PPP: Joint Pointwise and Pairwise Image Label Prediction

Yilin Wang[1] Suhang Wang[1] Jiliang Tang[2] Huan Liu[1] Baoxin Li[1]
[1]Department of Computer Science, Arizona State Univerity
[2]Yahoo Research
{yilinwang,suhang.wang,huan.liu,baoxin.li}@asu.edu jlt@yahoo-inc.com

Abstract

Pointwise label and pairwise label are both widely used in computer vision tasks. For example, supervised image classification and annotation approaches use pointwise label, while attribute-based image relative learning often adopts pairwise labels. These two types of labels are often considered independently and most existing efforts utilize them separately. However, pointwise labels in image classification and tag annotation are inherently related to the pairwise labels. For example, an image labeled with "coast" and annotated with "beach, sea, sand, sky" is more likely to have a higher ranking score in terms of the attribute "open"; while "men shoes" ranked highly on the attribute "formal" are likely to be annotated with "leather, lace up" than "buckle, fabric". The existence of potential relations between pointwise labels and pairwise labels motivates us to fuse them together for jointly addressing related vision tasks. In particular, we provide a principled way to capture the relations between class labels, tags and attributes; and propose a novel framework PPP(Pointwise and Pairwise image label Prediction), which is based on overlapped group structure extracted from the pointwise-pairwise-label bipartite graph. With experiments on benchmark datasets, we demonstrate that the proposed framework achieves superior performance on three vision tasks compared to the state-of-the-art methods.

1. Introduction

The increasing popularity of social media generates massive data at an unprecedented rate. The ever-growing number of images has brought new challenges for efficient and effective image analysis tasks, such as image classification, annotation and image ranking. Based on the types of labels, we can roughly divide the supervised vision tasks into two categories – pointwise label based approaches and pairwise label based approaches. Pointwise approaches adopt pointwise labels such as image categories or tags as training targets [10, 20, 6, 19, 8, 23, 24]. Class labels in classification often capture high-level image content, while tags in tag annotation are likely to describe a piece of information in the image, such as "high heel, buckle, leather" in a shoe image. In [21], these two tasks are considered together because the labels and tags may have some relations in an image. Recently, due to the semantic gap between low-level image features and high-level image concepts, human nameable visual attributes are proposed to solve the vision tasks[7, 14, 1, 13]. However, for a large variety of attributes, the pointwise binary setting is restrictive and unnatural. For example, it is very difficult to assign or not assign "sporty" to the middle car in Figure 1 because different people have different opinions. Thus, pairwise approaches [17, 11, 12] have been proposed, which aim to learn a ranking function to predict the attribute strength for images. For example, in Figure 1. most of the people would agree that the middle car is more "sporty" than the left one and less "sporty" than the right one

Figure 1. An Illustrative Example of poinwise labels and pairwise labels. Pointwise label "4 door" is better than the pairwise label to describe presence of 4 door in a car, while "sporty" is better to use pairwise label to describe the car style, as the right is more sporty than the left. For example it is hard to label the middle (we ask 10 human viewer – 40% agree with the non sporty and 60% agree with sporty, but 100% agree with middle one is more sporty than the left one and less sporty than right one).

Pointwise and pairwise labels have their own advantages as well as limitations in terms of labeling complexity and representational capability. **Labeling complexity**: given 10 images, we only need 10 sets of class categories/tags. However, we need to label at least 45 image pairs to capture the overall ordering information. (Although the ranking rela-

tion is considered as transferable, e.g. $A \succ B \& B \succ C \Rightarrow A \succ C$). **Representational capability**: pointwise labels such as tags/class labels imply the presence of content properties such as whether a shoe is made of leather, contains a heel, buckle, etc. While pairwise labels capture the relations in a same property, e.g., A has a higher heel than B. Solely relying on pointwise labels may cause ambiguity or produce noisy data for the models as in the example of assigning "sporty" to the middle car in Figure 1, while only using pairwise labels may also cause problems when the images have very similar properties.

As pointwise and pairwise labels encapsulate information of different types and may have different benefits for vision problems and recommendation systems[22] , we develop a new framework for fusing different types of training data by capturing their underlying relations. For example, in Figure 2, the tags, "leather, cognac, lace up" may suggest the left shoe with a higher score on the "formal" attribute, while the "high heel" may indicate the right shoe with a lower score on the "comfort" attribute. On the other hand, the higher score on "formal" and "comfort" with tag "Oxford" could help label the left image as "shoe" and enable the rare tag annotation such as "wingtip". To the best of our knowledge, there are only a few recent works that fused pointwise and piarwise labels [18, 4]. However, they simply combined regression and ranking in the loss functions for ranking tasks and totally ignored the relations between pointwise labels and pairwise labels.

In this paper, we investigate the problem of fusing pointwise and pairwise labels by exploiting their underlying relations for joint pointwise label prediction such as, image classification and annotation, and pairwise label prediction, e.g., relative ranking. We derive a unified bipartite graph model to capture the underlying relations among two types of labels. Since traditional approaches cannot take advantages of relations among pointwise and pairwise labels, we proceed to study two fundamental problems: (1) how to capture relations between pointwise and pairwise labels mathematically; and (2) how to make use of the relations for jointly addressing vision tasks. These two problems are tackled by the propose framework PPP and our contributions are summarized as follows:

- We provide a principled approach to modeling relations between pointwise and pairwise labels;
- we propose a novel joint framework PPP, which can predict both pointwise and pairwise labels for images simultaneously; and
- We conduct experiments on various benchmark datasets to understand the working of the proposed framework PPP.

In the remaining of the paper, we first give a formal problem definition and basic model in Section 2. Then the proposed framework and an optimization method for model learning is presented in Section 3. Experiments and results are demonstrated in Section 4, with further discussion in section 5.

