Multi-label Learning with Incomplete Class Assignments

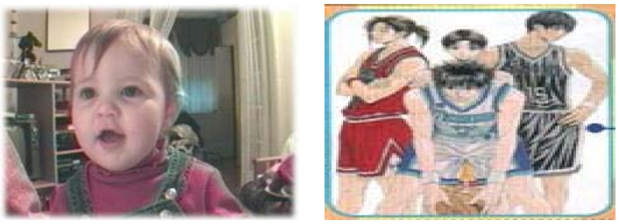
Serhat Selcuk Bucak* Rong Jin* Anil K. Jain*†
*Dept. of Computer Science & Eng., Michigan State University, East Lansing, MI, U.S.A.
†Dept. of Brain & Cognitive Eng., Korea University, Anam-dong, Seoul, Korea
{bucakser,rongjin,jain}@cse.msu.edu

Abstract

We consider a special type of multi-label learning where class assignments of training examples are incomplete. As an example, an instance whose true class assignment is (c_1, c_2, c_3) is only assigned to class c_1 when it is used as a training sample. We refer to this problem as ***multi-label learning with incomplete class assignment****. Incompletely labeled data is frequently encountered when the number of classes is very large (hundreds as in MIR Flickr dataset) or when there is a large ambiguity between classes (e.g., jet vs plane). In both cases, it is difficult for users to provide complete class assignments for objects. We propose a ranking based multi-label learning framework that explicitly addresses the challenge of learning from incompletely labeled data by exploiting the group lasso technique to combine the ranking errors. We present a learning algorithm that is empirically shown to be efficient for solving the related optimization problem. Our empirical study shows that the proposed framework is more effective than the state-of-the-art algorithms for multi-label learning in dealing with incompletely labeled data.*

1. Introduction

Multi-label learning is an important problem in machine learning, and has found applications in several computer vision problems (e.g., visual object recognition and automatic image annotation). Many algorithms have been developed for multi-label learning [3, 17, 29, 27, 10, 21, 13]. In this work, we consider the multi-label learning problem in which only a subset of the true class assignments is available for each training instance. As an example, an instance whose true class assignment is (c_1, c_2, c_3) is only presented with class c_1 when it is used for training. Our goal is to learn a multi-labeling model from the training examples with incomplete class assignments. We refer to this problem as **multi-label learning with incomplete class assignments**, and the training data as *incompletely labeled data*. Multi-label learning with incomplete class assignments is frequently encountered in automatic image annotation when the number of classes is very large, and it is only feasible for users to provide limited number of class labels for a given instance. Fig. 1 shows examples of annotated images from the ESP Game. We see some of the annotated words can cause ambiguity. For instance the keywords *baby*, *kid* and *boy* can be used interchangeable, and this can be given as an example of label ambiguity. Note that these annotations are generated by collapsing annotated words from multiple users. It is thus very likely that each individual user only provides incomplete annotation with a few keywords.



baby, *boy,* **child**, *eye, face, girl, hair, house,* **kid**, *mouth, nose, pink, smile*

anime, *ball, boy,* **cartoon**, **drawing**, *girl, group, hair, kid, man, people, play, red, sport*

Figure 1. Example images from ESP Game dataset and their annotations. The annonations highlighted by bold font, which are used to annotate the same concept/object in the corresponding images, are examples of label ambiguity problem.

It is important to distinguish the learning scenario studied in this work from the related ones in the previous studies: (i) *partial labeling* [23, 18] where for each training instance, only one of its class assignments is correct; (ii) *weakly labeled data* [24] where confidence score is computed for each assigned class label to facilitate the learning process; (iii) *harvesting weakly tagged* image databases [9] that focuses on removing false class assignments for training set; (iv) *partially labeled data* used in the image annotation study [14] that refers to the training images where only a subset of their segments are labeled; (v) *bandit multi-class learning* [19, 32] that focuses on multi-class learning, where each instance is only assigned to one class.

There is a rich body of literature on multi-label learning, ranging from the simple approaches that divide multi-label learning into a set of binary classification problems [5] to more sophisticated approaches that explicitly explore the correlation among classes [29, 27, 10, 21] and to multi-label ranking approaches [13, 4, 3, 6, 27, 2] that cast multi-label learning into a ranking problem. But none of these approaches addresses the challenge of multi-label learning from incompletely labeled data, which is a more realistic scenario. To this end, we present a multi-label learning framework based on the idea of multi-label ranking [13, 6, 27, 2]. Unlike the classification approaches that make a binary decision about the class assignment for a given instance, multi-label ranking ranks classes for the given instance such that the "true" classes are ranked before the other classes. By avoiding a binary decision, multi-label ranking is usually more robust than the classification approaches, particularly when the number of classes is very large [29, 2]. In order to handle the problem of incomplete class assignment, we extend multi-label ranking by exploiting the group lasso technique [33] to combine the errors in ranking the assigned classes against the unassigned classes. As will be seen in later discussion, by using group lasso to combine ranking errors, the proposed framework may be able to automatically detect the missed class assignment and consequentially improve the classification accuracy.

We present an efficient learning algorithm for the proposed framework. This is important since the naive implementation of multi-label ranking will result in a pairwise comparison between every pair of classes, making it difficult to scale to a large number of classes and training instances. Our empirical studies on three benchmark datasets for image annotation and visual object recognition indicate that (i) our framework is robust to missing class assignments compared to the state-of-the-art approaches for multi-label learning, and (ii) the proposed approach is computationally efficient and scales well to the number of training examples.

