Neurocomputing 167 (2015) 614–624

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Image tag-ranking via pairwise supervision based semi-supervised model

Yonghao He *, Cuicui Kang, Jian Wang, Shiming Xiang, Chunhong Pan

Institute of Automation, Chinese Academy of Sciences, No. 95, Zhongguancun East Road, Haidian District, Beijing, China

ARTICLE INFO

Article history:
Received 11 November 2014
Received in revised form
12 April 2015
Accepted 12 April 2015
Communicated by X. Gao
Available online 25 April 2015

Keywords:
Image tag-ranking
Pairwise supervision
Learning to rank

ABSTRACT

Image tag-ranking, the task to sort tags based on their relevance to the related images, has become a hot topic in the field of multimedia. Most existing methods do not incorporate the tag-ranking order information into the models, which is actually very important to solve the issue of image tag-ranking. In this paper, by taking advantage of such important information, we propose a novel model which uses images with ranked tag lists as its supervision information. In the proposed method, each ranked tag list is decomposed into a number of image–tag pairs, all of which are pooled together for training a scoring function. With this pairwise supervision, the model is able to capture the intrinsic ranking structures. In addition, unsupervised data, namely images with unranked tag lists, is also integrated for digging the binary order: relevant or irrelevant. By leveraging both the pairwise supervision and unsupervised structural information, our model sufficiently exploits the tag relevance to images as well as the ranking structures of tag lists. Extensive experiments are conducted on both image tag-ranking and tag-based image search with three benchmark datasets: SUNAttribute, Labelme and MSRC, demonstrating the effectiveness of the proposed model.

1. Introduction

In recent years, Internet users are willing to share their personal information (such as blogs, videos and pictures) and enjoy the information from others at the same time, which brings the prosperity of social networks. In order to take advantage of the uploaded information from users, the providers of social networks encourage users to attach meaningful tags while uploading the associated information. For example, when users share their pictures, system may ask users to type some keywords (tags) or to choose some recommended keywords that best describe the contents of their pictures. By doing this, system can facilitate applications such as image search and interests group recommendation. Actually, these applications can be further improved by utilizing the carefully sorted keywords. For instance, a user who loves cats may upload a picture containing a cute cat walking in the wild, and he may attach the keywords in a random order: “grass”, “tree”, “path”, “cat” and “sky”. One interests group recommendation system that considers the order of keywords is very likely to mis-categorize the user to the groups such as “Nature Photography” by simply analyzing that the first three keywords (“grass”, “tree”, “path”) are closely related to the “nature”, which deviates from the real intention of the user. However, if the uploaded keywords can be properly sorted before being fed into the recommendation system, say “cat”, “path”, “grass”, “tree” and “sky”, it is more likely to categorize the user to the “Cat Fans” group.

The goal of image tag-ranking is to sort tags according to their relevance to the contents of the images. The issue of tag-ranking has been investigated in [1], where the statistics of position distribution of the most important tags is presented: for 1200 images with at least 10 tags randomly selected from Flickr,[1] there are only less than 10% of the images having their most relevant tags at the first place. Furthermore, we also make an analogous analysis on the benchmark datasets, the SUNAttribute [2], Labelme [3] and MSRC [4], which are used in the experiments. Since each tag is assigned to a relevance level from 0 to 3, corresponding to irrelevant to the most relevant, we calculate the average relevance levels of the top tags. In Fig. 1, it is obviously observed that the original average relevance levels of the top tags are lower than the optimal ones, which indicates that there are many highly relevant tags that are not placed at the top positions.

By now, there have been some methods proposed for image tag-ranking. Most of them, such as [1,5–7], are unsupervised methods.[2] In

* Corresponding author.
E-mail address: yhhe@nlpr.ia.ac.cn (Y. He).

[1] http://www.flickr.com

[2] In the field of image tag-ranking, “unsupervised” means that the tags of the images are in random orders, not properly ranked. And “supervised” indicates that the tags are ranked according to their relevance to the associated images. We emphasize that the “tags” themselves, existing for both “unsupervised” and “supervised” scenarios, are not related to unsupervised or supervised information in the case that the tags in many classification problems are supervised information.

http://dx.doi.org/10.1016/j.neucom.2015.04.027
0925-2312/© 2015 Elsevier B.V. All rights reserved.

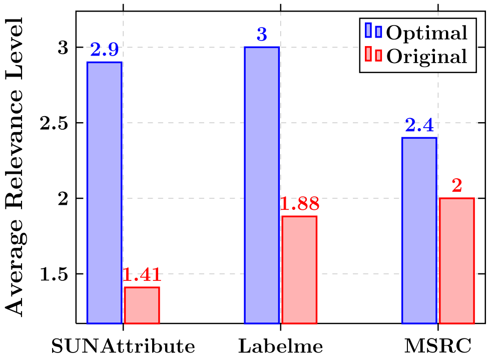


Fig. 1. Average relevance levels at the first position on SUNAttribute, Labelme and MSRC.

practise, unsupervised data is much more than supervised data, which makes researchers put lots of endeavor to utilize unsupervised information. Since supervised data (the images with ranked tag lists) is so limited that there are very few algorithms proposed based on it. For example, [8,9] present a semi-supervised method and a purely supervised method, respectively.

In this paper, a pairwise supervision based semi-supervised model is proposed to address the issue of image tag-ranking (we name it **PSTR**). In the literature, pairwise supervision is well studied in learning to rank (L2R) techniques [10], which motivates the proposed model. It addresses the ranking problem by viewing a ranked list as a number of item pairs to preserve the relative ranking structure, which is mostly ignored by the existing tag-ranking methods. By utilizing the pairwise supervision, the proposed model will gain the global view of ranking structure by decomposing the ranked tag lists into image–tag pairs. Moreover, we also integrate the unsupervised information into our model. The tags in unranked tag lists are deemed as weak ranking information, since we can tell whether the tags are relevant or irrelevant. So the proposed model is semi-supervised by using both the pairwise supervision and unsupervised information. The main contributions of our work are listed below:

- The idea of pairwise supervision is introduced into the proposed model for image tag-ranking. The pairwise supervision treats the tag lists as image–tag pairs—the items in the lists are no longer viewed independently and have mutual interactions, which makes the model capable of predicting the tag relevance by considering all pairs in a list.
- We leverage the unsupervised data which is viewed as the weak ranking information to facilitate the proposed model. The final objective function in the semi-supervised model consists of two components: pairwise supervision item and unsupervised item, and it can be elegantly optimized with a closed-form solution.
- Two experiments (image tag-ranking and tag-based image search) have been carried out to compare the proposed model with state-of-the-art algorithms on three benchmark datasets. The experimental results show that the proposed method can produce better ranked tag lists.

The remainder of this paper is structured as follows. Section 2 briefly summarizes the related work. In Section 3, the motivation deriving from L2R algorithms is first introduced. Then the proposed model is described in detail. Extensive experiments are shown in Section 4. Finally, the conclusions are drawn in Section 5.

2. Related work

The issue of image tag-ranking has attracted remarkable attentions, and various methods have been proposed for it. *Automatic image annotation* [11–15], which automatically assigns meaningful and content-related tags to the corresponding images, provides preliminary insights into this issue. These methods only generate coarse tag lists for the untagged images. They do not consider the orders of the tags. Whereas, image tag-ranking sorts tags in existing tag lists. Following image annotation, *tag refinement* methods, such as [16–18], are required for more precise image–tag association. These methods are built upon image annotation, namely taking the results from automatic image annotation as the initializations, and subsequently explore which tags are more appropriate to be annotated. One problem is that the initial tags may not be correctly provided by image annotation methods, whereas in image tag-ranking tags are all supposed to appropriately describe the images.

To address the issue of image tag-ranking, a number of specifically designed methods have been proposed. As mentioned in Section 1, they are divided into three categories, i.e., unsupervised, semi-supervised and supervised methods.

Unsupervised methods: For tag-ranking, most methods, such as [1,5–7], are designed in the unsupervised fashion. Ref. [1] is the first attempt to address image tag-ranking problem. In [1], Liu et al. assign initial relevance scores to the tags by using kernel density estimation (KDE), and then perform a random walk based refinement on a tag–tag similarity graph. Li et al. [5] propose to learn the tag relevance by neighbor voting. The idea of [5] includes two steps: (1) calculate the image nearest neighbors; (2) accumulate the tag votes within the top K neighbors. The method in [6] assumes that in the view of an image, the image can be represented as a weighted combination of the relevant tags. And for a tag, the tag can be expressed as a weighted combination of the representative images. Then a image–tag correlation matrix is learned under the criterion that two images with high similarity are close in the tag view, and vice versa. The final tag relevance scores are the elements in the image–tag correlation matrix. Differently, Sun et al. [7] use commercial search engines as auxiliaries to collect images for each tag, and then a Bayesian based model is proposed to estimate the initial relevance scores for the tags by using the collected images. Finally, a random walk is performed on the tag graph to refine the tag scores. To address the issue of personalized tag recommendation, Zhao et al. [19] propose a graph based ranking method, leveraging the benefit of traditional manifold ranking. This method can achieve good performance to recommend tags for users. Thus, we can see that these unsupervised methods rely on the tag relevance propagation via visual similarity, but ignoring the ranking structure information within the tag lists.

