $\Gamma_i \in \mathbf{S}^{K \times K}, i = 1, \dots, n$ are symmetric matrices that are obtained by solving the following optimization problem

$$\begin{aligned} \max \quad & \sum_{i=1}^n \sum_{k=1}^K [\Gamma_i]^k - \frac{1}{2}\sum_{k=1}^K \sum_{i,j=1}^n \kappa(x_i, x_j) y_i^k y_j^k [\Gamma_i]^k [\Gamma_j]^k \\ \textit{s. t.} \quad & \Gamma_i^{k,l} = \begin{cases} 0 \leq \Gamma_i^{k,l} \leq C & y_i^k \neq y_i^l \\ 0 & \textit{otherwise} \end{cases} \\ & \Gamma_i = [\Gamma_i]^\top, i = 1, \dots, n;\ k, l = 1, \dots, K. \end{aligned} \qquad (4)$$

Proof. See Appendix A.1 □

The constraints in Eq (4) explicitly capture the relationship between the classes. When an instance x_i belongs to class c_k, but does not belong to class c_l, the value of $\Gamma_i^{k,l}$ is positive, causing x_i to be a support vector. The positive terms $\Gamma_i^{k,l}$ are combined into $[\Gamma_i^k]$, which is used in computing the ranking function for class c_k.

4. Approximate formulation

A straightforward approach that directly solves (4) by a standard quadratic programming approach is computationally expensive when the number of classes K is large because the number of constraints is $O(K^2)$. We show that the relationship between multi-label ranking and one-versus-all approach provides insight for deriving an approximate formulation for (4) that can be solved efficiently.

4.1. Relation to one-versus-all approach

Consider constructing $f_k(x)$ in (2) by the OvA approach. The resulting representer theorem for $f_k(x)$ is

$$f_k(x) = \sum_{i=1}^{n} y_i^k \alpha_i^k \kappa(x_i, x), k = 1, \dots, K \tag{5}$$

where $\alpha_k^i, i = 1 \dots, n; k = 1, \dots, K$, are obtained by solving the following optimization problem

$$\begin{aligned} \max \quad & \sum_{i=1}^{n}\sum_{k=1}^{K} \alpha_i^k - \frac{1}{2}\sum_{k=1}^{K}\sum_{i,j=1}^{n} \kappa(x_i, x_j) y_i^k y_j^k \alpha_i^k \alpha_j^k \\ \text{s. t.} \quad & \alpha_i^k \in [0, C], \quad i = 1, \dots, n; k = 1, \dots, K. \end{aligned} \tag{6}$$

Comparing the above formulation to (4), we clearly see the mapping, i.e., $[\Gamma_i]^k \leftrightarrow \alpha_i^k$. Hence, the first simplification is to relax (4) by treating each $[\Gamma_i]^k$ as an independent variable, which approximates (4) into the following optimization problem

$$\begin{aligned} \max \quad & \sum_{i=1}^{n}\sum_{k=1}^{K} \alpha_i^k - \frac{1}{2}\sum_{k=1}^{K}\sum_{i,j=1}^{n} \kappa(x_i, x_j) y_i^k y_j^k \alpha_i^k \alpha_j^k \\ \text{s. t.} \quad & 0 \le \alpha_i^k \le C \sum_{l=1}^{K} I(y_i^k \neq y_i^l), \\ & i = 1, \dots, n; k = 1, \dots, K. \end{aligned} \tag{7}$$

Note that the constraint $\alpha_i^k \le C \sum_{l=1}^{K} I(y_i^k \neq y_i^l)$ follows

$$[\Gamma_i]^k = \sum_{l=1}^{K} I(y_i^k \neq y_i^l)\Gamma_i^{k,l} \le C \sum_{l=1}^{K} I(y_i^k \neq y_i^l).$$

While the problem in Eq (7) can be decomposed into K independent problems, similar to an OvA SVM, this is not adequate for multi-label ranking as the depdendence between the functions $f_k(x), k = 1, \dots, K$ cannot be captured.

4.2. Proposed approximation

In this section, we present a better approximation of (4) compared to the one presented in Eq (7). Without loss of generality, consider a training example x_i that is assigned to the first a classes, and is not assigned to the remaining $b = K - a$ classes. According to the definition of Γ_i in (4), we can rewrite Γ as

$$\Gamma = \begin{pmatrix} 0 & Z \\ Z^\top & 0 \end{pmatrix} \tag{8}$$

where $Z \in [0, C]^{a \times b}$. Using this notation, variable $\tau_k = [\Gamma_i]^k$ is computed as

$$\tau_k = \begin{cases} \sum_{l=1}^{b} Z_{k,l} & 1 \le k \le a \\ \sum_{l=1}^{a} Z_{l,k} & a+1 \le k \le K \end{cases}$$