2. A Framework for Multi-label Learning from Incompletely Labeled Data

In order to handle incompletely labeled data, we consider exploring the group lasso regularizer when estimating the error in ranking the assigned classes against the unassigned ones. The key idea is to selectively penalize the ranking errors. To facilitate our discussion, we consider an instance x that is assigned to classes $c_1, \ldots, c_a$. Consequently, classes $c_{a+1}, \ldots, c_m$ are remained as the unassigned classes for x. If example x is fully labeled, following [2], the ranking error for given classification functions $f_k(x), k \in [m]$ is expressed as

$$\sum_{k=1}^{a} \sum_{l=a+1}^{m} \max(0, f_l(x) - f_k(x) + 1) \tag{1}$$

However, given the data is only partially labeled, some of the unassigned class labels may indeed be the true classes, and the above loss function for x may overestimate the classification error. To address this issue, we introduce a slack variable, denoted by $\varepsilon_{k,l}$, to account for the error of ranking an unassigned class l before the assigned class k. This introduces the following constraint

$$\varepsilon_{k,l} + f_k(x) \geq 1 + f_l(x) \tag{2}$$

Now, instead of adding all the errors together for example x, i.e., $\sum_{k=1}^{a} \sum_{l=a+1}^{m} \varepsilon_{k,l}$, we combine the ranking errors $\varepsilon_{k,l}$ via a group lasso regularizer, i.e.,

$$\sum_{l=a+1}^{m} \sqrt{\sum_{k=1}^{a} \varepsilon_{k,l}^2} \tag{3}$$

The motivation of using the group lasso for aggregating ranking errors is two fold: first, as stated in the general theory, group lasso is able to select a group of variables, which in our case, is to select the group of ranking errors $\{\varepsilon_{k,l}, k = 1, \ldots, a\}$ for each unassigned class c_l. In particular, an unassigned class c_l is likely to be a missing class assignment for example x when many of its ranking errors $\{\varepsilon_{k,l}\}_{k=1}^{a}$ are non-zero, which coincides with the criterion of group selection by group lasso. Thus, by using the group lasso regularizer, we may be able to decide which unassigned class is indeed the missing correct class assignment. Second, group lasso usually results in a sparse solution in which most of the group variables are zero and only a small number of groups are assigned non-zero values. In our case, the sparse solution implies that most of the unassigned classes for x are indeed correct, and only a few unassigned classes are the true class assignments for x that are missed by manual labeling.

Let $x_1, \ldots, x_n$ be the collection of training instances that are labeled by $Y_1, \ldots, Y_n$, where each $Y_i \subset \mathcal{Y}$. For the convenience of presentation, we represent each class assignment Y_i by a binary vector $y^i = (y_1^i, \ldots, y_m^i) \in \{-1, +1\}^m$, where $y_k^i = +1$ if $k \in Y_i$ and $y_k^i = -1$ if $k \notin Y_i$. Using the group lasso regularizer described above, we have the following optimization problem:

$$\min_{f_k \in \mathcal{H}_\kappa} \frac{1}{2} \sum_{k=1}^{m} |f_k|_{\mathcal{H}_\kappa}^2 + C \sum_{i=1}^{n} \sum_{l \notin Y_i} \sqrt{\sum_{k \in Y_i} \ell^2(f_k(x_i) - f_l(x_i))} \tag{4}$$

where $\ell(z) = \max(0, 1 - z)$ is the hinge loss function that assesses the error in ranking two classes c_k and c_l. In the next section, we discuss the strategy for efficiently optimizing Eq. (4).

3. Optimization Algorithm

First, we have the following representer theorem for $f(x)$ that optimizes Eq. (4).

Theorem 1 *The optimal solution to Eq. (4) admits the following expression for $f(x)$, i.e.,*

$$f_k(x) = \sum_{i=1}^{n} y_k^i \alpha_k^i \kappa(x, x_i), \quad k = 1, \ldots, m$$

where $\alpha_k^i, i = 1, \ldots, n$ are the combination weights.

It is straightforward to verify the above representer theorem. Next, in order to solve Eq. (4) efficiently, we aim to linearize the objective function in Eq. (4) by using the following lemma.

Lemma 1 $\sum_{l=a+1}^{m} \sqrt{\sum_{k=1}^{a} \ell^2(f_k(x_i) - f_l(x_i))}$ *is equivalent to the following expression:*

$$\max_{\gamma^i \in \mathbb{R}^{a \times (m-a)}} \left\{ \sum_{l=a+1}^{m} \sum_{k=1}^{a} \gamma_{k,l}^i \ell(f_k(x_i) - f_l(x_i)) \right\} \tag{5}$$

$$\text{s.t.} \quad \max_{1 \leq l \leq m-a} |\gamma_{\cdot,l}^i|_2 \leq 1 \tag{6}$$

where $\gamma_{\cdot,l}$ stands for the lth column vector of matrix γ^i.

Lemma 1 follows directly from the fact that $\sum_{l=a+1}^{m} \sqrt{\sum_{k=1}^{a} \ell^2(f_k(x_i) - f_l(x_i))}$ is a $L_{1,2}$ norm of the loss function $\ell(f_k(x) - f_l(x))$ and the dual norm of $L_{1,2}$ is $L_{\infty,2}$.

Using lemma 1, we turn Eq. (4) into a convex-concave optimization problem as revealed in the following theorem.