Semi-supervised methods: To our knowledge, [8] is the only method in the semi-supervised fashion. The aim of this method is to obtain a projection matrix that projects visual features to tag relevance space. The supervised component is formulated as the linear regression between the tag relevance scores and the projected image visual features. The unsupervised component is a regularizer that restricts the large relevance scores only appearing for the tags that are annotated to the associated images. Finally, it results in a quadratic programming problem. In our view, a single linear projection cannot capture complicated relationships between visual feature space and tag relevance space, and the tag-biased regularization is also not related to the inner ranking structures among the tag lists.

Supervised methods: Lan et al. [9] propose a Max-Margin Rifled Independence Model for tag-ranking. The main idea is that the max-margin formalism with riffled independence factorization proposed in [20] can perform structure learning. Therefore, this model can predict the tag orders in the tag permutation space.

Besides the above methods, *learning to rank* (L2R) techniques [10] have the potential to accomplish the task of image tag-ranking. However, there is no existing methods utilizing L2R to tackle the image tag-ranking issue. Some difficulties that prevent researchers from directly applying L2R to tag-ranking will be discussed in the next section, and we introduce the pairwise supervision that originates from L2R techniques into the proposed model for image tag-ranking.

3. Image tag ranking using pairwise supervision

In this section, the motivation, which derives from the pairwise based L2R algorithms, is first described. Then, the pairwise supervision based semi-supervised model is proposed to address image tag-ranking problem.

3.1. *Motivation from learning to rank algorithms.*

The proposed model is originally motivated by the pairwise based learning to rank algorithms. To begin with, we introduce the L2R algorithms and their difficulties to be applied to image tag-ranking problem.

Without loss of generality, we take document retrieval as an example to introduce the L2R algorithms. Assume that the training data contains two sets: the query set $\mathcal{Q}$ and the document set $\mathcal{D}$. Each query $q_i \in \mathcal{Q}$ is associated with a document set $\mathcal{D}_i = \{d_{i,1}, d_{i,2}, \ldots, d_{i,j}, \ldots, d_{i,n_i}\}$ and a relevance level set $\mathcal{Y}_i = \{y_{i,1}, y_{i,2}, \ldots, y_{i,j}, \ldots, y_{i,n_i}\}$, where n_i is the number of documents for query q_i. The related relevance level set $\mathcal{Y}_i$ depicts how relevant document $d_{i,j}$ is related to query q_i, and each element $y_{i,j}$ in $\mathcal{Y}_i$ is assigned to a nonnegative integer (the larger, the more relevant). By now, the training set can be represented as triads $\mathcal{T} = \{(q_i, \mathcal{D}_i, \mathcal{Y}_i)\}_{i=1}^{N}$, where N is the number of queries.

The goal of L2R is to learn a relevant level prediction function $f(q_i, d_{i,j})$ by minimizing the empirical risk function

$$\hat{R}(f) = \frac{1}{N} \sum_{i=1}^{N} L\big(F(q_i, \mathcal{D}_i), \mathbf{y}_i\big), \tag{1}$$

where $F(q_i, \mathcal{D}_i) = [f(q_i, d_{i,1}), f(q_i, d_{i,2}), \ldots, f(q_i, d_{i,n_i})]^T$, $\mathbf{y}_i = [y_{i,1}, y_{i,2}, \ldots, y_{i,n_i}]^T$ is the vector form of $\mathcal{Y}_i$, $L(\cdot)$ is the loss function to measure the errors between the predicted relevance levels and the ground-truth. Since L2R is feature based, namely a feature vector $\mathbf{x}_{i,j} = \phi(q_i, d_{i,j})$ is generated based on each query–document pair $(q_i, d_{i,j})$, the relevant level prediction function $f(q_i, d_{i,j}) = f(\mathbf{x}_{i,j})$ actually deals with the feature vectors rather than the raw query–document pairs. There exist several methods, such as BM25 [21] and PageRank [22], to generate the feature vectors for the query–document pairs. The relevant level prediction function, in most L2R algorithms, is defined as the linear model

$$f(\mathbf{x}_{i,j}) = \mathbf{w}^T \mathbf{x}_{i,j}, \tag{2}$$

where $\mathbf{w}$ is the parameter to be learned.

Generally, based on the definitions of the loss function $L(\cdot)$, L2R algorithms are divided into three categories: pointwise based, pairwise based and listwise based methods. In pointwise based L2R approaches, each query–document pair is processed independently, and the ranking problem is simply viewed as classification or regression problems. In pairwise based L2R approaches, the training data is reconstructed into tuples: under the same query, two query–document pairs form a tuple. When the first pair is more relevant than the second, the formed tuple is positive, otherwise it is negative. Ranking SVM [23] uses these tuples to transform the ranking problem to pairwise classification. Listwise based L2R approaches view the document ranking lists of each query as instances in the learning procedure, rather than a single document–query pair or a two-pair tuple. We focus on pairwise supervision, in which the document ranking lists are decomposed into a number of query–document pair tuples. In this way, the inner ranking structure information is reflected by the relative relationships in these tuples.

Although well studied and applied to many practical applications, it is difficult to directly use L2R for image tag-ranking. The images and their associated tags are from two modalities, i.e., visual information and textual words, whereas the queries and the documents belong to the same modality, i.e., text. Due to the different modalities, it is hard to jointly represent the images and the tags to form feature vectors for L2R algorithms. Joint subspace learning [24] can be adopted here to learn a subspace for the images and the tags, if the preliminary that the samples in two modalities should be characterized as feature vectors is met. However, the tags, usually in the form of a single word, are less informative. It is unlikely to be extracted meaningful feature vectors from the tags, which prevents us from using joint subspace learning. Since the joint feature representations between the images and the tags are difficult to obtain, the feature based linear prediction model (2) is not suitable in this case. In view of the above issues, L2R algorithms cannot be directly used to solve the image tag-ranking problem. However, we can absorb the thought of pairwise supervision in L2R into the proposed model.

3.2. *Semi-supervised image tag ranking*

We propose a model for image tag-ranking by using the pairwise supervision which is significant in L2R algorithms. On one hand, the relevance prediction function in our model is similarity based, rather than feature based. Thus our model adopts the superiority of the pairwise supervision while avoiding to construct the joint feature representations of images and tags compared with L2R algorithms. On the other hand, our model incorporates the images with the unranked tag lists (unsupervised information). Hence, it is semi-supervised.

3.2.1. *Notations*

Some notations are redefined in the scenario of image tag-ranking. Suppose that the image set is $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^{N}$. N is the number of images and $\mathbf{x}_i$ is the feature vector of the ith image. The tag set is $\mathcal{T} = \{t_1, t_2, t_3, \ldots, t_M\}$ and M is the size of the vocabulary. For each image $\mathbf{x}_i$, its associated tag indicator vector is $\mathbf{y}_i \in \{0,1\}^M$. $y_{i,j} = 0$ means that the jth tag is absent for the ith image, while $y_{i,j} = 1$ means the presence of the jth tag. Furthermore, assume that the first l images have ranked tag lists and the remaining $N-l$ images only have tag lists but without ranking. Let $\mathbf{r}_i \in \mathcal{R}^M$ $(i = 1, 2, 3, \ldots, l)$ denote the tag relevance vector of the ith image. $r_{i,j} \in \mathcal{R}$ in $\mathbf{r}_i$ depicts how the jth tag is related to the ith image (the larger, the more relevant). By now, the first l images have both the tag indicator vectors and the tag relevance vectors, but the last $N-l$ images only have the tag indicator vectors. The goal is to rank the tags of the last $N-l$ images.

3.2.2. *Pairwise supervision in image tag ranking*

In this subsection, we introduce pairwise supervision in our model. In the pairwise supervision based L2R algorithms, the ranked lists are viewed as pair tuples that are fed into the learning procedure. Incorporating pairwise supervision into the L2R algorithms is characterized by designing specific loss functions. Thus, in order to utilize pairwise supervision in the proposed model, we have to define a pairwise based loss function.