where $Z_{k,l}$ is an element in Z that is bounded by 0 and C. According to the above definition, for each instance, τ_k is the sum of either the k^{th} column or the k^{th} row of Z depending on whether the label k is relevant to that instance or not. Formulating τ_k by using Z brings several advantages. Firstly, it enables us to derive constraints for τ_k explicitly in the optimization. Secondly, all τ_k variables depend on each other in the optimization since the components of these variables are taken from a closed domain Z. This relationship is in fact a special case of the constraint given in Eq (4). The constraint in Eq (4) intuitively forces a balance between the irrelevant and relevant labels of an instance by requiring the sum of the upper bounds of $[\Gamma_i]^k$ that correspond to relevant classes to be equal to that of $[\Gamma_i]^k$ that correspond to irrelevant classes. Obtaining τ_k from Z as formulated above introduces an additional constraint by forcing the sum of the weights corresponding to the relevant labels to be equal to the sum of the weights that are associated with irrelevant ones. This constraint is useful in dealing with the imbalance between the number of relevant and irrelevant labels as well as capturing the dependencies between the classes for that instance.

In order to convert $\tau_k, k = 1, \dots, K$ into free variables, we need to derive explicit constraints on τ_k that will ensure that each solution of τ_k will result in a feasible solution for Z. Let us first consider a simple case in which we only require elements in Z to be non-negative. Theorem 2 provides the constraints on τ_k.

Theorem 2. *The following two domains Q_1 and Q_2 for vector $\tau = (\tau_1, \dots, \tau_K)$ are equivalent*

$$\begin{aligned} Q_1 &= \{\tau \in \mathbb{R}^K : \exists Z \in \mathbb{R}_+^{a \times b} \text{s. t.} \\ & \quad \tau_{1:a} = Z\mathbf{1}_b, \tau_{a+1:K} = Z^\top \mathbf{1}_a\} \end{aligned} \tag{9}$$

$$Q_2 = \left\{\tau \in \mathbb{R}_+^K : \sum_{k=1}^{a} \tau_k = \sum_{k=a+1}^{K} \tau_k\right\} \tag{10}$$

Proof. See Appendix A.2. □

Theorem 2 which states that the two domains Q_1 and Q_2 are equivalent for vector τ leads to the following corollary.

Corollary 1. *Consider the following two domains Q_1 and Q_2 for vector $\tau = (\tau_1, \ldots, \tau_K)$*

$$\begin{aligned} Q_1 &= \{\tau \in \mathbb{R}^K : \exists Z \in [0,C]^{a\times b} \textit{ s. t.} \\ &\quad \tau_{1:a} = Z\mathbf{1}_b, \tau_{a+1:K} = Z^\top \mathbf{1}_a\} \qquad (11) \\ Q_2 &= \left\{\tau \in [0,C]^K : \sum_{k=1}^{a} \tau_k = \sum_{k=a+1}^{K} \tau_k \right\} \qquad (12) \end{aligned}$$

We have $\tau \in Q_2 \Rightarrow \tau \in Q_1$.

The above corollary becomes the basis for our approximation. Instead of defining matrix variables $\Gamma_i, i = 1, \ldots, n$ as in (4), we introduce the variable α_i^k for $[\Gamma_i]^k$. We furthermore restrict $\alpha_i = (\alpha_i^1, \ldots, \alpha_i^k)$ to be in the domain $\mathcal{G} = \left\{\tau \in [0,C]^K : \sum_{k=1}^{a} \tau_k = \sum_{k=a+1}^{K} \tau_k\right\}$ to ensure that feasible Γ_i can be recovered from a solution of α_i^k. The resulting approximate optimization is

$$\begin{aligned} \max \quad & \sum_{i=1}^{n}\sum_{k=1}^{K} \alpha_i^k - \frac{1}{2}\sum_{k=1}^{K}\sum_{i,j=1}^{n} \kappa(x_i,x_j) y_i^k y_j^k \alpha_i^k \alpha_j^k \\ \text{s. t.} \quad & \sum_{k=1}^{K} I(y_i^k = 1)\alpha_i^k = \sum_{k=1}^{K} I(y_i^k = -1)\alpha_i^k, \\ & \alpha_i^k \in [0,C], \quad i = 1, \ldots, n, k = 1, \ldots, K \quad (13) \end{aligned}$$

Unlike Eq (7), Eq (13) cannot be solved as K independent problems since for each instance x_i, the α_i^k from all the classes $c_k, k = 1, \ldots, K$ are involved in the constraint. According to these constraints, for each instance the sum of the weights corresponding to the relevant labels should be equal to the sum of the weights that are associated with irrelevant ones. Theorem 2 showed that by adding this constraint to the problem, the relationships between the classes can be exploited and used without explicitly determining the set Z and the matrices Γ_i. Another advantage of this formulation is that no assumptions on the form of these relationships (e.g., pairwise relationship) is made.