Theorem 2 *The problem in Eq. (4) is equivalent to the following convex-concave optimization problem*

$$\max_{\{\gamma^i \in \Delta_i\}_{i=1}^n} \min_{\{f_k \in \mathcal{H}_\kappa\}_{k=1}^m} L = \frac{1}{2} \sum_{k=1}^{m} |f_k|_{\mathcal{H}_\kappa}^2 \tag{7}$$

$$+ C \sum_{i=1}^{n} \sum_{l \notin Y_i} \sum_{k \in Y_i} \gamma_{k,l}^i \ell(f_k(x_i) - f_l(x_i))$$

where $\gamma^i = [\gamma_{k,l}^i]_{m \times m}$ and

$$\Delta_i = \left\{ \gamma^i \in \mathbb{R}^{m \times m} : \begin{array}{c} \gamma_{k,l}^i \geq 0, k, l = 1, \ldots, m, \\ \gamma_{k,l}^i = 0 \text{ if } l \in Y_i \text{ or } k \notin Y_i, \\ \max\limits_{1 \leq l \leq m} |\gamma_{\cdot,l}^i|_2 \leq 1 \end{array} \right\}$$

The above theorem follows by directly plugging the result of Lemma 1 into Eq. (4). As indicated by the above theorem, the introduction of the group lasso is equivalent to introducing a different weight $\gamma_{k,l}^i$ for each comparison between an assigned class and an unassigned class. It is the introduction of these weights that allows us to determine which unassigned class is missed in the user's annotation.

Theorem 3 *The optimal solution $f(x)$ to Eq. (7) can be expressed as follows:*

$$f_k(x) = \sum_{i=1}^{n} y_k^i \alpha_k^i \kappa(x, x_i)$$

where $\alpha^i = (\alpha_1^i, \ldots, \alpha_m^i)^\top, i = 1 \ldots n$ is the optimal solution to the following optimization problem:

$$\max_{\{\alpha^i \in \Omega_i\}_{i=1}^n} \sum_{k=1}^{m} \left(\sum_{i=1}^{n} \alpha_k^i - \sum_{i,j=1}^{n} \alpha_k^i \alpha_k^j y_k^i y_k^j K_{i,j} \right) \tag{8}$$

where

$$\Omega_i = \left\{ \alpha^i \in \mathbb{R}^m : \exists \gamma^i \in \Delta_i \textit{ s. t. } \alpha^i = C\gamma^i \mathbf{1} + C[\gamma^i]^\top \mathbf{1} \right\}$$

The proof of this theorem can be found in Appendix A. Note that although the objective function in Eq. (8) is similar to that of SVM, it is the constraints specified in domain Ω_i that makes this problem computationally more challenging.

In order to efficiently solve Eq. (8), we consider the block coordinate descent method. In particular, we aim to optimize α^i with the other $\{\alpha^j, j \neq i\}$ being fixed. Without a loss of generality, we assume that example x_i is assigned to the first a classes and is not assigned to the remaining $b = m - a$ classes. For the convenience of presentation, we drop the index i and write α^i as α. We thus have the following optimization problem for α^i.

$$\max_{\alpha \in \Omega} \sum_{k=1}^{m} \alpha_k - K_{i,i} \sum_{k=1}^{m} \alpha_k^2 - 2 \sum_{k=1}^{m} y_k \alpha_k \sum_{j \neq i} \alpha_k^j y_k^j K_{i,j} \tag{9}$$

where Ω is defined as

$$\Omega = \left\{ \alpha \in \mathbb{R}^m : \exists \gamma \in \mathbb{R}_+^{a \times b}, |\gamma_{\cdot,l}|_2 \leq 1, l \in [b] \right.$$
$$\left. \text{s.t. } \alpha_{1:a} = C\gamma \mathbf{1}_b, \ \alpha_{a+1:a+b} = C\gamma^\top \mathbf{1}_a \right\}$$

In the above, we use the notation $\alpha_{i:j} = (\alpha_i, \ldots, \alpha_j)$ to represent a subset of vector α whose index ranges from i to j. $\mathbf{1}_a$ represents a vector of a dimensions with all its elements being one. We now aim to simplify the problem in Eq. (9). First, we have for any $\alpha \in \Omega$

$$\sum_{k=1}^{m} \alpha_k = 2C(\mathbf{1}_a^\top \gamma \mathbf{1}_b) \tag{10}$$

Second, we have

$$\sum_{k=1}^{m} \alpha_k^2 = \sum_{k=1}^{a} \alpha_k^2 + \sum_{k=a+1}^{a+b} \alpha_k^2 = C^2 \left(\mathbf{1}_b^\top \gamma^\top \gamma \mathbf{1}_b + \mathbf{1}_a^\top \gamma \gamma^\top \mathbf{1}_a \right) \tag{11}$$

To simplify the last term in Eq. (9), we define

$$f_k^{-i}(x_i) = y_k \sum_{j \neq i} \alpha_k^j y_k^j \kappa(x_i, x_j) \tag{12}$$

and vector $\mathbf{f}^{-i} = (f_1^{-i}(x_i), \ldots, f_i^{-i}(x_i)) = (\mathbf{f}_a^{-i}, \mathbf{f}_b^{-i})$. Using these notations, the third term in Eq. (9) becomes

$$\sum_{k=1}^{m} \alpha_k f_k^{-i}(x_i) = \alpha^\top \mathbf{f}^{-i} = C\text{tr}\left(\left(\mathbf{1}_b[\mathbf{f}_a^{-i}]^\top + \mathbf{f}_b^{-i}\mathbf{1}_a^\top]\right)\gamma\right) \quad (13)$$