2. The Proposed Method

Before detailing the proposed framework, we first introduce notations used in this paper. We use $\mathbf{X} \in \mathbb{R}^{n \times d}$ to denote a set of images in the database where n is the number of images and d is the number of features. Note that there are various ways to extract features such as SIFT, Gist or the features learned via deep learning frameworks. Let $\mathbf{Y}_t \in \mathbb{R}^{n \times c_1}$ and $\mathbf{Y}_c \in \mathbb{R}^{n \times c_3}$ be the data-tag and data-label matrices which represent the pointwise labels. $\mathbf{Y}(i,j) = 1$ if the i-th image is annotated/classified with j-th tag/class label, $\mathbf{Y}(i,j) = 0$ otherwise. Given a fixed training set D, a candidate pair set P can be drawn. The pair set implied by the fixed training set D uses pairwise labels. In the proposed framework, given a pair of images $< a, b >$ on the attribute q, if $y_a \succ y_b$, then a has a positive attribute score $y(a,q,1) = |y_a - y_b|$, and a negative score $y(a,q,2) = 0$; while b has a positive attribute $y(b,q,1) = 0$, and a negative score $y(b,q,2) = |y_a - y_b|$. Thus, the pairwise label is defined as $\mathbf{Y}_r \in \mathbb{R}^{m \times c_2}$, where m is the number of pairs drawn from training samples and $c_2 = 2q$ where q is the number of attributes. For example, let $< a, b >$ be the first pair, the pairwise label $\mathbf{Y}_r(1, 2(q-1)+1)$ represents how likely the $y_a \succ y_b$ and $\mathbf{Y}_r(1, 2(q-1)+2)$ represents how likely $y_a \prec y_b$ on attribute q.

2.1. Baseline Models

In our framework, pointwise labels are considered for classification and annotation tasks. For classification, we assume that there is a linear classifier $\mathbf{W}_c \in \mathbb{R}^{d \times c_3}$ to map $\mathbf{X}$ to the pointwise label $\mathbf{Y_c}$ as $\mathbf{Y}_c = \mathbf{X}\mathbf{W}_c$. $\mathbf{W}_c$ can be obtained by solving the following optimization problem:

$$\min_{\mathbf{W}_c} \Omega(\mathbf{W}_c) + \mathcal{L}(\mathbf{W}_c, \mathbf{Y}_c, D) \tag{1}$$

where $\mathcal{L}()$ is a loss function and Ω is a regularization penalty to avoid overfitting, D is the training sample set. Here we employ least square for loss function $\mathcal{L}$.

For tag annotation, we also assume that there is a linear function $\mathbf{W}_t \in \mathbb{R}^{d \times c_1}$ which captures the relation between data $\mathbf{X}$ and pointwise label $\mathbf{Y}_t$ as $\mathbf{Y}_t = \mathbf{X}\mathbf{W}_t$. Similarly, the optimization problem to learn $\mathbf{W}_t$ is:

$$\min_{\mathbf{W}_t} \Omega(\mathbf{W}_t) + \mathcal{L}(\mathbf{W}_t, \mathbf{Y}_t, D) \tag{2}$$

For pairwise label based approaches, a simple and successful approach to utilizing the pairwise label is Rank

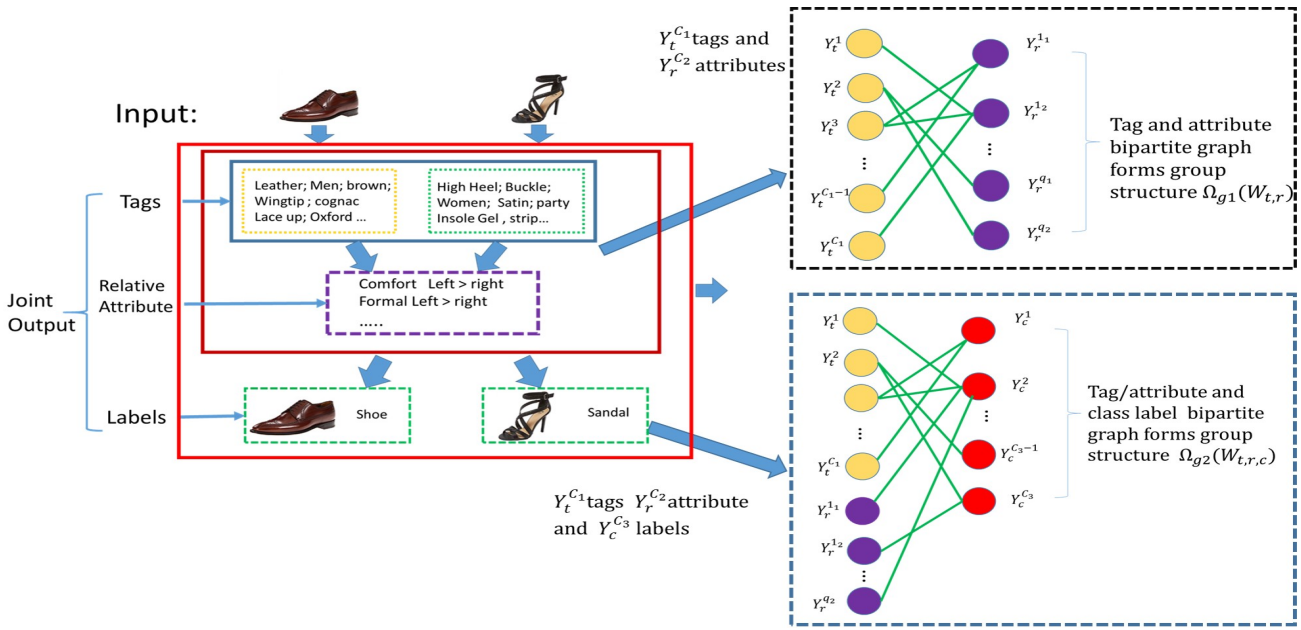

Figure 2. The demonstration of capturing the relations between pointwise label and pairwise label via bipartite graph. For example, the attribute "formal" with tags "leather, lace up, congnac" will form a group via the upper bipartite graph, while label "sandal" with attribute "less formal" and tags "high heel, party" will form a group via the lower bipartite graph.

SVM, whose goal is to learn a model $\mathbf{W}$ that achieves little loss over a set of previously unseen data, using a prediction function. Similar to RankSVM, in our framework, the original distribution of training examples are expanded into a set of candidate pairs and the learning process is over a set of pairwise feature vectors as:

$$\min_{\mathbf{W}} \mathcal{L}(\mathbf{W}, \mathbf{Y}_r, P) + \Omega(\mathbf{W}_r) \tag{3}$$

where P is a set of training pairs. The loss function $\mathcal{L}$ is defined over the pairwise difference vector x:

$$\mathcal{L}(\mathbf{W}, \mathbf{Y}_r, P) = \sum_{((a,y_a,q_a),(b,y_b,q_b))\in P} l(t(y_a - y_b), f(w, a-b)) \tag{4}$$

where the transformation function $t(y)$ transforms the difference of the labels [18]. In our framework, the transformation function is defined as $t(y) = sign(y)$.

Note that one may form a unified model by simply adding all the above objective functions together. Such an approach would still essentially treat the component models as independent tasks (albeit trade-off among them might be considered via weighting), since no explicit relations among them are considered.