5. Efficient algorithm

We follow the work of Lin *et al.* [10] and solve Eq (13) by coordinate descent. At each iteration, we choose one training example (x_i, y_i) and the related variables $\alpha_i = (\alpha_i^1, \ldots, \alpha_i^K)$, while fixing the remaining variables. The resulting optimization problem becomes

$$\begin{aligned} \max \quad & \sum_{k=1}^{K} \alpha_i^k - \frac{1}{2}\sum_{k=1}^{K} y_i^k f_k^{-i}(x_i)\alpha_i^k - \frac{\kappa(x_i,x_i)}{2}\sum_{k=1}^{K}(\alpha_i^k)^2 \\ \text{s. t.} \quad & \alpha_i \in [0,C]^K, \; y_i^\top \alpha_i = 0 \qquad (14) \end{aligned}$$

where $f_k^{-i}(x_i)$ is the leave-one-out prediction that can be computed as $f_k^{-i}(x) = \sum_{j\neq i} y_j^k \alpha_j^k \kappa(x_j, x)$.

Theorem 3. *The optimal solution to (14) is written as*

$$\alpha_i^k = \pi_{[0,C]}\left(\frac{1 + \lambda y_i^k - \frac{1}{2} y_i^k f_k^{-i}(x_i)}{\kappa(x_i,x_i)}\right), k = 1, \ldots, K \quad (15)$$

where λ is the solution to the following equation

$$g(\lambda) = \sum_{k=1}^{K} h\left(\frac{y_i^k + \lambda - \frac{1}{2} f_k^{-i}(x_i)}{\kappa(x_i,x_i)}, y_i^k C\right) = 0. \quad (16)$$

Here $h(x,y) = \pi_{[0,y]}(x)$ if $y > 0$ and $h(x,y) = \pi_{[y,0]}(x)$ if $y \leq 0$. Function $\pi_G(x)$ projects x onto the region G.

Proof. See Appendix A.3. □

The function $g(\lambda)$ defined in (16) is a monotonically increasing function of λ which can be solved using bisection search. The lower and upper bounds for λ for bisection search are shown in the proposition below.

Proposition 1. *The value of λ that satisfies (16) is bounded by $\lambda_{\min}$ and $\lambda_{\max}$. Define, $\kappa_{ii} = \kappa(x_i, x_i)$ and $G = [0, C]$,*

$$\eta_{k+}^{-i} = 1 + \frac{1}{2} f_k^{-i}(x_i) \qquad \eta_{k-}^{-i} = 1 - \frac{1}{2} f_k^{-i}(x_i)$$

$$\Delta = \sum_{k=1}^{K} \delta(y_i^k, 1)\pi_G\left(\frac{\eta_{k-}^{-1}}{k_{ii}}\right) - \sum_{k=1}^{K} \delta(y_i^k, -1)\pi_G\left(\frac{\eta_{k+}^{-i}}{\kappa_{ii}}\right)$$

$$a_{\min} = -C\kappa_{ii} + \min_{y_i^k=-1} \eta_{k+}^{-i} \qquad b_{\min} = -\max_{y_i^k=1} \eta_{k-}^{-i}$$

$$a_{\max} = Ck_{ii} - \min_{y_i^k=1} \eta_{k-}^{-i} \qquad b_{\max} = \max_{y_i^k=-1} \eta_{k+}^{-i}$$

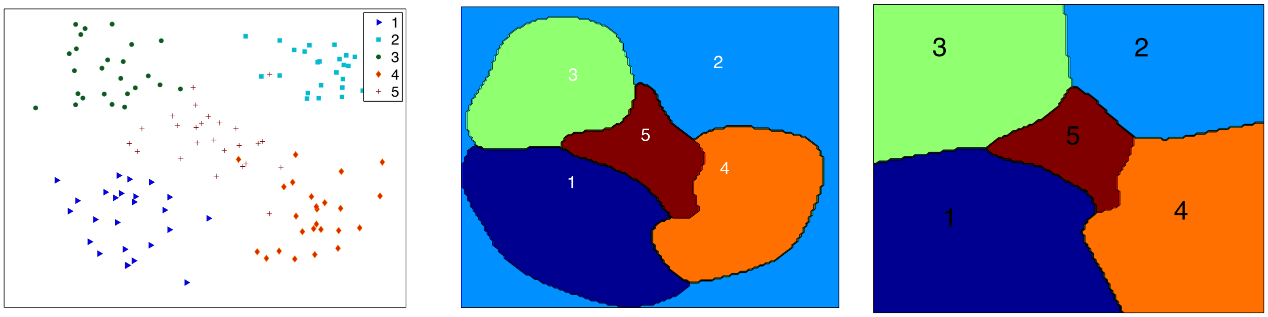
If $\Delta < 0$,we have $\lambda_{\min} = 0$ and $\lambda_{\max} = \max(a_{\max}, b_{\max})$. If $\Delta > 0$, we have $\lambda_{\max} = 0$ and $\lambda_{\min} = \min(a_{\min}, b_{\min})$.