Thus, we have the following optimization problem to solve

$$\max_{\gamma\in\Delta} \mathbf{1}_a^\top \gamma \mathbf{1}_b - \frac{1}{2} C K_{i,i} \left(\mathbf{1}_b^\top \gamma^\top \gamma \mathbf{1}_b + \mathbf{1}_a^\top \gamma\gamma^\top \mathbf{1}_a\right) \quad (14)$$
$$-\text{tr}\left(\left(\mathbf{f}_b^{-i}\mathbf{1}_a^\top + \mathbf{1}_b[\mathbf{f}_a^{-i}]^\top\right)\gamma\right)$$

where $\Delta = \{\gamma \in \mathbb{R}_+^{a\times b} : |\gamma_{\cdot,l}|_2 \leq 1, l = 1, \ldots, b\}$. The problem in Eq. (14) is indeed a Second Order Cone Programming (SOCP) problem [1]. Although a SOCP problem can be solved by a standard tool like SeDuMi [28], it can still be computationally expensive to solve a large-scale SOCP problem. We thus further simplify Eq. (14) by the following approximation

$$\mathbf{1}_b^\top \gamma^\top \gamma \mathbf{1}_b + \mathbf{1}_a^\top \gamma\gamma^\top \mathbf{1}_a \approx \eta\text{tr}(\gamma^\top\gamma + \gamma\gamma^\top) = 2\eta\text{tr}(\gamma^\top\gamma) \quad (15)$$

where $\eta > 1$ is a parameter introduced for approximation. Using the approximation in Eq. (15), we have

$$\max_{\gamma\in\Delta} \mathbf{1}_a^\top \gamma \mathbf{1}_b - C K_{i,i}\eta\text{tr}(\gamma^\top\gamma) - \text{tr}\left(\left(\mathbf{f}_b^{-i}\mathbf{1}_a^\top + \mathbf{1}_b[\mathbf{f}_a^{-i}]^\top\right)\gamma\right) \quad (16)$$

Define

$$\left((\mathbf{1}_b\mathbf{1}_a^\top) - \mathbf{f}_b^{-i}\mathbf{1}_a^\top - \mathbf{1}_b[\mathbf{f}_a^{-i}]^\top\right)^\top = 2\mathbf{H} = (2\mathbf{h}_1, \ldots, 2\mathbf{h}_b). \quad (17)$$

Lemma 2 shows a closed form solution to Eq. (16).

Lemma 2 *The optimal solution to Eq. (16) is*

$$\gamma_{\cdot,s} = \frac{\pi_{\mathcal{G}}(\mathbf{h}_s)}{|\pi_{\mathcal{G}}(\mathbf{h}_s)|_2} \min\left(1, \frac{|\pi_{\mathcal{G}}(\mathbf{h}_s)|_2}{CK_{i,i}\eta}\right), \quad s = 1, \ldots, b, \quad (18)$$

where $\mathcal{G} = \{\mathbf{z} : \mathbf{z} \in \mathbb{R}_+^a\}$ and $\pi_{\mathcal{G}}(\mathbf{h})$ projects vector $\mathbf{h}$ into the domain $\mathcal{G}$.

The proof of this lemma can be found in Appendix B.

4. Experimental Results

To study the problem of incomplete class assignment, we evaluate the proposed approach on the image annotation and visual object recognition tasks, which are usually treated as special cases of multi-label learning with each image being annotated by multiple keywords/objects. The focus of this experiment is to verify the effectiveness of the proposed approach in handling incompletely labeled data.

Datasets. Two multi-labeled datasets for automatic image annotation are used in our study: ESP Game [31] and MIR Flickr [16] datasets. The number of classes is 457 for MIR Flickr dataset and 268 for ESP Game dataset. We remove the images that are assigned to fewer than three classes form MIR Flickr and images that are assigned to fewer than five classes form ESP Game dataset. The dataset statistics are given in Table 1. For both datasets, we randomly take 75% of the examples to form the training set and use the rest for testing. We repeat the experiments ten times, each with a random partitioning of data for training and testing, and report the performance averaged over the ten trials. The bag-of-words model based on dense sampling, provided by [11] and [12], is used for image representation. To simulate the situation of incomplete class assignment, we conduct experiments in four different settings for ESP Game and MIR Flickr datasets. In the first setting, termed **case-1**, there is no missing class assignment for any training image. In the next three settings, termed **case-2**, **case-3**, and **case-4**, for each training image, we randomly choose 20%, 40%, and 60% of the assigned class labels, respectively, and remove them from the training data.

Table 1. Dataset statistics

	# samples	# classes	avg. label/img	avg img/label
VOC07	9963	20	1.47	729.85
ESP Game	10457	268	6.41	250.29
MIR Flickr	10199	457	5.30	118.43

In addition to the two multi-labeled datasets, we also include the VOC2007 dataset to show that proposed algorithm yields comparable results with the state-of-the-art methods for visual object recognition. The majority of the images in VOC2007 dataset are labeled by a single class, as shown in Table 1. This property does not make VOC2007 an ideal dataset for evaluating multi-label learning algorithms. Nevertheless, the performance over VOC2007 dataset will allow us to examine if the proposed algorithm is effective for visual object recognition.

Evaluation metric. Since our study is focused on multi-label ranking, we evaluate the results of ranked class labels by following the protocol in [2]. In particular, we first rank all the classes for each test image in the descending order of their scores; we then vary the number of predicted classes from 1 to the total number of classes, and compute the ROC curve by calculating true positive rate (TPR) and false positive rate (FPR) for each number of predicted classes. We finally compute the Area Under ROC curve (AUC) as the final evaluation metric. Note that our approach for computing AUC is different than [8] which computes AUC by ranking the output scores for each class.