2.2. Capturing Relations between Poinwise and Pairwise Labels

In the previous subsection, we defined three tasks that use pointwise and pairwise labels separately. Capturing the relations between pointwise and pairwise labels can further pave a way for us to develop a joint framework that enables interaction between classification, annotation and ranking simultaneously.

First, the relations between attributes and tags can be denoted as a bipartite graph as shown in Figure 2. We assume that $\mathbf{B} \in \mathbb{R}^{c_2 \times c_1}$ is the adjacency matrix of the graph where $\mathbf{B}(i,j) = 1$ if both the i-th tag and the j-th attribute co-occur in the same image and $\mathbf{B}(i,j) = 0$ otherwise. Note that in this paper, we do not consider the concurrence frequencies of tags and attributes and we would like to leave it as one future work. From the bipartite graph, we can identify groups of attributes and tags where attributes and tags in the same group could share similar properties such as semantical meanings. A feature $\mathbf{X}(:,i)$ should be either relevant or irrelevant to the attributes and tags in the same group. For example, $\mathbf{W}_r(i,j)$ indicates the effect of the i-th feature on predicting the j-th attribute; while $\mathbf{W}_t(i,k)$ denotes the impact of the i-th feature on the k-th tag. Therefore we can impose constraints on $\mathbf{W}_t$ and $\mathbf{W}_r$ together, which are derived from group information on the bipartite graph, to capture relations between attributes and tags.

We can adopt any community detection algorithms to identify groups from the bipartite graph. In this paper, we

use a very simple way to extract groups from the bipartite graph – for the j-th attribute, we consider the tags that connect to that attribute in the bipartite graph as a group, i.e., $\mathbf{B}(i,j) = 1$. Note that a tag may connect to several attributes thus extracted groups via the aforementioned process have ***overlaps***. Assume that $\mathcal{G}$ is the set of groups we detect from the attribute-tag bipartite graph and we propose to minimize the following term to capture relations between attributes and tags as:

$$\Omega_{\mathcal{G}}(\mathbf{W}_{t,r}) = \sum_{i=1}^{d} \sum_{g \in \mathcal{G}} \alpha_g \left\| \mathbf{w}_g^i \right\|_2 \tag{5}$$

where $\mathbf{W}_{t,r} = [\mathbf{W}_t, \mathbf{W}_r]$ and α_g is the confidence of the group g and $\mathbf{w}_g^i$ is a vector concatenating $\{\mathbf{W}_{t,r}(i,j)\}_{j \in g}$. For example, if $g = \{1,5,9\}$, $\mathbf{w}_g^i = [\mathbf{W}_{t,r}(i,1), \mathbf{W}_{t,r}(i,5), \mathbf{W}_{t,r}(i,9)]$. Next we discuss the inner workings of Eq. (5). Let us check terms in Eq. (5) related to a specific group g, $\sum_{i=1}^{d} \left\| \mathbf{w}_g^i \right\|_2$, which is equal to adding a ℓ_1 norm on the vector $\mathbf{g} = [\mathbf{w}_g^1, \mathbf{w}_g^2, \ldots, \mathbf{w}_g^d]$, i.e., $\|\mathbf{g}\|_1$. That ensures a sparse solution of $\mathbf{g}$; in other words, some elements of $\mathbf{g}$ could be zero. If $\mathbf{g}_i = 0$ or $\|\mathbf{w}_g^2\|_2 = 0$, the effects of the i-th feature on both the attribute and tags in the group g are eliminated simultaneously.

Similarly, we build the bipartite graph to capture the underlying relations for the attributes and class labels. In [21], it was suggested that the co-occurrence of tags and labels should also be considered. Thus, we build a mixture bipartite graph to extract the group information between class labels, tags, and attributes. The group regularization $\Omega_{\mathcal{G}2}(\mathbf{W}_{t,r,c})$ is similar to Eq. 5 and illustration is shown in Figure 2, where a tag or an attribute will connect to the class label if they are associated with each other. Note that a group extracted from Figure 2 could include a class label, a set of attributes and a set of tags.

2.3. The Proposed Framework

With the model component to exploit the bipartite graph structures, the proposed framework is to solve the following optimization problem:

$$\begin{aligned} \min_{\mathbf{W}} \; & \mathcal{L}(\mathbf{W}_c, \mathbf{Y}_c, D) + \mathcal{L}(\mathbf{W}_t, \mathbf{Y}_t, D) + \mathcal{L}(\mathbf{W}_r, \mathbf{Y}_r, P) \\ & + \lambda(\|\mathbf{W}_c\|_F^2 + \|\mathbf{W}_t\|_F^2 + \|\mathbf{W}_r\|_F^2) \\ & + \alpha\Omega_{\mathcal{G}1}(\mathbf{W}_{t,r}) + \beta\Omega_{\mathcal{G}2}(\mathbf{W}_{t,r,c}) \end{aligned} \tag{6}$$

In Eq. 6, the first six term is from the basic models to predict the class label, tags and ranking order. The seventh and eighth term are to capture the overlapped structure of the output, which is controlled by α and β respectively. The group regularization is defined as blow:

$$\Omega_{\mathcal{G}}(\mathbf{Z}) = \sum_{i \in \mathcal{G}} \|\mathbf{Z}_g\|_2 = \sum_{i=1}^{d} \sum_{i \in \mathcal{G}} \|\mathbf{z}_g^i\|_2 \tag{7}$$

3. An Optimization Method for PPP

Since the group structures are overlapped, directly optimizing the objective function is difficult. We propose to use Alternating Direction Method of Multiplier (ADMM)([25, 2]) to optimize the objective function. We first introduce two auxiliary variables $\mathbf{P} = [\mathbf{W}_t, \mathbf{W}_r]\mathbf{M}_1$ and $\mathbf{Q} = [\mathbf{W}_t, \mathbf{W}_r, \mathbf{W}_c]\mathbf{M}_2$. $\mathbf{M}_1 \in \{0,1\}^{(c_1+c_2)\times c_2(c_1+c_2)}$ is defined as: if $i-th$ tag connects to the jth attribute then $\mathbf{M}_1(i,(c_1+c_2)(j-1)+i) = 1$, otherwise it is zero. The definition of $\mathbf{M}_2 \in \{0,1\}^{(c_1+c_2+c_3)\times c_3(c_1+c_2+c_3)}$ is similar to $\mathbf{M}_1$. With these two variable, solving the overlapped group lasso on $\mathbf{W}$ is transfered to the non-overlapped group lasso on $\mathbf{P}$ and $\mathbf{Q}$, respectively. Therefore, the objective function becomes:

$$\begin{aligned} \min_{\mathbf{W},\mathbf{P},\mathbf{Q}} \; & \mathcal{L}(\mathbf{W}_c, D) + \mathcal{L}(\mathbf{W}_t, D) + \mathcal{L}(\mathbf{W}_r, P) \\ & + \alpha\Omega_{\mathcal{G}}(\mathbf{P}) + \beta\Omega_{\mathcal{G}2}(\mathbf{Q}) \\ & + \lambda(\|\mathbf{W}_c\|_F^2 + \|\mathbf{W}_t\|_F^2 + \|\mathbf{W}_r\|_F^2) \\ & s.t. \mathbf{P} = [\mathbf{W_t}, \mathbf{W_r}]\mathbf{M}_1; \mathbf{Q} = [\mathbf{W}_t, \mathbf{W}_r, \mathbf{W}_c]\mathbf{M}_2; \end{aligned} \tag{8}$$

which can be solved by the following ADMM problem:

$$\begin{aligned} \min_{\mathbf{W},\mathbf{P},\mathbf{Q}} \; & \mathcal{L}(\mathbf{W}_c, \mathbf{Y}_c, D) + \mathcal{L}(\mathbf{W}_t, \mathbf{Y}_t, D) + \mathcal{L}(\mathbf{W}_r, \mathbf{Y}_r, P) \\ & + \lambda(\|\mathbf{W}_c\|_F^2 + \|\mathbf{W}_t\|_F^2 + \|\mathbf{W}_r\|_F^2) + \alpha\Omega_{\mathcal{G}}(\mathbf{P}) \\ & + \beta\Omega_{\mathcal{G}2}(\mathbf{Q}) + \langle \mathbf{\Lambda}_1, \mathbf{P} - [\mathbf{W}_t, \mathbf{W}_r]\mathbf{M}_1 \rangle \\ & + \langle \mathbf{\Lambda}_2, \mathbf{Q} - [\mathbf{W}_t, \mathbf{W}_r, \mathbf{W}_c]\mathbf{M}_2 \rangle \\ & + \frac{\mu}{2}\|\mathbf{P} - [\mathbf{W}_t, \mathbf{W}_r]\mathbf{M}_1\|_F^2 \\ & + \frac{\mu}{2}\|\mathbf{Q} - [\mathbf{W}_t, \mathbf{W}_r, \mathbf{W}_c]\mathbf{M}_2\|_F^2 \end{aligned} \tag{9}$$

where $\mathbf{\Lambda}$ is the Lagrangian multiplier and μ is a scaler to control the penalty for the violation of equality constrains $\mathbf{P} = [\mathbf{W}_t, \mathbf{W}_r]\mathbf{M}_1$ and $\mathbf{Q} = [\mathbf{W}_t, \mathbf{W}_r, \mathbf{W}_c]\mathbf{M}_2$. Noting that the loss function L has lots of choices, we use the least square loss function in this paper.

3.1. Updating W

To update $\mathbf{W}$, we fix the other variable except $\mathbf{W}$ and remove terms that are irrelevant to $\mathbf{W}$. Then the Eq. 9 becomes:

$$\begin{aligned} \min_{\mathbf{W}} \; & \sum_{x \in D} \|x\mathbf{W}_t - y_t\|_2^2 + \sum_{x \in D} \|x\mathbf{W}_c - y_c\|_2^2 \\ & + \sum_{x_i, x_j \in P} \|(x_i - x_j)\mathbf{W}_r - y_r\|_2^2 \\ & + \lambda(\|\mathbf{W}_c\|_F^2 + \|\mathbf{W}_t\|_F^2 + \|\mathbf{W}_r\|_F^2) \\ & + \frac{\mu}{2}\|(\mathbf{P} + \frac{1}{\mu}\Lambda_1) - [\mathbf{W}_t, \mathbf{W}_r]\mathbf{M}_1\|_F^2 \\ & + \frac{\mu}{2}\|(\mathbf{Q} + \frac{1}{\mu}\Lambda_2) - [\mathbf{W}_t, \mathbf{W}_r, \mathbf{W}_c]\mathbf{M}_2\|_F^2 \end{aligned} \tag{10}$$

Setting the derivative of Eq. 10 w.r.t $\mathbf{W}_t$ to 0, we get:

$$\begin{aligned}&\mathbf{X}_D^T\mathbf{X}_D\mathbf{W}_t + \lambda\mathbf{W}_t + \mathbf{W}_t(\mathbf{M}_1^t\mathbf{M}_1^{tT} + \mathbf{M}_2^t\mathbf{M}_2^{tT}) \\ &= \mathbf{X}^T\mathbf{Y} + \frac{\mu}{2}[(\mathbf{P} + \frac{1}{\mu}\Lambda_1)\mathbf{M}_1^t + (\mathbf{Q} + \frac{1}{\mu}\Lambda_2)\mathbf{M}_2^t]\end{aligned} \tag{11}$$

where $\mathbf{M}_1^t$ is the part of $\mathbf{M}_1$ corresponding to $\mathbf{W}_t$. Directly getting the close form solution from Eq. 11 is intractable. On the other hand $\mathbf{X}_D^T\mathbf{X}_D + \frac{1}{2}\lambda\mathbf{I}$ and $\mathbf{M}_1^t\mathbf{M}_1^{tT} + \mathbf{M}_2^t\mathbf{M}_2^{tT} + \frac{1}{2}\lambda\mathbf{I}$ are symmetric and positive definite. Thus, we employ eigen decomposition for each of them:

$$\begin{aligned}\mathbf{X}_D^T\mathbf{X}_D + \frac{1}{2}\lambda\mathbf{I} &= \mathbf{U}_1\Sigma_1\mathbf{U}_1^T \\ \mathbf{M}_1^t\mathbf{M}_1^{tT} + \mathbf{M}_2^t\mathbf{M}_2^{tT} + \frac{1}{2}\lambda\mathbf{I} &= \mathbf{U}_2\Sigma_2\mathbf{U}_2^T\end{aligned} \tag{12}$$

where$\mathbf{U}_1, \mathbf{U}_2$ are eigen vectors and Σ_1, Σ_2 are diagonal matrices with eigen value on the diagonal. Substituting Eq. 12 into Eq. 11:

$$\begin{aligned}\mathbf{U}_1\Sigma_1\mathbf{U}_1^T\mathbf{W}_t + \mathbf{W}_t\mathbf{U}_2\Sigma_2\mathbf{U}_2^T &= \mathbf{X}_D^T\mathbf{Y}_t + \frac{\mu}{2}(\mathbf{P} + \frac{1}{\mu}\Lambda_1)\mathbf{M}_1^t \\ &+ \frac{\mu}{2}(\mathbf{Q} + \frac{1}{\mu}\Lambda_2)\mathbf{M}_2^t\end{aligned} \tag{13}$$