First, for the supervised part, the tag relevance vectors $\mathbf{r}_n$ are transformed to the relative relation matrices $\mathbf{C}_n \in \mathcal{R}^{M \times M}$ by using

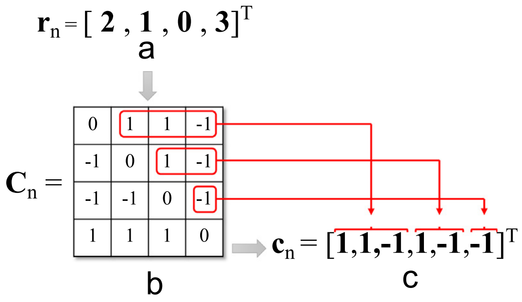

$\mathbf{r}_n = [\,2\,,1\,,0\,,3]^T$
a
$\mathbf{C}_n =$
0 1 1 -1
-1 0 1 -1
-1 -1 0 -1
1 1 1 0
b
$\mathbf{c}_n = [1,1,-1,1,-1,-1]^T$
c

Fig. 2. Transformation from the tag relevance vector $\mathbf{r}_n$ to the relative relation matrix $\mathbf{C}_n$ and the final vector $\mathbf{c}_n$.

the following criterion:

$$\mathbf{C}_n(i,j) = \begin{cases} 1, & \mathbf{r}_{n,i} > \mathbf{r}_{n,j} \\ -1, & \mathbf{r}_{n,i} < \mathbf{r}_{n,j} \\ 0, & \mathbf{r}_{n,i} = \mathbf{r}_{n,j} \end{cases}, \quad i,j = 1,2,3,\ldots,M, \tag{3}$$

in which $\mathbf{C}_n(i,j)$ is the element of $\mathbf{C}_n$ in the ith row and jth column. This process is shown in Fig. 2(a) and (b). Evidently, the tag relevance vectors and the relative relation matrices can represent the identical ranking order equivalently. Each element in $\mathbf{C}_n$ describes the relative ranking positions of two tags with respect to image $\mathbf{x}_n$ (namely image–tag pairs), hence the entire relative relation matrix $\mathbf{C}_n$ has the capability to capture the inner tag-ranking structure. Subsequently, we define the predicted relative relation matrices as follows:

$$\hat{\mathbf{C}}_n(i,j) = \text{sgn}(f(\mathbf{x}_n, t_i) - f(\mathbf{x}_n, t_j)), \tag{4}$$

where $f(\mathbf{x}_n, t_i)$ is the relevance prediction function for scoring the image–tag pair $(\mathbf{x}_n, t_i)$, and $\text{sgn}(\cdot)$ is the sign function. Actually, we do not care about the output values of the relevance prediction function, but concern about the relative relations between these values. For example, if t_i is more relevant than t_j, the only requirement is $f(\mathbf{x}_n, t_i) > f(\mathbf{x}_n, t_j)$, no matter what the real values of $f(\mathbf{x}_n, t_i)$ and $f(\mathbf{x}_n, t_j)$ will be. According to the above transformations, the loss function is defined as

$$\mathcal{L}(f) = -\frac{1}{l} \sum_{n=1}^{l} \sum_{i=1}^{M} \sum_{j=1}^{M} \mathbf{C}_n(i,j) \hat{\mathbf{C}}_n(i,j). \tag{5}$$

When the two matrices $\mathbf{C}_n$ and $\hat{\mathbf{C}}_n$ are identical, the loss value reaches the minimum. Since the relative relation matrices leverage the pairwise supervision, the proposed loss function (5) is pairwise based.

To avoid using the feature based relevance prediction function, we propose a similarity based relevance prediction function. As mentioned before, forming joint feature representations of the images and the tags is very difficult, and we only have the features of the images. As a result, we define the relevance prediction function as

$$f(\mathbf{x}_n, t_i) = (\mathbf{M}\mathbf{s}_n)_i, \tag{6}$$

where $\mathbf{M} \in \mathcal{R}^{M \times N}$ is the image–tag correlation matrix to be learned, $\mathbf{s}_n \in \mathcal{R}^N$ collects the similarities between the nth image and all images, and $(\cdot)_i$ represents the ith element of a vector. The image–tag correlation matrix bridges the images and the tags: each row indicates the relevance of all images to a tag, and each column depicts the relevance of all tags to an image. To measure the similarity of two images, we use the following metric:

$$s_{n,m} = \exp\left(-\frac{d(\mathbf{x}_n, \mathbf{x}_m)}{\sigma \cdot \log_2(2 + |\mathcal{T}_n \cap \mathcal{T}_m|)} \right), \tag{7}$$

where $s_{n,m}$ indicates the similarity between the nth image and the mth image, $d(\cdot,\cdot)$ is a distance measurement of two images, $\mathcal{T}_n$ and $\mathcal{T}_m$ are the tag sets, $\cap$ is the intersection of two sets, $|\cdot|$ represents the cardinality of a set and $\sigma \in (0,1]$ is a balance parameter. The component $2 + |\mathcal{T}_n \cap \mathcal{T}_m|$ in the denominator is to ensure that $\log_2(\cdot)$ results in a positive number no less than 1. Metric (7) is calculated by using both the low-level visual features and the high-level semantic features, thus the "semantic gap" problem can be alleviated. From another point of view, the relevance prediction function (6) is a process of relevance linear propagation based on image similarities throughout the whole dataset.

Substituting (6) into (4), we obtain

$$\hat{\mathbf{C}}_n(i,j) = \text{sgn}\Big((\mathbf{M}\mathbf{s}_n)_i - (\mathbf{M}\mathbf{s}_n)_j \Big). \tag{8}$$

The sign function is non-differentiable, thus it makes the final objective function difficult to be optimized. To address this problem, we use the signed magnitude as an approximate surrogate of the sign function

$$\hat{\mathbf{C}}_n(i,j) \approx (\mathbf{M}\mathbf{s}_n)_i - (\mathbf{M}\mathbf{s}_n)_j. \tag{9}$$

If the signed magnitude (9) is directly substituted into the loss function (5), the loss function will become hard to be optimized due to the operator $(\cdot)_i$ in (6). Fortunately, we observe that the relative relation matrices are skew-symmetric, which means that only half of the elements in $\mathbf{C}_n$ (the upper triangular part or the lower triangular part) are sufficient for computing the loss function. Therefore, we can convert the matrix $\mathbf{C}_n$ to a vector $\mathbf{c}_n$ by aligning the elements of the upper triangular part of the matrix $\mathbf{C}_n$ by row as follows:

$$\mathbf{c}_n = [\mathbf{C}_n(1,2), \mathbf{C}_n(1,3), \ldots, \mathbf{C}_n(1,M), \\ \mathbf{C}_n(2,3), \mathbf{C}_n(2,4), \ldots, \mathbf{C}_n(M-1,M)]^T. \tag{10}$$

The above conversion is illustrated in Fig. 2(b) and (c). Analogously, the predicted relative relation matrices $\hat{\mathbf{C}}_n$ are also skew-symmetric, and they can be converted to the vector forms by multiplying an alternating subtraction matrix

$$\mathbf{A} = \begin{bmatrix} 1 & -1 & 0 & 0 & \cdots \\ 1 & 0 & -1 & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \cdots \\ 1 & 0 & 0 & \cdots & -1 \\ 0 & 1 & -1 & 0 & \cdots \\ 0 & 1 & 0 & -1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \cdots & \cdots & \cdots & 1 & -1 \end{bmatrix}, \tag{11}$$

$$\hat{\mathbf{c}}_n = \mathbf{A}\mathbf{M}\mathbf{s}_n, \tag{12}$$

in which $\mathbf{A} \in \mathcal{R}^{a \times M}$, and $a = \binom{M}{2}$. By conducting the conversions (10) and (12), the sum of the elements in the matrix calculated from the element-wise multiplication between $\mathbf{C}_n$ and $\hat{\mathbf{C}}_n$ can be easily transformed to an inner product of two vectors $\mathbf{c}_n$ and $\hat{\mathbf{c}}_n$. Thus, the loss function in (5) is rewritten as

$$\mathcal{L}(\mathbf{M}) = -\frac{1}{l} \sum_{n=1}^{l} \sum_{i=1}^{M} \sum_{j=1}^{M} \mathbf{C}_n(i,j) \hat{\mathbf{C}}_n(i,j) = -\frac{2}{l} \sum_{n=1}^{l} \mathbf{c}_n^T \hat{\mathbf{c}}_n \\ = -\frac{2}{l} \sum_{n=1}^{l} \mathbf{c}_n^T \mathbf{A}\mathbf{M}\mathbf{s}_n = -\frac{2}{l} \text{tr}(\mathbf{C}^T \mathbf{A}\mathbf{M}\mathbf{S}_l), \tag{13}$$

where $\mathbf{C} = [\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_n, \ldots, \mathbf{c}_l] \in \mathcal{R}^{a \times l}$, $\mathbf{S}_l = [\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_n, \ldots, \mathbf{s}_l] \in \mathcal{R}^{N \times l}$ and $\text{tr}(\cdot)$ is the trace of a matrix. Since we only use the upper triangular parts of matrices $\mathbf{C}_n$ and $\hat{\mathbf{C}}_n$ to generate vectors $\mathbf{c}_n$ and $\hat{\mathbf{c}}_n$, the loss is twice (the constant number 2 in (13)) to the sum of all the inner products.

So far, the loss function only contains the item constructed with supervised information (namely pairwise supervision), and it is a linear function with respect to $\mathbf{M}$, making it impossible to reach the minimum of loss function (13). In the next subsection, the unsupervised information is incorporated into the loss function to introduce a quadratic item of $\mathbf{M}$, making the loss function solvable.