Baseline methods. We compare the proposed method to three baseline methods: (i) **LIBSVM**: LIBSVM imple-

Table 2. AUC results for VOC2007 dataset

	MLR-L1	LIBSVM	LIBSVM+platt	MLR-GL
AUC	91.07 ± 0.46	90.70 ± 0.27	90.47 ± 0.29	90.97 ± 0.32

Table 3. AUC results for ESP Game dataset. The results are highlighted when they are significantly better than the competing algorithms according to the paired t-test.

	case-1	case-2	case-3	case-4
MLR-GL	84.76 ± 0.24	84.11 ±0.11	**83.47 ±0.14**	**82.65 ±0.16**
LIBSVM	79.99± 0.24	77.90± 0.28	75.21 ±0.27	71.68 ±0.68
LIBSVM+Platt	82.67± 0.24	81.88 ±0.35	80.76 ±0.25	78.92 ±0.38
MLR-L1	84.80 ±0.27	83.83 ±0.34	82.79 ± 0.30	80.18 ± 0.81

Table 4. AUC results for MIR Flickr dataset. The results are highlighted when they are significantly better than the competing methods according to the paired t-test.

	case-1	case-2	case-3	case-4
MLR-GL	**76.24 ± 0.12**	**75.70 ± 0.13**	**75.04 ± 0.08**	**74.05 ± 0.13**
LIBSVM	70.18 ± 0.24	69.05 ± 0.41	67.60 ± 0.42	65.69 ± 0.41
LIBSVM+Platt	68.67 ± 0.27	67.57 ± 0.51	66.11 ± 0.49	64.31 ± 0.37
MLR-L1	73.41 ± 0.51	72.67 ± 0.24	71.70 ± 0.60	69.05 ± 1.25

mentation of One-versus-All (OvA) SVM classifier, which is widely used for visual object recognition [22, 7] and was shown to outperform multi-class SVM [15]; (ii) **LIBSVM+Platt**: it applies Platt's method to convert SVM scores to posterior probabilities [26]. This conversion makes it easy to compare the output scores of different SVM classifiers, leading to better performance for multi-label ranking; (iii) **MLR-L1**: an efficient multi-label ranking algorithm [2], which is shown to outperform a number of multi-label learning algorithms. We refer to the proposed method as **MLR-GL**[1].

We use chi-squared kernel $K(x, y) = \exp(-d(x, y)/\sigma)$, where $d(x, y) = \chi^2(x, y)$, for VOC2007 dataset, which gave good performance in [20]. A modified chi-squared kernel with $d(x, y) = |x - y|_2^2/|x + y|_2^2$, is used for ESP GAME and MIR Flickr datasets because it yields significantly better performance than the standard version. The optimal values for parameters C and η are found by cross validation. σ in is set to be chi-squared kernel is chosen as the mean of the pair-wise distances $d(x, y)$ [30].

Experimental Results We first compare the baselines on VOC2007 dataset. According to [22, 7], SVM classifier with chi-squared kernel, one of the baselines (LIBSVM) used in our study, yields comparable performance with the state-of-art methods in PACAL VOC evaluation. According to Table 2, the proposed algorithm yields similar performance as the LIBSVM method, indicating that the proposed method is effective for visual object recognition.

Table 3 shows the results for ESP Game dataset. First, we observe that LIBSVM+Platt significantly improves the performance of LIBSVM in all four settings. This is consistent with [25], where the conversion procedure makes the outputs from different SVM classifiers more comparable and consequently leads to better performance for multi-label ranking. On the other hand, both LIBSVM and LIBSVM+Platt are outperformed by the other two multi-label learning methods, indicating the importance of developing multi-label ranking methods for multi-label learning.

Second, we observe a significant decrease in classification accuracy for all the four methods when moving from case-1 to case-4, indicating that the missing class assignment could greatly affect the classification performance. On the other hand, compared to the three baseline methods, the proposed method MLR-GL is more resilient to the missing class labels: it only experiences a 2% drop in AUC metric when 60% of the assigned class labels are removed (case-4), while the other three methods suffer from 4% to 8% loss in AUC. This result indicates the robustness of the proposed method in handling missing class assignments.

In Figures 2 and 3, we provide sample images from the the ESP Game dataset for setting case-4 where 60% of the assigned class labels are missing from the training images. Figure 2 shows how different methods perfom in finding the missing true labels for training examples, where only the underlined true labels. We observe that MLR-GL is able to find more missing labels than the other baselines. Unlike the baselines, MLR-GL does not always rank the labels provided examples at top, In contrast, it ranks some keywords which are initially labeled as irrelevant higher than the relevant keywords assigned to training instances. This is why the proposed method outperforms the baselines in this task. Figure 3 shows examples of annotations generated for test images. These examples confirm that the proposed method gives better annotation results than the baseline methods.

Finally, we report the results on MIR Flickr data in Table 4. We note that MIR Flickr dataset is more challenging than ESP Game dataset because it has a larger number of classes and a smaller number of labeled images per class. This fact is clearly reflected in Table 4, where the best AUC of MIR Flickr dataset is below 77%, while the best AUC results for VOC2007 and ESP Game are 91.07% and 84.80%, respectively. Similar to ESP Game dataset, we observe (i) a significant drop in AUC metric for all the methods when some class assignments are missing from training examples, and (ii) MLR-GL experiences the least degradation in AUC compared to the three baseline methods. We also noticed that unlike ESP Game dataset, LIBSVM+Platt is outperformed by LIBSVM for MIR Flickr dataset, indicating that the conversion procedure does not work for this dataset.

Based on the above results, we conclude that the proposed method for multi-label learning (i) is effective for visual object recognition and automatic image annotation, and (ii) is more effective in handling incompletely labeled data than the state-of-the-art methods for multi-label learning.