Multiplying $\mathbf{U}_1^T$ and $\mathbf{U}_2$ from left to right on both sides, and letting $\widetilde{\mathbf{W}_t} = \mathbf{U}_1^T\mathbf{W}_t\mathbf{U}_2$ and $\mathbf{Z}_t = \mathbf{U}_1^T[\mathbf{X}_D^T\mathbf{Y}_t + \frac{\mu}{2}[(\mathbf{P} + \frac{1}{\mu}\Lambda_1)\mathbf{M}_1^t + (\mathbf{Q} + \frac{1}{\mu}\Lambda_2)\mathbf{M}_2^t]]\mathbf{U}_2$, we can obtain:

$$\Sigma_1\widetilde{\mathbf{W}_t} + \widetilde{\mathbf{W}_t}\Sigma_2 = \mathbf{Z}_t \tag{14}$$

Then, we can get $\widetilde{\mathbf{W}_t}$ and $\mathbf{W}_t$ as:

$$\widetilde{\mathbf{W}_t}(s,t) = \frac{\mathbf{Z}_t(s,t)}{\sigma_1^s + \sigma_2^t} \tag{15}$$

$$\mathbf{W}_t = \mathbf{U}_1\widetilde{\mathbf{W}_t}\mathbf{U}_2^T \tag{16}$$

Similarly, setting the derivative of Eq. 10 w.r.t $\mathbf{W}_c$ to zero and apply the eigen decomposition, we have the closed form solution of $\mathbf{W}_c$:

$$\widetilde{\mathbf{W}_c}(s,t) = \frac{\mathbf{Z}_c}{\sigma_1^s + \sigma_3^t} \tag{17}$$

$$\mathbf{W}_c = \mathbf{U}_1\widetilde{\mathbf{W}_c}\mathbf{U}_3^T \tag{18}$$

where $\mathbf{Z}_c = \mathbf{U}_3^T[\mathbf{X}_D^T\mathbf{Y}_c + \frac{\mu}{2}(\mathbf{Q} + \frac{1}{\mu}\Lambda_2)]\mathbf{M}_2^c$ and $\mathbf{U}_3, \sigma_3$ are the eigen vector and eigen value for the symmetric and positive definite matrix $\mathbf{M}_2^c\mathbf{M}_2^{cT} + \frac{1}{2}\lambda\mathbf{I}$.

Noting that for $\mathbf{W}_r$, which input is data pairs, we can use the same learning process by using the transform label function mentioned above. For example, we regard the pair difference as one data sample for $\mathbf{X}_P$ and use the positive and negative label for label transformation. Setting the Eq. 10 w.r.t $\mathbf{W}_r$ to zero, we can obtain:

$$\begin{aligned}&\mathbf{X}_P^T\mathbf{X}_P\mathbf{W}_r + \lambda\mathbf{W}_r + \mathbf{W}_r(\mathbf{M}_1^r\mathbf{M}_1^{rT} + \mathbf{M}_2^r\mathbf{M}_2^{rT}) \\ &= \mathbf{X}_P^T\mathbf{Y}_r + \frac{\mu}{2}[(\mathbf{P} + \frac{1}{\mu}\Lambda_1)\mathbf{M}_1^r + (\mathbf{Q} + \frac{1}{\mu}\Lambda_2)\mathbf{M}_2^r]\end{aligned} \tag{19}$$

Similar to $\mathbf{W}_c$, with eigen decomposition, we can get the closed form solution for $\mathbf{W}_r$ as:

$$\widetilde{\mathbf{W}_r}(s,t) = \frac{\mathbf{Z}_r}{\sigma_4^s + \sigma_5^t} \tag{20}$$

$$\mathbf{W}_r = \mathbf{U}_4\widetilde{\mathbf{W}_r}\mathbf{U}_5^T \tag{21}$$

where $\mathbf{Z}_r = \mathbf{U}_4[\mathbf{X}_P^T\mathbf{Y}_r + \frac{\mu}{2}[(\mathbf{P} + \frac{1}{\mu}\Lambda_1)\mathbf{M}_1^r + (\mathbf{Q} + \frac{1}{\mu}\Lambda_2)\mathbf{M}_2^r]\mathbf{U}_5^T$, $\mathbf{U}_4, \sigma_4$ are eigen vector and eigen values for $\mathbf{X}_P^T\mathbf{X}_P + \frac{1}{2}\lambda\mathbf{I}$, and $\mathbf{U}_5, \sigma_5$ are eigen vector and eigen value for $\mathbf{M}_1^r\mathbf{M}_1^{rT} + \mathbf{M}_2^r\mathbf{M}_2^{rT} + \frac{1}{2}\lambda\mathbf{I}$.

3.2. Updating P

After removing terms that are irrelevant to $\mathbf{P}$, Eq. 9 becomes:

$$\min_{\mathbf{P}} \frac{\mu}{2}\|\mathbf{P} - [\mathbf{W}_t, \mathbf{W}_r]\mathbf{M}_1\|_F^2 + \alpha\Omega_{\mathcal{G}}(\mathbf{P}) + Tr(\Lambda_1\mathbf{P}) \tag{22}$$

When applied to the collection of group for the parameters, $\mathbf{P}$, $\Omega_{\mathcal{G}}(\mathbf{P}))$ no longer have overlapping groups. We denote $j-th$ group in i-th row as $\mathbf{P}_{i,j} = \mathbf{P}(i, (c_1+c_2)(j-1)+1 : (c_1+c_2)j)$. Hence, we can solve the problem separately for each row of $\mathbf{P}$ within one group by the following optimization:

$$\min_{\mathbf{P}_{i,j}} \alpha\|\mathbf{P}_{i,j}\|_2^2 + \frac{\mu}{2}\|\mathbf{P}_{i,j} - (([\mathbf{W}_c, \mathbf{W}_r]\mathbf{M}_1)_{i,j} - \frac{\Lambda_{1ij}}{\mu})\|_F^2 \tag{23}$$