Proof. See supplementary documents. □

Once λ is calculated by applying bisection search between the bounds $\lambda_{\min}$ and $\lambda_{\max}$, it is straightforward to calculate the coefficients α_i^k and finally the ranking functions $f_k(x)$ for any new instance x.

6. Experimental results

We start with a simple example to demonstrate the advantage of a multilabel ranking method over methods that combine several binary classifiers for multiclass learning. Figure 1 shows an illustration of the proposed approach, applied to a single-label multiclass classification task, on a synthetic dataset. The two dimensional data with the true labels are shown in Figure 1(a). The decision boundaries obtained by one-vs-rest (OvA) SVM and the proposed approach are shown in Figures 1(b) and (c), respectively. We used an RBF kernel with the parameter $\sigma = 1$ to generate the decision boundaries. We observe that in the OvA



(a)

(b)

(c)

Figure 1. Illustration of the proposed approach on a single-label five-class classification task. (a) Two dimensional data points with labels, (b) Decision boundary obtained using OvA SVM and (c) decision boundary obtained using the proposed ranking approach.

approach, the decision boundary fits tightly around classes 1,3,4 and 5. The region outside these class boundaries is assigned to class 2, which is clearly not acceptable based on the input data. The proposed approach partitions the space in a more reasonable way, as shown in Figure 1(c).

Data sets The PASCAL VOC Challenge 2006 and 2007 data sets [1] are used in our study. VOC 2006 data contains 5304 images with 9507 annotated objects while VOC 2007 has 9963 images with 14319 objects. Since the focus of this study is multi-label learning and about 70% of images in these data sets are labeled by a single object, we did not use the default partition. Instead, we formed the training data set for VOC 2006 experiments by randomly selecting 1600 images with a single object and 800 images with multiple objects, and used the remaining images for testing. Similarly, we randomly chose 3200 images with a single object and 2000 images with multiple objects for training from VOC 2007. It should also be noted that there are a total of 10 classes in VOC 2006 set while this number is 20 for VOC 2007. A bag-of-words model is used to represent image content. Following the standard approach [4], we obtained SIFT descriptors from each image in the dataset and then clustered these feature vectors into $5,000$ clusters by an approximate K-means algorithm [18].

Evaluation metric: Area under the ROC Curve (AUC) is used as the evaluation metric in our study. Since we focus on multi-label ranking, we rank the classes in the descending order of their scores. For each image, we predict its categories by the first k objects with the largest scores. We vary k, i.e., the number of predicted objects, from 1 to the number of total categories, and compute the true positive and false positive rates, which lead to the calculation of AUC. Note that this is different from other studies of object recognition where AUC is computed for each category. We did not compute AUC for each category because our method only ranks object categories for an image without making binary decision. Since the focus of this study is multi-label learning, we also evaluate AUC for images with single object and AUC for images with multiple objects, separately. All the experiments are repeated several times, and AUC averaged over these runs is reported as the final result.

Baseline methods: We compare ranking ability of the proposed method to three baseline methods: (i) LIBSVM [7] implementation of OvA SVM classifier, which is shown to outperform multi-class SVM methods in [11]. (ii) SVM-perf [14] that is designed to optimize Area Under ROC Curve (AUC), which are used as the evaluation metrics in our study. (iii) Multiple Label Shared Space Model (MLSSM) in [12] that makes use of the class correlations and is reported to give the best performance compared to other state-of-the-art methods that explore class correlation.

We use the chi-squared kernel in our experiments, which has shown to outperform the other kernels for object recognition. The same values of the parameters C and σ are used for all the binary classifiers in the OvA SVM. The optimal values C and σ are chosen by a cross-validation grid search in which different values of $C = \{10^{-4}, 10^{-2}, \cdots, 10^{6}\}$ and $\sigma = \{2^{-11}, 2^{-9}, \cdots, 2^{3}\}$ are tried.

Object recognition: The goal of this study is to verify (i) the proposed multi-label ranking approach is more effective for object recognition than binary classification based methods such as SVM, and (ii) the proposed multi-label ranking approach is computationally more efficient than the binary classification based methods for multi-label learning.