3.2.3. Unsupervised information in image tag ranking

Unsupervised data is very important for addressing the image tag-ranking problem, and the reasons are two-folds: (1) in practice, the supervised information is in the minority, which may lead to a low vocabulary coverage. That is to say, some tags that appear in the unsupervised set may be absent in the supervised set. As a result, some image–tag correlation cannot be sufficiently learned. (2) unsupervised information can not only result in a high vocabulary coverage, but also provide weak ranking information. Although the majority of images do not have ranked tag lists, they are annotated with tags. We call the annotation weak ranking information because it can separate tags into two groups: relevant and irrelevant. The tags that are relevant to the image should be ranked ahead of those irrelevant ones. To incorporate this weak ranking information into the model, we propose the following binary-relevance regression item:

$$\frac{1}{N-l}\sum_{i=l+1}^{N}\|\mathbf{y}_i-\mathbf{M}\mathbf{s}_i\|_2^2, \tag{14}$$

where $\|\cdot\|_2$ is ℓ_2-norm. This item is to guide the image–tag correlation matrix $\mathbf{M}$ to capture the coarse ranking information introduced by the tag indicator vector $\mathbf{y}_i$.

Actually, to further validate the effectiveness of the proposed semi-supervised model, two trivial variations based on the original unsupervised term (14) are constructed. The first variation (named as PSTR-S) uses only supervised data in the binary-relevance regression item. This configuration will make the model solvable with only supervised data. As mentioned above, this may lead to a low vocabulary coverage and then insufficient learning of the image–tag correlation. Another variation (named as PSTR-US) leverages both unsupervised and supervised data to construct the term (14). In our view, adding supervised data in the unsupervised term may be redundant, since the weak ranking information from supervised data is included by the strong ranking information that is modeled by the pairwise based linear term (13).

Algorithm 1. Workflow of the proposed semi-supervised model.

```
Input:
    The supervised data ${(\mathbf{x}_i,\mathbf{y}_i,\mathbf{r}_i)}_{i=1}^{l}$ and the unsupervised
    data ${(\mathbf{x}_j,\mathbf{y}_j)}_{j=1}^{N-l}$.
Output:
    The ranked tag lists for each image in the unsupervised set.
1: Transform tag relevance vectors $\mathbf{r}_n$ into relative relation
   matrices $\mathbf{C}_n$ using (3).
2: Convert relative relation matrices $\mathbf{C}_n$ into the corresponding
   vectors $\mathbf{c}_n$ by using (10).
3: Generate the alternating substraction matrix $\mathbf{A}$ that has the
   form of (11).
4: Calculate the mutual image similarities using (7) to obtain $\mathbf{S}_l$
   and $\mathbf{S}_u$.
5: Compute the image–tag correlation matrix $\mathbf{M}$ by
   substituting $\mathbf{A}$, $\mathbf{C}$, $\mathbf{S}_l$, $\mathbf{S_u}$ and $\mathbf{Y}_u$ into the closed-form solution
   in (17).
6: Use the relevance prediction function (6) to score each tag,
   and the final tag lists are ranked in the descending order
   based on the tag relevance scores.
```

3.2.4. Final model for image tag ranking

So far, we leverage both the pairwise supervision and the unsupervised information to design the respective loss functions. Combining (13) and (14), the final semi-supervised model for image tag-ranking is formulated as follows:

$$\min_{\mathbf{M}}\mathcal{L}(\mathbf{M})=-\frac{1}{l}tr(\mathbf{C}^T\mathbf{AMS}_l)+\frac{\beta}{N-l}\|\mathbf{Y}_u-\mathbf{MS}_u\|_F^2+\lambda\|\mathbf{M}\|_F^2, \tag{15}$$

where $\mathbf{Y}_u=[\mathbf{y}_{l+1},\mathbf{y}_{l+2},\dots,\mathbf{y}_N]\in\mathcal{R}^{M\times(N-l)}$, $\mathbf{S}_u=[\mathbf{s}_{l+1},\mathbf{s}_{l+2},\dots,\mathbf{s}_N]\in\mathcal{R}^{N\times(N-l)}$, and β and λ are penalty parameters. The constant number 2 is omitted in the first item, and the item $\lambda\|\mathbf{M}\|_F^2$ is to ensure an available solution and avoid over-fitting.

To solve the optimization problem (15), we take the derivative of loss function $\mathcal{L}$ with respect to $\mathbf{M}$

$$\frac{\partial\mathcal{L}}{\partial\mathbf{M}}=-\frac{1}{l}\mathbf{A}^T\mathbf{CS}_l^T+\frac{2\beta}{N-l}(\mathbf{MS}_u\mathbf{S}_u^T-\mathbf{Y}_u\mathbf{S}_u^T)+2\lambda\mathbf{M}, \tag{16}$$

and set it to 0

$$\mathbf{M}=\left(2l\beta\mathbf{Y}_u\mathbf{S}_u^T+(N-l)\mathbf{A}^T\mathbf{CS}_l^T\right)\cdot\left(2l\beta\mathbf{S}_u\mathbf{S}_u^T+2\lambda l(N-l)\mathbf{I}\right)^{-1}. \tag{17}$$

The optimization results in a closed-form solution. Once the image–tag correlation matrix $\mathbf{M}$ is obtained, the relevance scores are calculated using the relevance prediction function (6). Finally, the tags are ranked in the descending order based on their relevance scores. The proposed method is summarized in Algorithm 1.

As for the two variations, Eqs. (15) and (17) need to be rewritten as

- *For PSTR-S*:

  $$\min_{\mathbf{M}}\mathcal{L}(\mathbf{M})=-\frac{1}{l}\,\mathrm{tr}(\mathbf{C}^T\mathbf{AM}\hat{\mathbf{S}}_l)+\frac{\beta}{l}\|\mathbf{Y}_l-\mathbf{M}\hat{\mathbf{S}}_l\|_F^2+\lambda\|\mathbf{M}\|_F^2, \tag{18}$$

  $$\mathbf{M}=\left(\mathbf{A}^T\mathbf{C}\hat{\mathbf{S}}_l^T+2\beta\mathbf{Y}_l\hat{\mathbf{S}}_l^T\right)\left(2\beta\hat{\mathbf{S}}_l\hat{\mathbf{S}}_l^T+2\lambda l\mathbf{I}\right)^{-1}, \tag{19}$$

  where $\hat{\mathbf{S}}_l\in\mathcal{R}^{l\times l}$ (mutual similarities within all supervised images), $\mathbf{Y}_l=[\mathbf{y}_1,\mathbf{y}_2,\dots,\mathbf{y}_l]\in\mathcal{R}^{M\times l}$ and $\mathbf{M}$ has the dimensions of $M\times l$. In test phase, the similarities between the test image and all supervised images are calculated. The relevance scores of image–tag pairs are also obtained using (4).
- *For PSTR-US*:

  $$\min_{\mathbf{M}}\mathcal{L}(\mathbf{M})=-\frac{1}{l}\,\mathrm{tr}(\mathbf{C}^T\mathbf{AMS}_l)+\frac{\beta}{N}\|\mathbf{Y}-\mathbf{MS}\|_F^2+\lambda\|\mathbf{M}\|_F^2, \tag{20}$$

  $$\mathbf{M}=\left(2l\beta\mathbf{YS}^T+N\mathbf{A}^T\mathbf{CS}_l^T\right)\cdot\left(2l\beta\mathbf{SS}^T+2\lambda lN\mathbf{I}\right)^{-1}, \tag{21}$$

  where $\mathbf{Y}=[\mathbf{y}_1,\mathbf{y}_2,\dots,\mathbf{y}_N]\in\mathcal{R}^{M\times N}$, $\mathbf{S}=[\mathbf{s}_1,\mathbf{s}_2,\dots,\mathbf{s}_N]\in\mathcal{R}^{N\times N}$ and $\mathbf{M}\in\mathcal{R}^{M\times N}$.

3.2.5. Complexity analysis

In this subsection, we make a brief analysis about computational complexity and space cost. A big anxiety is related to matrix $\mathbf{A}$. When the vocabulary size M is large, the row dimension $(a=\binom{M}{2})$ of $\mathbf{A}$ will become very large. However, it is evident that $\mathbf{A}$ is highly sparse, namely only two non-zero elements (1 and -1) in each row and $M-1$ non-zero elements in each column, which makes the computation efficient. In the closed-form solution (17), $\mathbf{A}$ is engaged in $\mathbf{A}^T\mathbf{CS}_l^T$. Generally, dynamic programming can be used to handle the matrix multiplication with more than two dense matrices. In our case, there are only three matrices, thus it is simple to solve the problem by exhaustive search. A preliminary knowledge is that the computational complexity of multiplying two matrices of dimensions $a\times b$ and $b\times c$ results in $O(abc)$. If $\mathbf{A}$ is viewed as a dense matrix, the complexity of computing $\mathbf{A}^T\mathbf{CS}_l^T$ is $\min\left(O\left(M\times\binom{M}{2}\times l+M\times l\times N\right),O\left(\binom{M}{2}\times l\times N+M\times\binom{M}{2}\times N\right)\right)$. However, when $\mathbf{A}$ is sparse in our case, the computational complexity decreases to $\min(O(M\times(M-1)\times l+M\times l\times N)$, $O\left(\binom{M}{2}\times l\times N+M\times(M-1)\times N\right))$. In practice, matrix

Table 1
Summary of the characteristics of three datasets.