Note that Eq. 23 is the proximal operator [27] of $\frac{1}{\mu}(P)_{i,j}$ applied to $(([\mathbf{W}_c, \mathbf{W}_r]\mathbf{M}_1)_{i,j} - \frac{\Lambda_{1ij}}{\mu})$. Let $\mathbf{Z}_{i,j}^P = ([\mathbf{W}_c, \mathbf{W}_r]\mathbf{M}_1)_{i,j} - \frac{\Lambda_{1ij}}{\mu}$. The solution by applying the proximal operator used in non-overlapping group lasso to each sub-vector is:

$$\mathbf{P}_{i,j} = prox(\mathbf{Z}_{i,j}^P) = \begin{cases} 0 & if\|\mathbf{Z}_{i,j}^P\|_2 \leq \frac{\alpha}{\mu} \\ \frac{\|\mathbf{Z}_{i,j}^P\|_2 - \frac{\alpha}{\mu}}{\|\mathbf{Z}_{i,j}^P\|_2}\mathbf{Z}_{i,j}^P & otherwise \end{cases} \tag{24}$$

3.3. Updating Q

Similar to $\mathbf{P}$, we can update $\mathbf{Q}$ by proximal operator used in non-overlapping group lasso to each sub-vector of $\mathbf{Q}$:

$$\mathbf{Q}_{i,j} = prox(\mathbf{Z}_{i,j}^Q) = \begin{cases} 0 & if\|\mathbf{Z}_{i,j}^Q\|_2 \leq \frac{\beta}{\mu} \\ \frac{\|\mathbf{Z}_{i,j}^Q\|_2 - \frac{\beta}{\mu}}{\|\mathbf{Z}_{i,j}^Q\|_2}\mathbf{Z}_{i,j}^Q & otherwise \end{cases} \tag{25}$$

where $\mathbf{Q}_{ij} = \mathbf{Q}(i, (c_1+c_2+c_3)(j-1)+1 : (c_1+c_2+c_3)j)$ and $\mathbf{Z}_{i,j}^Q = ([\mathbf{W_t}, \mathbf{W_r}, \mathbf{W_c}]M_2)_{i,j} - \frac{\Lambda_{i,j}}{\mu}$

3.4. Updating Λ_1, Λ_2 and μ

After updating the variables, we now need to update the ADMM parameters. According to [2], they are updated as follows:

$$\Lambda_1 = \Lambda_1 + \mu(\mathbf{P} - [\mathbf{W}_t, \mathbf{W}_r]\mathbf{M}_1) \quad (26)$$

$$\Lambda_2 = \Lambda_2 + \mu(\mathbf{Q} - [\mathbf{W}_t, \mathbf{W}_r, \mathbf{W}_c]\mathbf{M}_2) \quad (27)$$

$$\mu = min(\rho\mu, \mu_{max}) \quad (28)$$

Here, $\rho > 0$ is a parameter to control the convergence speed and μ_{max} is a large number to prevent μ from becoming too large.

With these updating rules, the optimization method for our proposed method is summarized in Algorithm 1

Algorithm 1 The algorithm for the proposed framwork

Input: $\mathbf{X}_D \in \mathbf{R}^{N\times d}$ and $\mathbf{X}_P \in \mathbf{R}^{m\times d}$ and corresponding label $\mathbf{Y}_t, \mathbf{Y}_c$ and $\mathbf{Y}_r$
Output: c_1 tags label c_2 relative score and c_3 class label for each data instance

1: Initialize random Sample training set D and drawn random pair set P from D.
2: Setting $\mu = 10^{-3}, \rho = 1.1, \mu_{max} = 10^8$ and building $\mathbf{M}_1$ and $\mathbf{M}_2$
3: Precompute the eigen decomposition
4: **repeat**
5: Calculate $\widetilde{\mathbf{W}_t}$, $\widetilde{\mathbf{W}_t}$ and $\widetilde{\mathbf{W}_r}$
6: Update $\mathbf{W}_t, \mathbf{W}_r$ and $\mathbf{W}_c$ by Eq. 16, Eq. 21, and Eq. 18, respectively.
7: Calculate $\mathbf{Z}^P$ and $\mathbf{Z}^Q$
8: Update $\mathbf{P}$ and $\mathbf{Q}$
9: Update Λ_1, Λ_2 and μ
10: **until** convergence
11: Using max pooling for testing use $\mathbf{XW}$ to predict tags, relative relation and labels.

3.5. Convergence Analysis

Since the sub-problems are convex for $\mathbf{P}$ and $\mathbf{Q}$, respectively, Algorithm 1 is guaranteed to converge because they satisfy the two assumptions required by ADMM. The proof of the convergence can be found in [2]. Specially, Algorithm 1 has dual variable convergence. Our empirical results show that our algorithm often converges within 100 iterations for all the datasets we used for evaluation.

3.6. Time Complexity Analysis

The main computation cost for $\mathbf{W}$ involves the eigen decomposition on $\mathbf{X^T X} + \frac{1}{2}\beta\mathbf{I}$, while other terms that involve eigen decomposition is very fast because the feature dimension of $\mathbf{MM}^T$ is small. The time complexity for eigen decomposition is $O(d^3)$. However, in Algorithm 1 the eigen decomposition is only computed once before the loop and dimension reduction algorithm can be employed to reduce image feature dimensions d. The computation cost for $\mathbf{Z}$ is $O(nd^2)$ due to the sparsity of $\mathbf{M}$. The computation of $\mathbf{P}$ depends on the proximal method within each group. Since there are c_2 groups which have the group size $c_1 + c_2$ for each feature dimension, the total computation cost for $\mathbf{P}$ is $O(dc_2(c_1 + c_2))$ and it is similar for $\mathbf{Q}$. It is worth noting that $\mathbf{P}$ and $\mathbf{Q}$ can be computed in parallel for each feature dimension.

4. Experiment

In this section, we conduct experiments to evaluate the effectiveness of PPP. After introducing datasets and experimental settings, we compare PPP with the state-of-the-art methods of tag prediction, classification and ranking.

4.1. Experiments Settings

The experiments are conducted on 3 publicly available benchmark datasets.

Shoe-Zappo dataset [26]: It is a large shoe dataset consisting of 50,025 catalog images collected from Zappos.com. The images are divided into 4 major categories shoes, sandals, slippers, and boots. The tags are functional types and individual brands such as high-heel, oxford, leather, lace up, and pointed toe. The number of tags is 147 and 4 relative attribute is defined as "open" , "pointy", "sporty" and "comfortable". The ground truth is labeled from AmazonTurk.