Running time. Table 5 gives the average training time for the three methods [2] for ESP Game dataset. In this exper-

[1] Codes are available at http://www.cse.msu.edu/∼bucakser

[2] LIBSVM+Platt has almost the same training time as LIBSVM

Table 5. Training time (seconds) for different multi-label learning algorithms with varied numbers (n) of training examples.

	n=1000	n=3000	n=5000	n=7000
MLR-GL	13.55	180.06	590.31	1168.60
LIBSVM	17.03	165.70	559.40	1182.21
MLR-L1	43.26	193.18	533.44	1118.10

iment, we vary the number of training examples from 1000 to 7000. The proposed method, MLR-GL, is implemented in C; the C++ implementations of LIBSVM and MLR-L1 provided by the authors are used in our study. Overall, we observe that all the methods have similar running time. The computational complexity of MLR-L1 and MLR-GL per iteration is $O(n^2m)$, where n is the number of training examples and m is the number of classes. Note that the overhead of computing the kernel matrix is not included in this study because it is shared by all the methods.

5. Future Work

In future work, we are planning to analyse the approximation we make to the SOCP problem in detail. We also plan to extend our work to the scenario where not only some of the "true" class assignments are missing, but some of the class labels are incorrectly assigned to the training instances. This scenario often encountered in the problem of image tagging [12], where correct tags are often missed from the training data and incorrect tags are sometimes are given to the training examples. This is clearly a more challenging problem in which we need to address the uncertainty arising from missing class assignment as well as from noisy class assignments.

6. Acknowledgements

This work was supported in part by National Science Foundation (IIS-0643494), US Army Research (ARO Award W911NF-08-010403) and Office of Naval Research (ONR N00014-09-1-0663). 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, ARO, and ONR. Part of Anil Jain's research was supported by WCU (World Class University) program through the National Research Foundation of Korea funded by the Ministry of Education, Science and Technology (R31-2008-000-10008-0).

Appendix A: Proof of Theorem 3

Proof 1 *We can rewrite $\ell(z)$ as*

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

Using the above expression for $\ell(z)$, the objection function can be rewritten as

$$\min_{f_k\in\mathcal{H}_K} \max_{\gamma^i_{k,l}\in\Delta_i} \max_{\beta^i_{k,l}\in[0,1]} \frac{1}{2}\sum_{k=1}^m |f_k|^2_{\mathcal{H}_K} \quad (19)$$

$$+C\sum_{i=1}^n \sum_{k\in Y^i} \sum_{l\notin Y^i} \gamma^i_{k,l}\beta^i_{k,l}(1 - f_k(x_i) + f_l(x_i))$$

The problem now becomes a convex-concave optimization. By defining new variable $\Gamma^i_{k,l}$ as

$$\Gamma^i_{k,l} = \gamma^i_{k,l}\beta^i_{k,l} + \gamma^i_{l,k}\beta^i_{l,k},$$

we rewrite Eq. (20) as

$$\min_{f_k\in\mathcal{H}_K} \max_{\Gamma^i_{k,l}\in\Delta_i} \frac{1}{2}\sum_{k=1}^m |f_k|^2_{\mathcal{H}_K} \quad (20)$$

$$+\sum_{i=1}^n \sum_{k,l=1}^m \Gamma^i_{k,l}\left(1 - f_k(x_i) + f_l(x_i)\right)$$

Since Eq. (21) is a convex-concave optimization problem, according to von Newman's lemma, we can switch minimization with maximization. By taking the minimization with respect to f_k, we have

$$f_k(x) = C\sum_{i=1}^n \left(\sum_{l=1}^m \Gamma^i_{k,l} - \sum_{l=1}^m \Gamma^i_{l,k}\right)\kappa(x, x_i) \quad (21)$$

According to the definition of Δ_i, $\Gamma^i_{k,l}$ is nonzero only when $k \in Y^i$ (i.e., $y^i_k = 1$) and $l \notin Y^i$ (i.e., $y^i_k = -1$). We thus can rewrite $f_k(x)$ in Eq. (21) as

$$f_k(x) = C\sum_{i=1}^n \left(\sum_{l=1}^m \Gamma^i_{k,l} + \sum_{l=1}^m \Gamma^i_{l,k}\right) y^k_i \kappa(x, x_i)$$

By defining $\alpha^i_k = \sum_{l=1}^m \Gamma^i_{k,l} + \sum_{l=1}^m \Gamma^i_{l,k}$, we have the result in the theorem.

Appendix B: Proof of Lemma 2

Proof 2 *First, using the notation of $\mathbf{h}_k$, we rewrite the objective function in Eq. (16) as*

$$\max_{\gamma\in\Delta} -CK_{i,i}\eta\sum_{s=1}^b |\gamma_{\cdot,s}|_2^2 + 2\sum_{s=1}^b \mathbf{h}_s^\top\gamma_{\cdot,s}$$

Since all $\gamma_{\cdot,s}, s = 1,\dots,b$ are decoupled in both the domain Δ and the objective function, we can decompose the above problem into b independent optimization problems,

$$\max_{\gamma_{\cdot,s}\in\mathbb{R}^a_+} \left\{-CK_{i,i}\eta|\gamma_{\cdot,s}|_2^2 + 2\mathbf{h}_s^\top\gamma_{\cdot,s} : |\gamma_{\cdot,s}|_2 \le 1\right\} \quad , (22)$$

where $s = 1, \ldots, b$. For each independent optimization problem, we introduce a Lagrangian multiplier $\lambda_s \geq 0$ for constraint $|\gamma_{\cdot,s}|_2 \leq 1$, and have

$$\min_{\lambda_s \geq 0} \max_{\gamma_{\cdot,s} \in \mathbb{R}^a_+} -(CK_{i,i}\eta + \lambda_s)|\gamma_{\cdot,s}|_2^2 + 2\mathbf{h}_s^\top \gamma_{\cdot,s} + \lambda_s$$

The optimal solution to the maximization of γ is

$$\gamma_{\cdot,s} = \pi_{\mathcal{G}}\left(\frac{\mathbf{h}_s}{\lambda_s + CK_{i,i}\eta}\right)$$

In order to decide the value for λ_s, we use the complementary slackness condition, i.e., $\lambda_s(|\gamma_{\cdot,s}|_2^2 - 1) = 0$. There are two cases: $\lambda = 0$ implies $|\gamma_{\cdot,s}|_2^2 \leq 1$, and $\lambda > 0$ implies $|\gamma_{\cdot,s}|_2^2 = 1$. This leads to the result stated in the Lemma.