Name	# of images	# of tags	# of tags per image	Properties of tags	Features
SUNAttribute [25,2]	14,340	102	Min: 3 Max: 37 Average: 15.5	Gerunds: 37 Nouns: 38 Adjectives: 27	Gist, hog2 × 2, ssim, geometry specific histograms, bag of words
Labelme [3]	3825	190	Min: 1 Max: 26 Average: 7.0	Gerunds: 0 Nouns: 190 Adjectives: 0	Gist, color histogram, bag of words
MSRC [4]	461	23	Min: 2 Max: 7 Average: 2.9	Gerunds: 0 Nouns: 23 Adjectives: 0	Gist, color histogram, bag of words

multiplication can be highly paralleled, especially using GPU, which greatly boosts the computation efficiency. Another issue is to solve the large scale linear system in Eq. (17). In general, it can be solved in $O(N^3)$.

As for space cost, we can count the total elements contained in all matrices. For $\mathbf{Y}_u$, $\mathbf{A}$ and $\mathbf{I}$, they are highly sparse matrices including $\hat{M} \times (N-l)$, $2 \times \binom{M}{2}$ and N non-zero elements respectively, where $\hat{M}$ is the average number of tags for each image. For dense matrices $\mathbf{S}_l$ and $\mathbf{S}_u$, there are $N \times l$ and $N \times (N-l)$ elements, respectively. The total number of elements is $\left(\hat{M}N - \hat{M}l + M^2 - M + N + N^2\right)$. In practice, the number of images (N) is much larger than the number of tags (M), thus the space cost is dominated by N.

4. Experiments

In this section, we conduct experiments of two applications, image tag-ranking and tag-based image search on three benchmark datasets: SUNAttribute [25,2], Labelme [3] and MSRC [4]. Afterwards, the influence of hyper parameters is sufficiently analyzed.

4.1. The benchmark datasets

In the experiments, three benchmark datasets, i.e., SUNAttribute, Labelme and MSRC, are used to evaluate the effectiveness of the proposed algorithm. First, we give brief introductions to these datasets below (a shortcut view is in Table 1).

SUNAttirbue [25,2] database is the latest released and largest dataset among the three (actually, there are much larger datasets in which images have tags, such as NUS-WIDE [26], but these tag lists are not ranked, so it is unavailable to tag-ranking). It is for scene understanding and recognition and contains 14,340 images from 707 categories covering diverse scenes. The vocabulary size is 102, and a variety of attributes, such as materials, lighting, surface properties and spatial envelope properties, are described. There are 37 gerunds, 38 nouns of objects and 27 adjectives in the vocabulary. The diversity of the tags makes an important influence on the complexity of the dataset. The tag numbers of one image varies from 3 to 37 (15.5 on average). For each tag of an image, the votes from three AMT annotators are provided [2]. The more votes a tag receives, the more relevant the tag will be. Finally, each tag is scored into 4 levels: 0 (irrelevant), 1 (weakly relevant), 2 (relevant), and 3 (most relevant). The ranking groundtruth is based on these relevance levels. There are 4 types of features extracted for each image, namely gist [27], hog2 × 2 [28], ssim [29], bag of words [30] and geometry specific histograms [25]. As far as we know, SUNAttribute is the most complex and largest public-released dataset for image tag-ranking.

Labelme [3] database is collected by a powerful web-based tool[3] for image annotation. Since it is a big collection of images with related tags and is still growing, we only take a subset of Labelme which was released by Hwang et al. [31]. In this subset, there are 3825 images and 209 tags. However, we find that 19 tags are absent for all images, so the vocabulary size is reduced to 190. All the tags are nouns of objects. Each image has 7.0 tags on average, 1 at least and 26 at most. The tag relevance scores provided in [31] are continuous values, so we quantize these values into the same 4 levels as for SUNAttribute. The features for each image are gist, color histogram in HSV color space [32] and bag of words.

MSRC [4] is a small dataset for image segmentation. There are 591 images and 23 classes (tags of objects). We find that there are many images containing only one tag, which is not appropriate in the scenario of tag-ranking. So these images are removed. As a result, the final dataset contains 461 images. In this dataset, each image has 2.9 tags on average, 2 at least and 7 at most. The tag relevance scores are generated from the segmentation groundtruth: the scores are computed as the object area percentage of the images in the pixel level. These relevance scores are also quantized into 4 levels as for SUNAttribute. As for the features, gist, color histogram and bag of words are extracted for each image. Notice that, MSRC is a toy dataset for image tag-ranking.

Since our model is semi-supervised, three datasets are randomly split into supervised and unsupervised parts:

- On SUNAttribute, supervised: 4340 images; unsupervised: 10,000 images.
- On Labelme, supervised: 1500 images; unsupervised: 2325 images.
- On MSRC, supervised: 230 images; unsupervised: 231 images.

Our goal is to leverage both the supervised and unsupervised parts to rank the tags of images in the unsupervised parts.

4.2. Experimental settings

Due to the multiple types of features are used when calculating the image similarities, metric (7) is modified as

$$s_{n,i} = \exp\left(-\frac{\sum_{k=1}^{\mathcal{K}} d_k(\mathbf{x}_n^k, \mathbf{x}_i^k)}{\sigma \cdot \log_2(2+|\mathcal{T}_n \cap \mathcal{T}_i|)}\right), \tag{22}$$

where $\mathcal{K}$ is the number of feature types, $\mathbf{x}_n^k$ is the kth type of feature, and $d_k(\cdot,\cdot)$ is a feature-related distance measurement. Specifically, ℓ_2 distance is for gist, and χ^2 distance is for hog 2×2, ssim, bag of words and color histogram.

[3] http://labelme.csail.mit.edu/Release3.0/index.php

The following methods are taken as the baselines:

Unsupervised methods:

- *Probabilistic and Random Walk based Tag Ranking (PRWTR)* [1]: This method requires that the number of images for each tag should be 50. The dataset SUNAttribute is the only one that satisfies this requirement. Thus, the experiment of PRWTR is only carried out on SUNAttribute.
- *Learning Relevance by Neighbor Voting (LRNV)* [5]: The tag relevance is simply accumulated by neighbor voting in this method.
- *Two-view Learning for Tag Ranking (TLTR)* [6]: This method learns tag scores within a tag–image relevance matrix.

For these unsupervised methods, the entire datasets are used, not just on the split unsupervised parts, since all images are annotated with tags. The final performances are based on the split unsupervised parts.

Semi-supervised methods:

- *Learning to Rank Tags (L2RT)* [8]: As far as we know, this method is the only semi-supervised method for image tag-ranking. Therefore, it is meaningful to compare our method with L2RT.

Recall that, two variations (PSTR-S and PSTR-US) of the proposed model are also taken as comparative methods. Notice that, PSTR-S is a fully supervised method.

4.3. Experiments of image tag ranking

To verify the effectiveness of the proposed model, we first compare it with other approaches in terms of image tag-ranking. The measurement for evaluating the performance is Normalized Discounted Cumulative Gains (NDCG) [33] which is widely used to measure the performance of tag ranking. Here we use the top K NDCG scores

$$NDCG@K = \frac{1}{Z}\sum_{i=1}^{K}\frac{2^{rel(i)}-1}{\log(1+i)}, \qquad (23)$$

where K indicates that NDCG scores are calculated using the top K ranked tags (K is taken from 1 to 10 in image tag-ranking), Z is the normalization constant that guarantees the optimal NDCG score is 1. $rel(i)$ is the relevance score of the tag in the ith position. If the top K ranked tags are more relevant, the NDCG scores are higher.

The final performances of all methods are described in Table 2. We can make some observations from the results.

On SUNAttribute ((a) in Table 2): The proposed method and its variations are consistently better than the unsupervised methods PRWTR, LRNV, TLTR and the semi-supervised method L2RT, showing that the proposed model is adaptable to deal with different and complex scenes appearing in the images. PSTR and PSTR-US can perform equally well, but PSTR-S is marginally lower. As for the unsupervised methods, TLTR performs better than LRNV, and the two methods achieve satisfactory results benefitting from the training on the whole dataset as unsupervised information. Surprisingly, L2RT, as a semi-supervised method, shows a relatively low performance which is even worse than the unsupervised methods. The reason may lie on that a simple linear projection from visual feature space to tag relevance space is not enough to capture the complexity of the dataset. It can be seen that L2RT is able to predict good ranking results on the smaller and simpler datasets in the following.

On Labelme ((b) in Table 2): L2RT obtains comparable results with our methods, achieving marginally higher NDCG scores when $K=2,3$. However, the proposed method PSTR shows a 12% NDCG score improvement compared with L2RT when $K=1$. PSTR, PSTR-S and PSTR-US have similar performances. In this dataset, L2RT shows good ability to predict the relevance of noun tags. In general, semi-supervised methods are better than the unsupervised methods. LRNV gains more satisfactory results than TLTR.

On MSRC ((c) in Table 2): Since it is a toy dataset and each image has less tags (2.7 tags on average), NDCG scores stop changing when $K>6$. L2RT gains the highest NDCG scores when $K=1,2$, which further indicates its superiority in ranking the tags of nouns. When $K>2$, the proposed method PSTR performs the best among all compared methods. LRNV is also good at ranking the noun tags. Since there are only 23 tags in this dataset, some tags are absent in the supervised set, the performance gap between PSTR-S and PSTR is more evident.