OSR-scene dataset [16]: It is a dataset for out door scene recognition with 2688 images. The images are divided into 8 category named as coast, forest, highway, inside-city, mountain, open-country, street and tall-building. 6 attributes with pointwise label and pairwise label are provided by [17] named by natural, open, perspective, large-objects, diagonal-plane and close-depth.

Pubfig-face dataset [13]: It is a dataset containing 800 images from 8 random identities (100 images per person) named Alex Rodriguez, Clive Owen, Hugh Laurie , Jared Leto , Miley Cyrus, Scarlett Johansson , Viggo Mortensen and Zac Efron. We use the 11 attributes with pintwise label and pairwise label provided by [17]. The example attributes are named as masculine-looking, white, young, smiling and etc.

4.2. Performance Comparison

We compare PPP with the following representative algorithms:

- SVM [3]: It uses the state of the art classifier SVM for classification with linear kernel; We also apply it to tag prediction by considering tags as a kind of labels;

- GLasso [28]: The original framework of group lasso is to handle high-dimensional and multi-class data. To extend it for joint classification and tag prediction, we also consider tags as a kind of labels and apply GLasso to learn the mapping of features to tags and label. Note that it does not make use of the pointwise and pairwise label bipartite graph. We use the implementation in [15];

- sLDA [21]: It is a joint framework based on topic models, which learns both class labels and annotations given latent topics;

- LS [9]: A multi-label classification method that exploits the label correlation information. To apply LS for joint classification and tag prediction, we consider tags as a kind of labels and use tag and label relations to replace the label correlation in the original model; and

- FT [6]: It is one of the state-of-the art annotation method which is based on linear mapping and co-regularized joint optimization. To apply it for classification, we consider labels as tags to annotate; and

- RD: It predicts labels and tags by randomly guessing.

- MultiRank [5]: It is a ranking method based on the assumption that the correlation exists between attributes, where the ranking function learns all attributes together via multi task learning framework.

- RA [17]: It is the method for image ranking based on relative attributes.

Note that for all the baseline methods, none of them can utilize both pointwise and pairwise labels. Although we get the performance of the proposed framework by jointly predicting both pointwise and pairwise labels, we present our results for each task separately for a clear comparison. Moreover, we could use more advanced features, e.g., CNN feature, however, to compare with other methods fairly, we adopt the original feature provided by each datasets, which can easily show the performance gain from the proposed model.

4.3. Pointwise label Prediction

For pointwise label prediction, our method is compared with SVM, Glasso, sLDA, LS, FT, and RD. For all the baseline methods with parameters, we use cross validation to determine their values. For the Shoe dataset, we use the same data split and features (990 gist and color features) in [26]. It contains 11102 data samples for training and 2400 data sample for testing. For OSR and Pubfig, we use the same data split and features in [17].

Table 1. Performance comparison in terms of classification. The number after each dataset means the class label number.

Method	Zappos(4)	OSR(8)	Pubfig (8)
SVM	67.41 %	42.21 %	50.77%
GLasso	78.31%	50.11%	59.13%
sLDA	74.32%	46.33%	56.21%
LS	84.46%	61.22%	66.56%
FT	84.69%	59.38%	67.45%
RD	25.01%	12.51%	12.50%
PPP	**89.39%**	**62.33%**	**74.95%**

Since OSR and Pubfig contain a small number of attributes, we leave one random-picked attribute for pairwise prediction and use the rest for tag annotation. Especially, to evaluate the performance of tag annotation, we rank all the tags based on their relevant scores and return the top K ranked tags. We use the average precision $AP@K$ as the evaluation metric which has been widely used in the literature [6, 21]. Meanwhile since the data samples are balanced, we use accuracy as the metric to evaluate the classification performance. The comparison results are shown in Table 1 and Table 2 for classification and tag annotation, respectively. We repeat 10 times for the training-testing process and report the average performance.

From the tables, we make the following observations:

- The proposed method that utilizes pairwise labels to predict pointwise labels tends to outperform the methods which solely rely on pointwise labels. These results support that (1) pairwise attributes can provide evidence for the pointwise label prediction; especially for the Pubfig dataset that contains 8 label classes, our method utilizes information from pairwise attributes significantly improve the classification performance. (2) The performance of tag prediction $AP@K$ indicates that the pairwise attributes contain important information for tag prediction;

- Our method with model components to capture relations between pairwise and pointwise labels outperforms those without. For example, compared to GLasso, the proposed framework, modeling the relations via the bipartite graph, gains remarkable performance improvement for both classification and tag prediction; and

- Most of the time, the proposed framework PPP performs the best among all the baselines, which demonstrates the effectiveness of the proposed algorithm. There are two major reasons. First, PPP jointly performs pointwise and pairwise label prediction. Second, PPP captures relations between labels by extracting group information from the bipartite graph, which

Table 2. Performance comparison in terms of tag recommendation.

Method	Zappo (147 tags)			OSR (5 tags)			Pubfig (10 tags)		
	AP@3	AP@5	AP@10	AP@1	AP@2	AP@3	AP@1	AP@3	AP@5
SVM	50.57%	40.89%	38.53%	68.51%	63.69%	60.12%	46.21%	38.31%	34.12%
GLasso	64.34%	59.81%	55.37%	87.11%	83.34%	80.87%	90.12%	88.76%	86.41%
sLDA	62.57%	57.22%	51.63%	90.15%	88.06%	84.78%	91.12%	87.93%	84.17%
LS	74.76%	66.62%	61.85%	94.19%	93.19%	**92.75%**	93.71%	92.66%	91.91%
FT	67.37%	59.52%	51.98%	**98.62%**	**94.45%**	92.22%	92.45%	91.50%	90.16%
RD	1.44%	1.43%	1.44%	20.00%	20.01%	20.01%	10.01%	10.00%	10.01%
PPP	**77.10%**	**71.08%**	**62.95%**	96.69%	94.21%	90.14%	**94.48%**	**93.67%**	**92.71%**

Table 3. The average ranking accuracy on three dataset

Method	Zappos	OSR	Pubfig
RA	70.37%	76.10%	71.23%
MultiRank	76.12%	84.93%	74.91%
PPP	**79.67%**	**88.40%**	**76.32%**

works as the bridge for building interactions between pointwise and pairwise labels.