The AUC results for PASCAL VOC Challenge 2006 and 2007 data sets are summarized in Table 1. Three AUC results are reported: *overall* AUC for all test images, *multi-obj* AUC for test images with multiple objects, and *single-obj* AUC for test images with a single object. When evaluating AUC for all the test images, both the proposed method and LIBSVM yield the best performance for VOC 2006 data set, and the difference between different methods is small.

Table 1. Mean and standard deviation of AUC (%)

VOC 2006	Proposed	LIBSVM	SVM-perf	MLSSM
overall	76.8 ± 0.4	76.4 ±0.6	74.2 ± 0.8	75.8 ± 0.6
multi-obj	81.2 ± 0.9	74.3 ± 0.7	74.0 ± 0.1	77.8 ± 0.7
single-obj	74.4 ± 1.0	76.8 ± 0.7	75.6 ± 0.7	75.6 ± 0.7
VOC 2007	Proposed	LIBSVM	SVM-perf	MLSSM
overall	76.0 ± 0.2	74.8 ±0.1	68.2 ± 0.6	74.7 ± 0.2
multi-obj	79.4 ± 0.7	77.9 ± 0.2	69.4 ± 0.8	78.6 ± 0.1
single-obj	73.1 ± 0.5	72.2 ± 0.2	67.9 ± 0.2	71.29 ± 0.1

Table 2. Mean and standard deviation for running times (sec)

	Proposed	LIBSVM	SVM-perf	MLSSM
VOC 06	43.2 ±1.4	1147.5 ± 349.7	673.7 ±65.8	324.2 ±16.9
VOC 07	447.3 ±0.3	7720.7 ± 34.2	1597.3 ±3.21	1821.04 ±5.1

However, for images with multiple objects, the two methods designed for multi-label learning, i.e., the proposed method and MLSSM perform better than the other two competitors. Compared to MLSSM, the proposed algorithm performs significantly better. We emphasize that unlike MLSSM that makes strong assumption about the correlation among classifiers (i.e., all the classifier share the same subspace), the proposed method makes no assumption regarding class correlation. In the future, we plan to investigate how to incorporate the class correlation into the proposed method for multi-label ranking. For images with a single object, although we observe that the proposed method is outperformed by the other three methods for VOC 2006, it gives the best results for all three cases in VOC 2007. This improvement is due to the increased number of object classes in VOC 2007. It is also surprising to observe that SVM-perf performs worse than LIBSVM even though it is targeted on the evaluation metric.

We also evaluate the efficiency of the proposed algorithm for both data sets. Table 2 summaries the running time of four algorithms in comparison. Note that both the number of classes and number of training samples in VOC 2007 set are twice of those in VOC 2006 data. We clearly observe that the proposed algorithm is computationally more efficient than the three baseline methods.

Finally, Figure 2 shows examples of images and the objects predicted by different methods. We clearly see that overall the objects identified by the proposed method are more relevant to the visual content of images than the three baseline methods, especially for the images that contain several objects.

7. Conclusions and discussions

We have introduced an efficient multi-label ranking scheme which offers a direct solution to multi-label ranking unlike the conventional methods that use a set of binary classifiers for multiclass classifier learning. This direct approach enables us to capture the relationships between the class labels without making any assumptions on them. The strength of the proposed approach lies in establishing the relationships between the classifiers by treating them as ranking functions. An efficient algorithm is presented for multi-label ranking. Empirical study of object recognition with PASCAL VOC Challenge 2006 and 2007 data sest demonstrates that the proposed method outperforms state-of-the-art methods.

8. Acknowledgements

This work is supported in part by National Science Foundation (IIS-0643494) and US Army Research (ARO Award W911NF-08-010403). Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of NFS and ARO.

References

[1] http://www.pascal-network.org/challenges/voc/databases.html. 5

[2] G. Chen, Y. Song, F. Wang, and C. Zhang. Semi-supervised multi-label learning by solving a sylvester equation. In *Proc. SIAM International Conference on Data Mining (SDM)*, pages 410–419, 2008. 2

[3] K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, and Y. Singer. Online passive-aggressive algorithms. *Journal of Machine Learning Research*, 7:551–585, 2006. 1

[4] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *Proc. ECCV*, pages 451–464, 2004. 5

[5] O. Dekel, C. Manning, and Y. Singer. Log-linear models for label ranking. In *NIPS 17*, pages 497–504, 2004. 1, 2

[6] T. G. Dietterich and G. Bakiri. Solving multiclass learning problems via error-correcting output codes. *Journal of Artificial Intelligence Research*, 2:263–286, 1995. 1