References

[1] F. Alizadeh and D. Goldfarb. Rcv1: A new benchmark collection for text categorization research. *Mathematical Programming*, 95:3–51, 2003.

[2] S. Bucak, P. K. Mallapragada, R. Jin, and A. K. Jain. Efficient multi-label ranking for multi-class learning: Application to object recognition. In *Proc. of ICCV*, 2009.

[3] G. Chen, Y. Song, F. Wang, and C. Zhang. Semi-supervised multi-label learning by solving a sylvester equation. In *Proc. SDM*, pages 410–419, 2008.

[4] 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.

[5] K. Crammer and Y. Singer. On the learnability and design of output codes for multiclass problems. *Machine Learning*, 47(2):201–233, 2002.

[6] O. Dekel, C. Manning, and Y. Singer. Log-linear models for label ranking. In *Proc. of NIPS*, pages 497–504, 2004.

[7] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html.

[8] M. Everingham, A. Zisserman, C. K. I. Williams, and L. Van Gool. The PASCAL Visual Object Classes Challenge 2006 (VOC2006) Results. http://www.pascal-network.org/challenges/VOC/voc2006/results.pdf.

[9] J. Fan, Y. Shen, N. Zhou, and Y. Gao. Harvesting large-scale weakly-tagged image databases from the web. In *Proc. of CVPR*, 2010.

[10] N. Ghamrawi and A. McCallum. Collective multi-label classification. In *Proc. of CIKM*, 2005.

[11] M. Guillaumin, T. Mensink, J. Verbeek, and C. Schmid. Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation. In *Proc. of ICCV*, 2009.

[12] M. Guillaumin, J. Verbeek, and C. Schmid. Multimodal semi-supervised learning for image classification. In *Proc. of CVPR*, 2010.

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

[14] X. He and R. S. Zemel. Learning hybrid models for image annotation with partially labeled data. In *Proc. of NIPS*, 2008.

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

[16] M. J. Huiskes and M. S. Lew. The mir flickr retrieval evaluation. In *Proc. of ACM Int. Conf. on Multimedia Information Retrieval*, 2008.

[17] S. Ji, L. Tang, S. Yu, and J. Ye. Extracting shared subspace for multi-label classification. In *Proc. of SIGKDD*, 2008.

[18] R. Jin and Z. Ghahramani. Learning with multiple labels. In *Proc. of NIPS*, 2002.

[19] S. M. Kakade, S. Shalev-Shwartz, and A. Tewari. Efficient bandit algorithms for online multiclass prediction. In *Proc. of ICML*, 2008.

[20] F. Li, J. Carreira, and C. Sminchisescu. Object recognition as ranking holistic figure-ground hypotheses. In *Proc. of CVPR*, 2010.

[21] Y. Liu, R. Jin, and L. Yang. Semi-supervised multi-label learning by constrained non-negative matrix factorization. In *Proc. of AAAI*, 2006.

[22] M. Marszałek and C. Schmid. Semantic hierarchies for visual object recognition. In *Proc. of CVPR*.

[23] N. Nguyen and R. Caruana. Classification with partial labels. In *Proc. of KDD*, 2008.

[24] A. Pentland. Expectation maximization for weakly labeled data. In *Proc. of ICML*, 2001.

[25] M. Petrovskiy. Paired comparisons method for solving multi-label learning problem. In *Proc. of Conf. of Hybrid Intelligent Systems*, 2006.

[26] J. C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in Large Margin Classifiers*, pages 61–74. MIT Press, 1999.

[27] 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.

[28] J. F. Sturm. Using sedumi 1. 02, a matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11-12:625–653, 1999.

[29] N. Ueda and K. Saito. Parametric mixture models for multi-labeled text. In *Proc. of NIPS*, 2002.

[30] M. Varma and D. Ray. Learning the discriminative power-invariance trade-off. In *Proc. of ICCV*, 2007.

[31] L. von Ahn and L. Dabbish. Labeling images with a computer game. In *Proc. of the Conf. on Human Factors in Computing Systems*, 2004.

[32] S. Wang, R. Jin, and H. Valizadegan. A potential-based framework for online milti-class learning with partial feedback. In *Proc. of AISTATS*, 2010.

[33] M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *J. Royal. Statist. Soc B.*, 68:49–67, 2006.