Table 2
Image tag-ranking performances on (a) SUNAttribute, (b) Labelme and (c) MSRC. K is taken from 1 to 10. On three datasets, our approach outperforms all other compared baselines.

Method	$K=1$	$K=2$	$K=3$	$K=4$	$K=5$	$K=6$	$K=7$	$K=8$	$K=9$	$K=10$
(a)										
PRWTR	0.418	0.438	0.464	0.494	0.525	0.555	0.583	0.610	0.635	0.657
LRNV	0.647	0.654	0.693	0.705	0.716	0.730	0.745	0.760	0.775	0.788
TLTR	0.678	0.675	0.696	0.707	0.721	0.737	0.753	0.771	0.787	0.801
L2RT	0.639	0.649	0.681	0.692	0.703	0.717	0.731	0.747	0.761	0.776
PSTR	0.712	0.716	0.724	0.735	0.747	0.761	0.775	**0.790**	0.804	**0.817**
PSTR-S	0.694	0.707	0.716	0.722	0.732	0.746	0.760	0.776	0.791	0.804
PSTR-US	**0.715**	**0.720**	**0.727**	**0.737**	**0.748**	**0.762**	**0.776**	**0.790**	**0.805**	**0.817**
(b)										
LRNV	0.527	0.571	0.630	0.682	0.718	0.743	0.762	0.777	0.788	0.795
TLTR	0.486	0.551	0.612	0.663	0.701	0.728	0.749	0.765	0.778	0.786
L2RT	0.517	**0.596**	**0.656**	**0.698**	0.731	0.755	0.774	0.787	0.796	0.803
PSTR	**0.579**	0.594	0.653	**0.698**	**0.733**	**0.759**	**0.777**	**0.791**	**0.800**	**0.808**
PSTR-S	0.577	0.593	0.652	0.697	0.732	0.758	0.776	0.790	**0.800**	0.807
PSTR-US	0.573	0.591	0.650	0.696	0.731	0.758	0.775	0.789	0.799	0.806
(c)										
LRNV	0.678	0.776	0.876	0.884	0.885	0.886	0.886	0.886	0.886	0.886
TLTR	0.631	0.752	0.854	0.871	0.872	0.872	0.872	0.872	0.872	0.872
L2RT	**0.711**	**0.820**	0.866	0.871	0.871	0.870	0.870	0.870	0.870	0.870
PSTR	0.701	0.807	**0.884**	**0.896**	**0.897**	**0.898**	**0.898**	**0.898**	**0.898**	**0.898**
PSTR-S	0.613	0.739	0.846	0.865	0.866	0.866	0.866	0.866	0.866	0.866
PSTR-US	0.688	0.802	0.881	0.892	0.894	0.894	0.894	0.894	0.894	0.894

In summary, compared with the other methods, the proposed method shows the superior performances on the three benchmark datasets, particularly it is capable of dealing with complex situations. The improvements of PSTR compared with PSTR-S on SUNAttribute and MSRC demonstrate that introducing the unsupervised data is essential and makes sense. Furthermore, the similar performances of PSTR and PSTR-US validate that using both the supervised and unsupervised data in the binary-relevance regression term may be somewhat redundant. LRNV and L2RT show the potential to rank noun tags. Generally, the semi-supervised methods are better than the unsupervised methods.

In addition, we also calculate the top one average relevance levels for each method on the three datasets, verifying that whether the most relevant tags are ranked on the top positions. The results are presented in Fig. 3. Some exemplary ranking results on SUNAttribute are shown in Fig. 4.

4.4. Experiments of tag-based image search

In this subsection, we carry out experiments of tag-based image search, which is an important real world application. The goal of tag-based image search is to retrieve the images whose tags have been ranked when users provide some keywords (tags). The performance of tag-based image search is directly rely on how well the tags are ranked.

We choose the unsupervised part (10,000 images) of SUNAttribute as the dataset for image search, since SUNAttribute is the largest and most complex dataset. Each tag in the vocabulary is taken as the query, and all images containing the query tag are ranked based on the tag-ranking results. The criteria for measuring the relevance between the query tag q and the image $\mathbf{x}_i$ are formulated as follows [1]:

$$rel(\mathbf{x}_i) = -p_i + 1/n_i, \tag{24}$$

where p_i is the ranking position of q in the ranked tag list of image $\mathbf{x}_i$, and n_i is the number of tags of image $\mathbf{x}_i$. From (24), we can see that the ranking position determines the relevance score: the higher the rank is, the higher the score will be. When the query has the same ranking position in the two images, the one with less tags results in a higher relevance score. In this time, the images are ranked based on the different query tags. To evaluate the performance of tag-based image search, we also use the top K NDCG

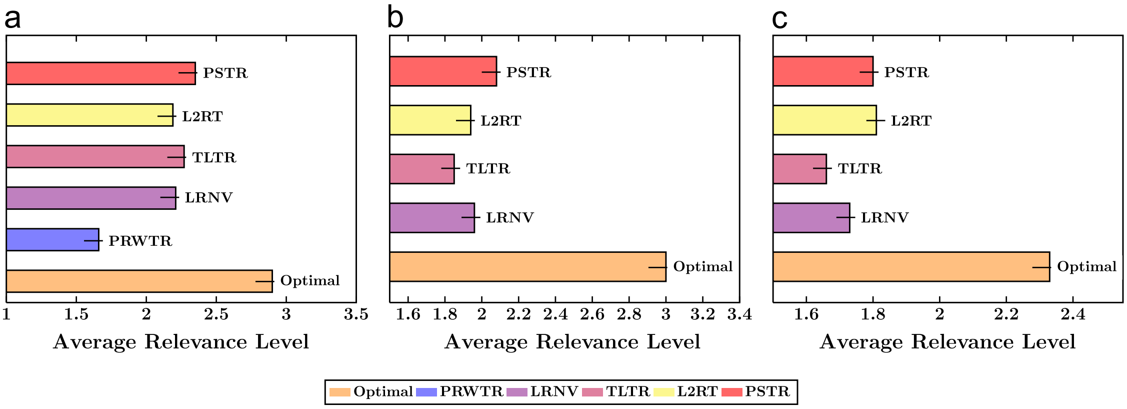
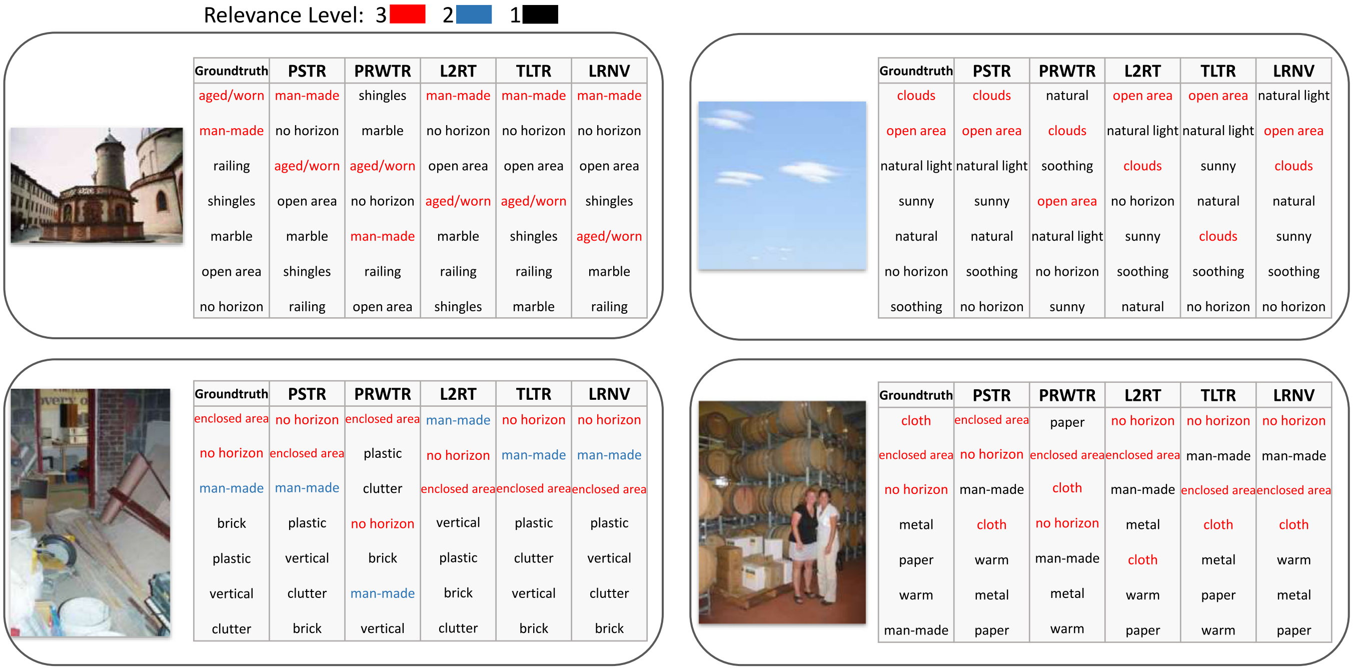


Fig. 3. Top one average relevance levels. Optimal means that the most relevant tags are ranked on the first place. Our method is better than the other methods on (a) SUNAttribute and (b) Labelme. On (c) MSRC, L2RT and our method are comparable, and both are better than the unsupervised methods.