4.4. Pairwise label Prediction

For pairwise label prediction, we generate pairs drawn from the training set used in the pointwise label prediction. For the Shoe dataset, we use 300 pairs; while for OSR and Pubfig, we use 100 pairs (the number suggested in [5]) drawn from training set. We compute the average ranking accuracy with standard deviation by running 10 rounds of each implementation. The results are shown in Table 3. Moreover, we also plot in Figure 3 to show how average accuracy changes with different sizes of training samples on the attributes on the Shoe dataset (due to the space limits, we omit the figure on OSR and Pubfig).

From Table 3 and Figure 3, we can have the following observations:

- The proposed method that leverages pointwise labels to predict pairwise labels often outperforms the methods which only use pairwise labels. These results support that pointwise labels can help the pairwise label prediction;
- The performance of the ranking accuracy varies with the number of the training pairs. With a small amount of labeled data, e.g., 10 pairs, the proposed method significantly outperforms relative attribute methods, which demonstrates that the pointwise labels contain important information for attribute ranking;
- The comparison based on multi-task attribute learning methods and our method demonstrates that simply combining the attributes together fails to differentiate these attributes which are not related to other attributes, while our methods use group structures,

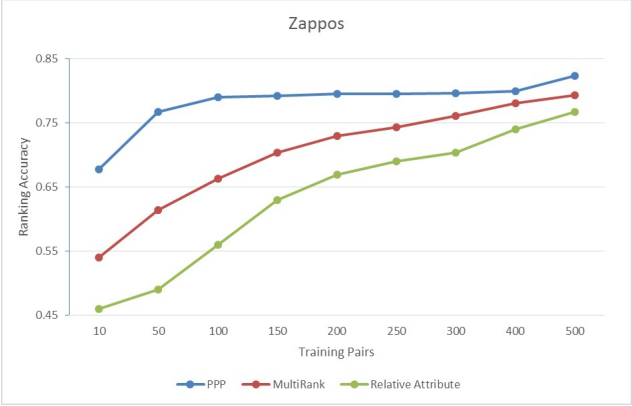

Figure 3. Learning curve of average ranking accuracy with regarding to different numbers of training pairs.

which makes the correlated attributes have strong overlaps, providing a discriminative way to capture the correlation between attributes.

5. Conclusion

In this paper, we propose a novel way to capture the relations between pointwise labels and pairwise labels. Moreover, PPP provides a new viewpoint for us to have a better understanding how pointwise and pairwise labels interact with each other. Experiments demonstrated : (1) the advantages of the proposed methods for pointwise label based tasks including image classification, tag annotation and pairwise label based image ranking; and(2) the importance of considering the group correlation between pointwise labels and pairwise labels.

6. Acknowledgement

The work was supported in part by ONR grants $N00014-15-1-2722$ and $N00014-15-1-2344$. Any opinions expressed in this material are those of the authors and do not necessarily reflect the views of ONR.

References

[1] T. L. Berg, A. C. Berg, and J. Shih. Automatic attribute discovery and characterization from noisy web data. In *Computer Vision–ECCV 2010*, pages 663–676. Springer, 2010.

[2] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

[3] C.-C. Chang and C.-J. Lin. Libsvm: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

[4] L. Chen, P. Zhang, and B. Li. Fusing pointwise and pairwise labels for supporting user-adaptive image retrieval. In *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*, pages 67–74. ACM, 2015.

[5] L. Chen, Q. Zhang, and B. Li. Predicting multiple attributes via relative multi-task learning. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1027–1034. IEEE, 2014.

[6] M. Chen, A. Zheng, and K. Weinberger. Fast image tagging. In *Proceedings of the 30th international conference on Machine Learning*, pages 1274–1282, 2013.

[7] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing objects by their attributes. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1778–1785. IEEE, 2009.

[8] D. Jannach, M. Zanker, A. Felfernig, and G. Friedrich. *Recommender systems: an introduction*. Cambridge University Press, 2010.

[9] S. Ji, L. Tang, S. Yu, and J. Ye. Extracting shared subspace for multi-label classification. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 381–389. ACM, 2008.

[10] Y.-G. Jiang, G. Ye, S.-F. Chang, D. Ellis, and A. C. Loui. Consumer video understanding: A benchmark database and an evaluation of human and machine performance. In *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, page 29. ACM, 2011.

[11] A. Kovashka and K. Grauman. Attribute adaptation for personalized image search. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 3432–3439. IEEE, 2013.

[12] A. Kovashka and K. Grauman. Attribute pivots for guiding relevance feedback in image search. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 297–304. IEEE, 2013.

[13] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Attribute and simile classifiers for face verification. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 365–372. IEEE, 2009.

[14] C. H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 951–958. IEEE, 2009.

[15] J. Liu, S. Ji, J. Ye, et al. Slep: Sparse learning with efficient projections.

[16] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International journal of computer vision*, 42(3):145–175, 2001.

[17] D. Parikh and K. Grauman. Relative attributes. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 503–510. IEEE, 2011.

[18] D. Sculley. Combined regression and ranking. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 979–988. ACM, 2010.

[19] B. Sigurbjörnsson and R. Van Zwol. Flickr tag recommendation based on collective knowledge. In *Proceedings of the 17th international conference on World Wide Web*, pages 327–336. ACM, 2008.

[20] G. Toderici, H. Aradhye, M. Paşca, L. Sbaiz, and J. Yagnik. Finding meaning on youtube: Tag recommendation and category discovery. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3447–3454. IEEE, 2010.

[21] C. Wang, D. Blei, and F.-F. Li. Simultaneous image classification and annotation. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1903–1910. IEEE, 2009.

[22] S. Wang, J. Tang, Y. Wang, and H. Liu. Exploring implicit hierarchical structures for recommender systems. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, pages 1813–1819, 2015.

[23] Y. Wang, Y. Hu, S. Kambhampati, and B. Li. Inferring sentiment from web images with joint inference on visual and social cues: A regulated matrix factorization approach. In *Proceedings of the Ninth International Conference on Web and Social Media, ICWSM 2015, University of Oxford, Oxford, UK, May 26-29, 2015*, pages 473–482, 2015.

[24] Y. Wang, S. Wang, J. Tang, H. Liu, and B. Li. Unsupervised sentiment analysis for social media images. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, pages 2378–2379, 2015.

[25] D. Yogatama and N. Smith. Making the most of bag of words: Sentence regularization with alternating direction method of multipliers. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 656–664, 2014.

[26] A. Yu and K. Grauman. Fine-grained visual comparisons with local learning. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 192–199. IEEE, 2014.

[27] L. Yuan, J. Liu, and J. Ye. Efficient methods for overlapping group lasso. In *Advances in Neural Information Processing Systems*, pages 352–360, 2011.

[28] M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006.