[7] R.-E. Fan, P.-H. Chen, and C.-J. Lin. Working set selection using second order information for training svm. *Journal of Machine Learning Research*, 6:1889–1918, 2005. 5

[8] N. Ghamrawi and A. McCallum. Collective multi-label classification. In *Proc. 14th CIKM*, pages 195–200, 2005. 1, 2

[9] S. Har-Peled, D. Roth, and D. Zimak. Constraint classification for multiclass classification and ranking. In *NIPS 15*, pages 809–816, 2002. 1

[10] C.-J. Hsieh, K.-W. Chang, C.-J. Lin, S. Keerthi, and S. Sundararajan. A dual coordinate descent method for large-scale linear svm. In *Proc. ICML*, pages 408–415, 2008. 4

[11] C.-W. Hsu and C.-J. Lin. A comparison of methods for multiclass support vector machines. *IEEE Transactions on Neural Networks*, 13(2):415–425, 2002. 1, 5

[12] S. Ji, L. Tang, S. Yu, and J. Ye. Extracting shared subspace for multi-label classification. In *Proc. 14th ACM SIGKDD*, pages 381–389, 2008. 2, 5

Input Image				
True objects	*people, motorbike, car*	*car, prople, dog*	*people, motorbike, car*	*car, people, bike*
Proposed	*people, motorbike, car*	*car, prople, dog*	*people, motorbike, car*	*car, people, bike*
LIBSVM	*people, car, bus*	*people, car, horse*	*people, cow, motorbike*	*motorbike, people, horse*
SVM-perf	*people, horse, car*	*car, people, cat*	*people, cat, car*	*people, motorbike, horse*
MLSSM	*people, car, bus*	*people, dog, cat*	*people, car, bus*	*bike, people, car*

Figure 2. For two images from the dataset, the original lables are given in addition to the outputs of the proposed method and the best method among the rest

[13] T. Joachims. Text categorization with support vector machines: Learning with many relevant features. In *Proc. ECML*, pages 137–142, 1998. 1

[14] T. Joachims. A support vector method for multivariate performance measures. In *Proc. 22nd ICML*, pages 377–384, 2005. 5

[15] Y. Liu, R. Jin, and L. Yang. Semi-supervised multi-label learning by constrained non-negative matrix factorization. In *Proc. 21st AAAI*, pages 421–426, 2006. 1, 2

[16] N. Loeff and A. Farhadi. Scene discovery by matrix factorization. In *Proc. ECCV*, pages 451–464, 2008. 2

[17] A. McCallum. Multi-label text classification with a mixture model trained by EM. In *Proc. AAAI Workshop on Text Learning*, 1999. 2

[18] M. Muja and D. G. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. In *Proc. Int. Conference on Computer Vision Theory and Applications*, 2009. 5

[19] G.-J. Qi, X.-S. Hua, Y. Rui, J. Tang, T. Mei, and H.-J. Zhang. Correlative multi-label video annotation. In *Proc. ACM Multimedia (MM)*, pages 17–26, 2007. 2

[20] A. Quattoni, M. Collins, and T. Darrell. Transfer learning for image classification with sparse prototype representations. In *CVPR*, pages 1–8, 2008. 2

[21] S. Shalev-Shwartz and Y. Singer. Efficient learning of label ranking by soft projections onto polyhedra. *Journal of Machine Learning Research*, 7:1567–1599, 2006. 1, 2

[22] N. Ueda and K. Saito. Parametric mixture models for multi-labeled text. In *NIPS 15*, pages 721–728, 2002. 1, 2

[23] J. Wang, Y. Zhao, X. Wu, and X.-S. Hua. Transductive multi-label learning for video concept detection. In *Proc. ACM International Conference on Multimedia Information Retrieval*, pages 298–304, 2008. 2

[24] K. Yu and W. Chu. Gaussian process models for link analysis and transfer learning. In *NIPS 20*, pages 1657–1664, 2008. 2

[25] K. Yu, S. Yu, and V. Tresp. Multi-label informed latent semantic indexing. In *Proc. SIGIR*, 2005. 2

A. Proofs of theorems

A.1. Proof of Theorem 1

For notational convenience, let us define

$$\Delta_i^{k,l} = \frac{y_i^k - y_i^l}{2}\langle f_k - f_l, \kappa(x_i,\cdot)\rangle_{H_\kappa}$$

Using this, the objective function in (2) can be rewritten as

$$h(f) = \frac{1}{2}\sum_{l=1}^{K}\langle f_l, f_l\rangle_{H_K} + C\sum_{i=1}^{n}\sum_{l,k=1}^{K} I(y_i^l \neq y_i^k)\ell\left(\Delta_i^{k,l}\right)$$