Relevance Level: 3
2
1

Groundtruth	PSTR	PRWTR	L2RT	TLTR	LRNV
aged/worn	man-made	shingles	man-made	man-made	man-made
man-made	no horizon	marble	no horizon	no horizon	no horizon
railing	aged/worn	aged/worn	open area	open area	open area
shingles	open area	no horizon	aged/worn	aged/worn	shingles
marble	marble	man-made	marble	shingles	aged/worn
open area	shingles	railing	railing	railing	marble
no horizon	railing	open area	shingles	marble	railing

Groundtruth	PSTR	PRWTR	L2RT	TLTR	LRNV
clouds	clouds	natural	open area	open area	natural light
open area	open area	clouds	natural light	natural light	open area
natural light	natural light	soothing	clouds	sunny	clouds
sunny	sunny	open area	no horizon	natural	natural
natural	natural	natural light	sunny	clouds	sunny
no horizon	soothing	no horizon	soothing	soothing	soothing
soothing	no horizon	sunny	natural	no horizon	no horizon

Groundtruth	PSTR	PRWTR	L2RT	TLTR	LRNV
enclosed area	no horizon	enclosed area	man-made	no horizon	no horizon
no horizon	enclosed area	plastic	no horizon	man-made	man-made
man-made	man-made	clutter	enclosed area	enclosed area	enclosed area
brick	plastic	no horizon	vertical	plastic	plastic
plastic	vertical	brick	plastic	clutter	vertical
vertical	clutter	man-made	brick	vertical	clutter
clutter	brick	vertical	clutter	brick	brick

Groundtruth	PSTR	PRWTR	L2RT	TLTR	LRNV
cloth	enclosed area	paper	no horizon	no horizon	no horizon
enclosed area	no horizon	enclosed area	enclosed area	man-made	man-made
no horizon	man-made	cloth	man-made	enclosed area	enclosed area
metal	cloth	no horizon	metal	cloth	cloth
paper	warm	man-made	cloth	metal	warm
warm	metal	metal	warm	paper	metal
man-made	paper	warm	paper	warm	paper

Fig. 4. Some exemplary ranking results on SUNAttribute. We compare our method with the other methods. Different levels are labeled in different colors. We can observe that the results of our method approaches the groundtruth most. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

scores, and K varies from 1 to 200. As for the groundtruth relevance of all images, we label each image into four levels based on the relevance scores of its tags.

The proposed method is compared to the other methods with the performances shown in Fig. 5. It can be observed that our method outperforms the other methods. TLTR obtains higher average NDCG scores than PRWTR, LRNV and L2RT when $K \geq 10$, although it is lower than LRNV when $K < 10$. PRWTR has similar performance with TLTR when $K < 10$, and it is close to LRNV when $K \geq 10$. For all methods, the average NDCG scores increase stably when $K > 30$. In the scenario of image search, users only care the top ranked images in many cases. Therefore, from the perspective of users, the performances of five methods are ranked in this order: PSTR > LRNV > TLTR = PRWTR > L2RT.

4.5. Parameter selection

In this subsection, we investigate the problem of parameter selection. There are three hyper parameters in the proposed model, namely σ, β and λ. σ controls the contributions of low-level visual features and high-level semantic features for computing the image similarity. β is to penalize the unsupervised term. λ is a penalty parameter of the regularization term.

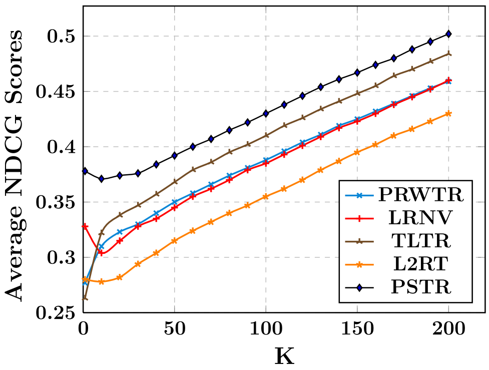


Fig. 5. Average NDCG scores of five methods for tag-based image search on the unsupervised part of SUNAttribute.

Before conducting a great deal of experiments for parameter selection, we make a brief analysis of the parameters. Different σ and the number of features used will influence the scale of the image similarities, and a proper similarity scale is important for relevance propagation. In the experiments, σ is set to $\{0.1, 0.2, 0.3, \ldots, 0.9\}$. As for β, it is to penalize the unsupervised term, and the optimal values should be empirically studied. β is also set to $\{0.1, 0.2, 0.3, \ldots, 0.9\}$. In our view, the main responsibility of λ is to guarantee the matrix in (17) is invertible. Besides, λ should be set to a relatively small value to reduce the penalty of $\mathbf{M}$ for the purpose of better fitting the data, but not too small for avoiding over fitting. Accordingly, we select λ from $\{0.001, 0.005, 0.01, 0.05, 0.1, 0.2\}$. To plot the results in 3D meshes, we have to set K to a fixed value ($K=1$ for all plots).

We take the experiments on SUNAttribute as an example to make parameter selection. The final results are shown in Fig. 6. From (a) to (c) in Fig. 6, we can see that when λ is small, the average NDCG scores vary greatly with σ. However, in Fig. 6(d)–(f), σ makes less influences when λ is relatively large. It is observed that the σ value within the interval $[0.1, 0.4]$ shows better performances than the other situations in all plots. In all plots, β has a little influence on the performances when λ and σ are fixed, and it still shows a weak tendency that small values ($[0.1, 0.5]$) can achieve slightly better results than those large values ($[0.6, 0.9]$). In summary, σ and λ are more important than β, and they should be set to proper values. The optimal parameter setting on SUNAttribute is $\sigma = 0.2$, $\beta = 0.4$ and $\lambda = 0.01$. Actually, we continue to increase λ till 0.9, but no further improvement is gained. Based on the analogous analysis, we set the parameters for the other two datasets as follows: On Labelme, $\sigma = 0.2$, $\beta = 0.1$ and $\lambda = 0.2$; On MSRC, $\sigma = 0.1$, $\beta = 0.7$ and $\lambda = 0.05$.

5. Conclusion and future work

In this paper, we proposed a semi-supervised model to address image tag-ranking problem. The proposed model takes both the pairwise supervision and the unsupervised information into consideration. The pairwise supervision can reveal the inner ranking structures by decomposing the ranked tag lists into image–tag pair

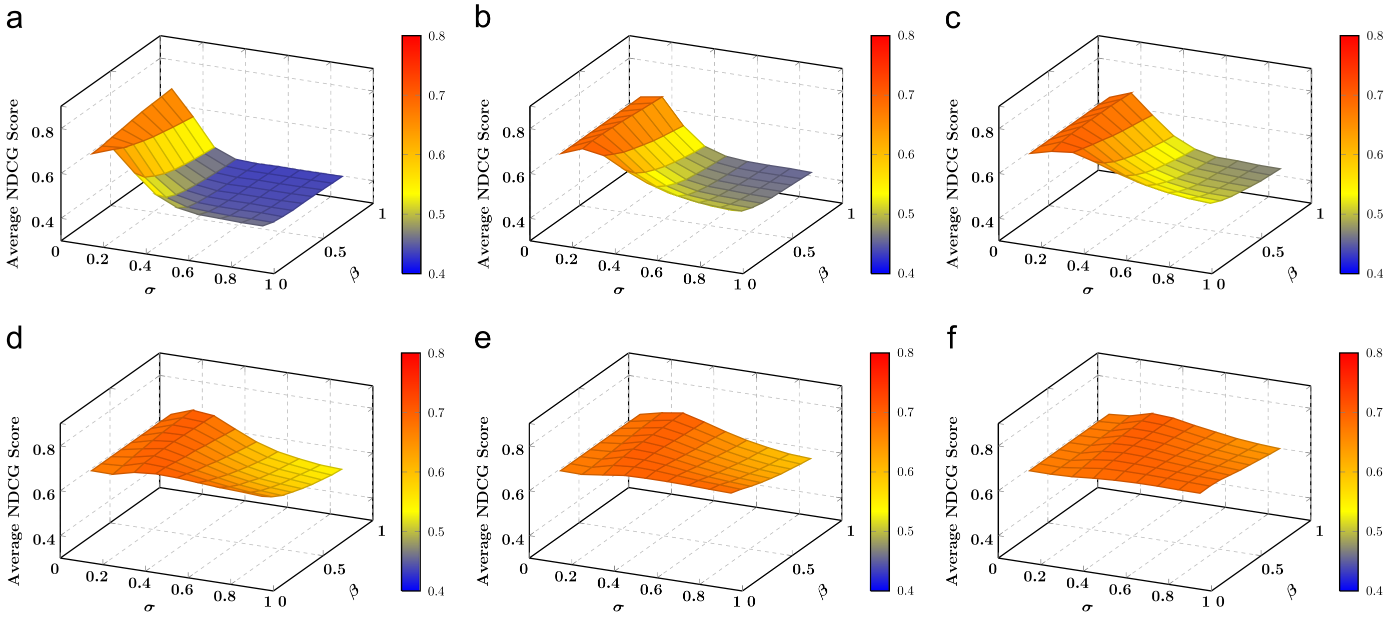


Fig. 6. Average NDCG scores with different parameters on SUNAttribute. K is fixed to 1, σ and β vary in $\{0.1, 0.2, 0.3, \ldots, 0.9\}$, λ is assigned to $\{0.001, 0.005, 0.01, 0.05, 0.1, 0.2\}$ (from (a) to (f) respectively). The high performances can be achieved with σ in $[0.1, 0.4]$, β in $[0.1, 0.5]$ and λ in $[0.01, 0.2]$.

tuples as instances for model learning. Furthermore, the unsupervised data is viewed as the weak ranking information to facilitate the model. The resulting objective function can be elegantly solved in closed form. The experiments of image tag-ranking and tag-based image search are conducted on three benchmark datasets: SUNAttribute, Labelme and MSRC. The final results firmly demonstrate the better effectiveness of the proposed method over the other state-of-the-art methods.