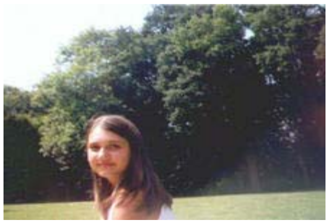
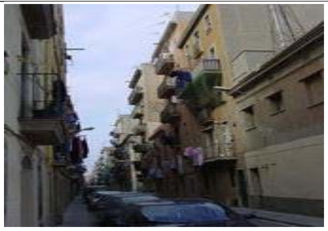
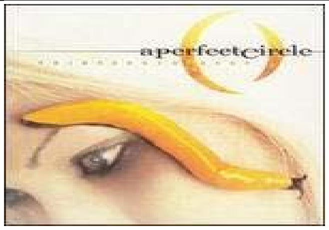
Images			
Labels	**brown girl grass green** **hair picture smile tree**	**blue, building car city cloud** **sky street white window**	**blonde circle eye face girl** **hair head woman yellow**
MLR-GL	*man black* **green** *people white red* *woman* **tree** *blue sky* **girl hair** **picture grass brown** *water light yellow* *old hat face* **smile** *house shirt eye*	**white** *man* **sky blue** *green red black* *woman water* **window** *tree people grass* *hair picture house yellow brown girl* **cloud building** *mountain smile face* **car**	*white man blue black red* **woman** **hair** *green sky* **yellow** *picture* **face** *girl brown people* **circle eye** water tree *smile hand hat old pink cartoon*
LIBSVM+Platt	**girl green** *blue black face hair woman* *people white glasses man group* **tree** **grass** *sky light pink chinese eye* *red plant dress hand flower forest*	**window city** *black hair man* **white** *water yellow smile chinese line tree* **sky** *lake mountain pink* **blue** *computer* *wood green table woman boy house hat*	**circle head** *black* **hair woman** *white* *hand book picture mountain line pink* *rock teeth photo square boy couple* *word old gray plant sea music ocean*
LIBSVM	**green girl** *space drink sky point face* *face woman shop metal family pot machine* *machine light truck forest star guy sit* *glasses white night* **hair** *black usa*	**city window** *metal truck* **car** *ball lake* *lake* **building** *room fly line wing roof* *water website mountain road helmet* **white** *chinese tent chair pink silver small*	**circle head** *room teeth metal ice black plant* *silver white* **hair** *hand shop book* *brick wall airplane bird horse plate* *flower photo music word pink*
MLR-L1	**green girl** *black* **tree** *people light* **hair** *man white metal dark band leaf* *star glasses sky space woman red* *night truck face street pot group*	**window city** *black* **sky** *water metal* *mountain pink wing* **building car** *hair* *boy computer lake truck insect person* *roof room man tree silver road ocean*	**circle head hair** *square ocean metal colors* *pink boy sea insect white black* *suit hand leaf line ball red* *old chart bird paper mountain silver*

Figure 2. Examples of training images from the ESP Game dataset with true labels and annotations generated by different multi-label learning methods. Only the underlined true labels are provided to the methods for training. For each method, the correct (returned) keywords are highlihted by bold font whereas the incorrect ones are highlighted by italic font.

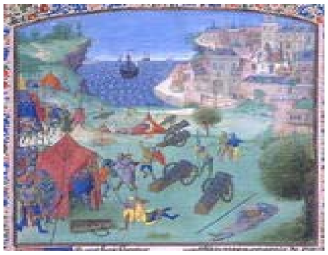
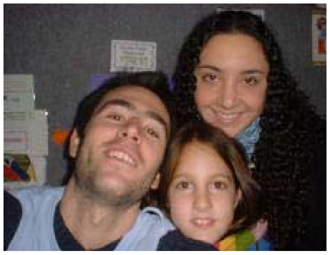
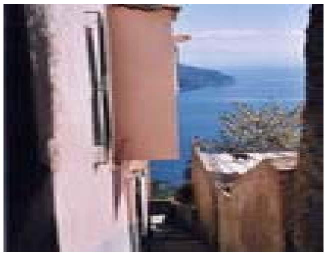
Images			
Labels	**tree water black picture** **drawing sea art blue** **boat green city**	**man woman people hair** **girl picture smile group** **photo kid family**	**sky tree water white** **house window wood sea** **ocean cloud blue door**
MLR-GL	*man white* **black** *woman people* **blue green** *red* **tree** *girl* *sky* **water** *hair* **picture** *old* *brown grass yellow face mountain*	**man woman** *black white* **people** *blue green red* **girl** *tree* **hair** *sky water* **picture** *old* *brown face yellow grass* **smile**	*man black* **white** *woman* **blue** *people green red girl* **tree** **sky** *hair* **water** *picture old* *brown yellow face grass* **window**
LIBSVM	*book smile gray sun flag* *computer brick man yellow street* *machine* **sea** *leaf road ocean* *couple forest fly purple toy*	**man hair** *black movie face* *food fire boy* **smile** *lady* *metal statue dance couple red* *table toy arm bike gold*	*building* **sky** *fence floor church* *shirt legs wall money glass* *ship room couple word city* *bald door guy orange chart*
LIBSVM+Platt	*book man smile white* **blue** *sky* **black** *woman red* **green** *people* **tree water** *computer girl* *face old hair yellow leaf*	*movie food* **man hair** *white* **smile woman** *blue face black* **people** *green red* **girl** *fire* *tree sky boy table eye*	*building* **sky** *man red floor* **white** *black woman* **blue** *church* *fence people green hair* **tree** *face shirt room grass chart*
MLR-L1	**tree green** *hair movie white* **black** *people grass statue leaf* *orange old bike red flower* *mountain* **picture** *dance eye dirt*	**hair** *tree black movie green* **man** *eye* **woman** *white hand* *face* **girl people smile** *dance* *red hat orange statue brown*	**tree** *hair movie black* **white** *green square people eye* **blue** *dance hand hat orange logo* *wall red statue man bike*

Figure 3. Examples of test images from the ESP Game dataset with annotations generated by different multi-label learning methods. The correct keywords are highlihted by bold font whereas the incorrect ones are highlighted by italic font.