We then rewrite $\ell(z)$ as

$$\ell(z) = \max_{x\in[0,1]}(x - xz)$$

Using the above expression for $\ell(z)$, the second term in $h(f)$ can be rewritten as,

$$\sum_{i=1}^{n}\sum_{l,k=1}^{K} I(y_i^l \neq y_i^k)\max_{\gamma_i^{k,l}\in[0,C]}\left(\gamma_i^{k,l} - \gamma_i^{k,l}\Delta_i^{k,l}\right)$$

The problem in (2) now becomes a convex-concave optimization problem as

$$\min_{f_l\in\mathcal{H}_K}\max_{\gamma_i^{l,k}\in[0,C]} g(f,\gamma)$$

where

$$\begin{aligned} g(f,\gamma) &= \sum_{i=1}^{n}\sum_{l,k=1}^{K} I(y_i^l \neq y_i^k)\gamma_i^{l,k} + \frac{1}{2}\sum_{l=1}^{K}\langle f_l, f_l\rangle_{H_K} \\ &\quad -\sum_{i=1}^{n}\sum_{l,k=1}^{K} I(y_i^l \neq y_i^k)\gamma_i^{l,k}\Delta_i^{k,l} \end{aligned}$$

According to von Newman's lemma, we could switch minimization with maximization. By taking the minimization over f_l first, we have

$$f_l(x) = \sum_{i=1}^{n} y_i^l\left(\sum_{k=1}^{K} I(y_i^l \neq y_i^k)\gamma_i^{l,k}\right)k(x_i,x)$$

In the above derivation, we use the relation $I(y_i^l \neq y_i^k)(y_i^l - y_i^k) = 2y_i^l$. To simplify our notation, we introduce $\Gamma_i \in [0, C]^{K\times K}$ where $\Gamma_i^{l,k} = \gamma_i^{l,k}$ if $y_i^l \neq y_i^k$ and zero otherwise. Note that since $\gamma_i^{l,k} = \gamma_i^{k,l}$, we have $\Gamma_i = [\Gamma_i]^\top$. We furthermore introduce the notation $[\Gamma_i]^l$ as the sum of the elements in the lth row, i.e., $[\Gamma_i]^l = \sum_{k=1}^K \Gamma_i^{l,k}$. Using these notations, we have $f_l(x)$ expressed as

$$f_l(x) = \sum_{i=1}^n y_i^l [\Gamma_i]^l k(x_i, x)$$

Finally, the remaining maximization problem becomes

$$\begin{aligned}
\max \quad & \sum_{i=1}^n \sum_{k=1}^K [\Gamma_i]^k - \frac{1}{2} \sum_{k=1}^K \sum_{i,j=1}^n k(x_i, x_j) y_i^k y_j^k [\Gamma_i]^k [\Gamma_j]^k \\
\text{s. t.} \quad & \Gamma_i^{k,l} = \begin{cases} 0 \leq \Gamma_i^{k,l} \leq C & y_i^k \neq y_i^l \\ 0 & \text{otherwise} \end{cases} \\
& \Gamma_i = [\Gamma_i]^\top, \quad i = 1, \ldots, n; k, l = 1, \ldots, K
\end{aligned}$$

A.2. Proof of Theorem 2.

It is straightforward to shown $\tau \in Q_1 \rightarrow \tau \in Q_2$. The main challenge is to show the other direction, i.e., $\tau \in Q_2 \rightarrow \tau \in Q_1$. For a given τ, in order to check if there exists $Z \in [0, C]^{a\times b}$ such that $\tau 1 : a = Z\mathbf{1}_b$ and $\tau_{a+1:K} = Z^\top \mathbf{1}_a$, we need show that the following optimization problem is feasible

$$\begin{aligned}
\min \quad & 0 && (17)\\
\text{s. t.} \quad & Z \in \mathbb{R}_+^{a\times b}, \tau 1 : a = Z\mathbf{1}_b,\ \tau_{a+1:K} = Z^\top \mathbf{1}_a
\end{aligned}$$

For the convenience of presentation, we denote by $\mu_a = \tau_{1:a} \in \mathbb{R}^a$, and by $\mu_b = \tau_{a+1:K} \in \mathbb{R}^b$, and rewrite the above feasibility problem as

$$\begin{aligned}
\min \quad & 0 && (18)\\
\text{s. t.} \quad & Z \in [0, C]^{a\times b}, \mu_a = Z\mathbf{1}_b,\ \mu_b = Z^\top \mathbf{1}_a
\end{aligned}$$