In our view, listwise supervision in L2R techniques is a more natural way to deal with ranking information. The essential point is to model the ranked list as a whole instead of pair tuples. Thus, next we attempt to design a listwise based loss function for the model.

Acknowledgments

This work was supported in part by the National Basic Research Program of China under Grant 2012CB316304, and the National Natural Science Foundation of China under Grants 61272331, 61305049 and 61375024.

References

[1] D. Liu, X. Hua, L. Yang, M. Wang, H. Zhang, Tag ranking, in: Proceedings of ACM International Conference on World Wide Web, 2009, pp. 351–360.

[2] G. Patterson, J. Hays, Sun attribute database: discovering, annotating, and recognizing scene attributes, in: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, 2012, pp. 2751–2758.

[3] B. Russell, A. Torralba, K. Murphy, W. Freeman, Labelme: a database and web-based tool for image annotation, Int. J. Comput. Vision 77 (1–3) (2008) 157–173.

[4] J. Shotton, J. Winn, C. Rother, A. Criminisi, Textonboost: joint appearance, shape and context modeling for multi-class object recognition and segmentation, in: Proceedings of European Conference on Computer Vision, 2006, pp. 1–15.

[5] X. Li, C. Snoek, M. Worring, Learning tag relevance by neighbor voting for social image retrieval, in: Proceedings of ACM International Conference on Multimedia Information Retrieval, 2008, pp. 180–187.

[6] J. Zhuang, S. Hoi, A two-view learning approach for image tag ranking, in: Proceedings of ACM International Conference on Web Search and Data Mining, 2011, pp. 625–634.

[7] F. Sun, H. Li, Y. Zhao, X. Wang, D. Wang, Towards tags ranking for social images, Neurocomputing 120 (2013) 434–440.

[8] Z. Wang, J. Feng, C. Zhang, S. Yan, Learning to rank tags, in: Proceedings of ACM International Conference on Image and Video Retrieval, 2010, pp. 42–49.

[9] T. Lan, G. Mori, A max-margin riffled independence model for image tag ranking, in: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, 2013, pp. 3103–3110.

[10] L. Hang, A short introduction to learning to rank, IEICE Trans. Inf. Syst. 94 (10) (2011) 1854–1862.

[11] M. Guillaumin, T. Mensink, J. Verbeek, C. Schmid, Tagprop: discriminative metric learning in nearest neighbor models for image auto-annotation, in: Proceedings of IEEE International Conference on Computer Vision, 2009, pp. 309–316.

[12] H. Wang, H. Huang, C. Ding, Image annotation using multi-label correlated green's function, in: Proceedings of IEEE International Conference on Computer Vision, 2009, pp. 2029–2034.

[13] Y. Verma, C. Jawahar, Image annotation using metric learning in semantic neighbourhoods, in: Proceedings of European Conference on Computer Vision, 2012, pp. 836–849.

[14] C. Wang, S. Yan, L. Zhang, H. Zhang, Multi-label sparse coding for automatic image annotation, in: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, 2009, pp. 1643–1650.

[15] M. Wang, F. Li, M. Wang, Collaborative visual modeling for automatic image annotation via sparse model coding, Neurocomputing 95 (2012) 22–28.

[16] C. Wang, F. Jing, L. Zhang, H. Zhang, Image annotation refinement using random walk with restarts, in: Proceedings of ACM International Conference on Multimedia, 2006, pp. 647–650.

[17] G. Zhu, S. Yan, Y. Ma, Image tag refinement towards low-rank, content-tag prior and error sparsity, in: Proceedings of ACM International Conference on Multimedia, 2010, pp. 461–470.

[18] C. Wang, F. Jing, L. Zhang, H. Zhang, Content-based image annotation refinement, in: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, 2007, pp. 1–8.

[19] W. Zhao, Z. Guan, Z. Liu, Ranking on heterogeneous manifolds for tag recommendation in social tagging services, Neurocomputing 148 (2015) 521–534.

[20] J. Huang, C. Guestrin, Riffled independence for ranked data, in: Advances in Neural Information Processing Systems, 2009, pp. 799–807.

[21] S. Robertson, S. Walker, Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval, in: Proceedings of ACM SIGIR International Conference on Research and Development in Information Retrieval, 1994, pp. 232–241.

[22] L. Page, S. Brin, R. Motwani, T. Winograd, The pagerank citation ranking: bringing order to the web.

[23] R. Herbrich, T. Graepel, K. Obermayer, Large margin rank boundaries for ordinal regression, 1999, pp. 115–132.

[24] D. Hardoon, S. Szedmak, J. Shawe-Taylor, Canonical correlation analysis: an overview with application to learning methods, Neural Comput. 16 (12) (2004) 2639–2664.

[25] J. Xiao, J. Hays, K. Ehinger, A. Oliva, A. Torralba, Sun database: large-scale scene recognition from abbey to zoo, in: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, 2010, pp. 3485–3492.

[26] T. Chua, J. Tang, R. Hong, H. Li, Z. Luo, Y. Zheng, Nus-wide: a real-world web image database from National University of Singapore, in: Proceedings of ACM Conference on Image and Video Retrieval, 2009.

[27] A. Oliva, A. Torralba, Modeling the shape of the scene: a holistic representation of the spatial envelope, Int. J. Comput. Vision 42 (3) (2001) 145–175.

[28] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, 2005, pp. 886–893.

[29] E. Shechtman, M. Irani, Matching local self-similarities across images and videos, in: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, 2007, pp. 1–8.

[30] F. Li, P. Pietro, A Bayesian hierarchical model for learning natural scene categories, in: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, vol. 2, 2005, pp. 524–531.

[31] S. Hwang, K. Grauman, Reading between the lines: object localization using implicit cues from image tags, IEEE Trans. Pattern Anal. Mach. Intell. 34 (6) (2012) 1145–1158.

[32] S. Hwang, K. Grauman, Accounting for the relative importance of objects in image retrieval, in: Proceedings of British Machine Vision Conference, 2010, pp. 1–12.

[33] K. Järvelin, J. Kekäläinen, Cumulated gain-based evaluation of ir techniques, ACM Trans. Inf. Syst. 20 (4) (2002) 422–446.

Yonghao He received the B.E. degree in software engineering from Sichuan University, in 2011. Currently, he is a fourth-year Ph.D. student in National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China. His research interests include machine learning, particularly deep learning and its applications related to multimedia.

Cuicui Kang received the B.S. degree in computer science from Beijing Jiaotong University, China, in 2010, along with the Excellent Graduate Award in Beijing. Then she was admitted to the Institute of Automation, Chinese Academy of Sciences (CASIA), Beijing, China, for the Ph.D. study. Currently, she is a graduate student in CASIA and supervised by Professor Pan. Her research interests include pattern recognition and machine learning for multimedia.

Jian Wang received the B.E. degree in School of Automation from Huazhong University of Science and Technology, in 2013. Currently, he is a second-year master student in National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China. His research interests include machine learning and multimedia retrieval.

Shiming Xiang received the B.S. degree in mathematics from Chongqing Normal University, China, in 1993, the M.S. degree from Chongqing University, China, in 1996, and the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences, China, in 2004. From 1996 to 2001, he was a Lecturer with the Huazhong University of Science and Technology, Wuhan, China. He was a postdoctorate with the Department of Automation, Tsinghua University, Beijing, China, until 2006. He is currently a Professor with the Institute of Automation, Chinese Academy of Sciences, Beijing. His research interests include pattern recognition and machine learning.

Chunhong Pan received his B.S. degree in automatic control from Tsinghua University, Beijing, China, in 1987, his M.S. degree from Shanghai Institute of Optics and Fine Mechanics, Chinese Academy of Sciences, China, in 1990, and his Ph.D. degree in pattern recognition and intelligent system from Institute of Automation, Chinese Academy of Sciences, Beijing, in 2000. He is currently a professor at the Institute of Automation, Chinese Academy of Sciences. His research interests include computer vision, image processing, computer graphics, and remote sensing.