It is important to note that, for the above optimization problem, its optimal value is 0 when the solution is feasible, and $+\infty$ when no feasible solution satisfies the condition. By introducing the Lagrangian multipliers $\lambda_a \in \mathbb{R}^a$ for $\mu_a = Z\mathbf{1}_b$ and $\lambda_b \in \mathbb{R}^b$ for $\mu_b = Z^\top \mathbf{1}_b$, we have

$$\min_{Z \succeq 0} \max_{\lambda_a, \lambda_b} \lambda_a^\top(\mu_a - Z\mathbf{1}_b) + \lambda_b^\top(\mu_b - Z^\top \mathbf{1}_a) \qquad (19)$$

By taking the minimization over Z, we have

$$\begin{aligned}
\max_{\lambda_a, \lambda_b} \quad & \lambda_a^\top \mu_a + \lambda_b^\top \mu_b && (20)\\
\text{s. t.} \quad & \lambda_a \mathbf{1}_b^\top + \mathbf{1}_a \lambda_b^\top \preceq \mathbf{0}
\end{aligned}$$

To decide if there is a feasible solution to (18), the necessary and sufficient condition is that the optimal value for (20) is zero. First, we show that the objective function of (20) is upper bounded by zero under the constraint $\lambda_a \mathbf{1}_b^\top + \mathbf{1}_a \lambda_b^\top \preceq \mathbf{0}$. We denote by λ_a^+ and λ_b^+ the maximum elements in vector λ_a and λ_b, respectively, i.e, $\lambda_a^+ = \max_{1\leq i\leq a} [\lambda_a]_i$ and $\lambda_b^+ = \max_{1\leq i\leq b} [\lambda_b]_i$. Evidently, according to the constraint $\lambda_a \mathbf{1}_b^\top + \mathbf{1}_a \lambda_b^\top \preceq \mathbf{0}$, we have $\lambda_a^+ + \lambda_b^+ \leq 0$. We then have the objective function bounded as

$$\lambda_a^\top \mu_a + \lambda_b^\top \mu_b \leq \lambda_a^+ \mathbf{1}_a^\top \mu_a + \lambda_b^+ \mathbf{1}_b^\top \mu_b = (\lambda_a^+ + \lambda_b^+) \mathbf{1}_a^\top \mu_a \leq 0$$

Second, it is straightforward to verify that zero optimal value is obtainable by setting $\lambda_a = \mathbf{0}_a$ and $\lambda_b = \mathbf{0}_b$. Combining the above two arguments, we have the optimal value for (20) is zero, which therefore indicates that there is a feasible solution to (18). By this, we prove that $\tau \in Q_2 \rightarrow \tau \in Q_1$.

A.3. Proof of Theorem 3

We first turn the problem in (14) into the following min-max problem

$$\begin{aligned}
\max_{\alpha_i \in [0,C]^K} \min_{\lambda} \quad & \sum_{l=1}^K \alpha_i^l - \frac{1}{2} \sum_{k=1}^K y_i^k f_k^{-i}(x_i) \alpha_i^k - \\
& \frac{k(x_i, x_i)}{2} \sum_{k=1}^K [\alpha_i^k]^2 + \lambda y_i^\top \alpha_i \qquad (21)
\end{aligned}$$

Since the objective function in (21) is convex in λ and concave in α^i, therefore according von Newman's lemma, switching minimization with maximization will not affect the final solution. Thus, we could obtain the solution by maximizing over α, i.e.,

$$\alpha_i^k = \pi_{[0,C]} \left(\frac{1 + \lambda y_i^k - \frac{1}{2} y_i^k f_k^{-i}(x_i)}{k(x_i, x_i)} \right)$$

where $\pi_{[0,C]}(x)$ projects x onto the region $[0, C]$. To compute λ, we aim to solve the following equation

$$\sum_{k=1}^K y_i^k \pi_{[0,C]} \left(\frac{1 + \lambda y_i^k - \frac{1}{2} y_i^k f_k^{-i}(x_i)}{k(x_i, x_i)} \right) = 0 \qquad (22)$$

Since when $y_i^k = 1$, the projection in Eq 22 is $\pi_{[0,C]}$ and when $y_i^k = -1$, it is $\pi_{[-C,0]}$, we could represent $y_i^k \pi_{[0,C]} \left(\frac{1+\lambda y_i^k - \frac{1}{2} y_i^k f_k^{-i}(x_i)}{k(x_i,x_i)} \right)$ by $h(\frac{y_i^k + \lambda - \frac{1}{2} f_k^{-i}(x_i)}{k(x_i,x_i)}, y_i^k C)$ where $h(x, y)$ is already defined in the theorem. Since $y_i^\top \alpha_i = 0$, we have the following equation for λ

$$g(\lambda) = \sum_{k=1}^K h \left(\frac{y_i^k + \lambda - \frac{1}{2} f_k^{-i}(x_i)}{k(x_i, x_i)}, y_i^k C \right) = 0 \qquad (23)$$