MLRank: Multi-correlation Learning to Rank for image annotation

Zechao Li, Jing Liu*, Changsheng Xu, Hanqing Lu

National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Zhongguancun East Road 95, Haidian District, Beijing 100190, PR China

1. Introduction

With the prevalence of digital image archives, the number of digital images has been growing rapidly in recent years. Thus, how to index and search for these images effectively and efficiently is an increasingly urgent research issue in the multimedia community. Although many Content-Based Image Retrieval (CBIR) systems, such as IBM QBIC [1], Columbia VisualSEEK [2] and MIT Photobook [3], have been proposed, it is rather difficult for users to represent their queries using the abstract image features such as color and texture. Instead, most users prefer to search images using textual queries, i.e., keyword-based image search. In light of this, automatic image annotation, targeting to assign relevant keywords to images, has also attracted a lot of research attention recently.

For image annotation, most previous work take their attention on learning the semantics of an image from tagged or noisily tagged images, in which the relevance ranking of tags to an image is an essential issue. In contrast, the ranking values of tags can also be convenient to image search or image tag recommendation, etc. It is a new path to formulate the image annotation problem into a learning to rank framework. Existing approaches on learning to rank are mainly applied in the field of information retrieval, and they are usually limited to rank objects with independent assumption and use only content information. Obviously, the assumption is unsuitable to the case of image annotation, since images and tags are two types of correlated objects in visual appearance and semantic meanings respectively. Therefore, in this paper we focus on learning to rank such inter- and intra-correlated objects by exploring visual features and tagging information of images.

On the other hand, the conventional learning to rank is a supervised learning approach which requires sufficient labeled training data. However, the collection of labeled data for image annotation is time-consuming and expensive. With the proliferation of Web 2.0 technology, we can freely collect billions of unlabeled or noisily labeled data from media sharing portals (e.g., Flickr, YouTube and Zooomr). Thus, how to benefit from the large-scale, low-cost, but noisy data is necessary for image annotation.

Motivated by above issues, we propose Multi-correlation Learning to Rank (MLRank), a semi-supervised learning framework for image annotation, in which the visual similarity among images and the semantic relevance among tags are explored simultaneously. From Fig. 1, we observe that visually similar images often reflect similar semantic themes and typically share similar tags, and vice versa. The observation has been explored as an important prior [4–6]. We call the two dual cases as the image-bias consistency (from visual similarity to tag relevance) and the tag-bias consistency (from tag relevance to visual similarity) respectively, and formulate them as two constraints in the optimization problem for rank learning. To obtain an explicit solution of the ranking model, we relax the optimization problem in two manners by attaching the constraints corresponding to the image-bias and tag-bias consistency with different sequential orders respectively, which lead to a uniform ranking model. Experimental results show that the proposed MLRank method outperforms the state-of-the-arts on three benchmarks including Corel5K, IAPR TC12 and NUS-WIDE.
relaxed optimization, in which the both constraints are incorporated with exchangeable sequential orders respectively. But fortunately, the two relaxation problems bring a uniform learning model. Through the optimization solution, the ranking function is learned in a semi-supervised manner, where the labeled and unlabeled images are jointly explored, and the unlabeled images are automatically annotated through the learned ranking function. Finally, we conduct extensive experiments on three benchmarks, namely Corel5K, IAPR TC12 and NUS-WIDE. MLRank achieves the better performance compared with some related work.

The key contributions of this work are summarized as follows.

1. We formulate the problem of image annotation as a Multi-correlation Learning to Rank framework, in order to rank the relevance of tags to an image. In the framework, inter- and intra-correlations among images and tags are jointly explored.

2. We consider the consistency between "visual similarity" and "tag relevance" as constraints of the optimization for learning to rank, which ensures the correct correspondences between images and tags, thus leads to the satisfied annotation performance.

3. We present a semi-supervised learning framework by leveraging the labeled images and unlabeled images to estimate the ranking function for image annotation, which makes the proposed method capable of benefiting from the large-scale web data.

The reminder of the paper is organized as follows. In Section 2, we introduce some previous work focusing on image annotation and learning to rank. Section 3 presents the formulation of the proposed MLRank for image annotation. The optimization of MLRank is detailed in Section 4. In Section 5, we detail the experimental settings, including the datasets, feature extraction, parameter setting and evaluation measure. Experimental results and discussions are given in Section 6. Finally some concluding remarks of this paper and the future work are given in Section 7.

2. Previous work

2.1. Image annotation

The image annotation problem has been extensively studied in recent years and many approaches have been proposed to address this problem. As claimed in [7], most of previous methods attempt to annotate images by leveraging possible available tagging information (i.e., partial image–tag relation, ITR), image-to-image relation (IIR) and tag-to-tag relation (TTR) individually or simultaneously. Accordingly, we will briefly review the related work according to their different leveraged resources as follows.

First, there are some work to exploit ITR [8,9]. In [8], visual features and tags of images are used to estimate the latent space via probabilistic Canonical Correlation Analysis (CCA). It is a high speed and high accuracy method. However, it utilizes only the labeled images and overlooks IIR and TTR. In [9], a video retrieval method is developed by exploiting the correlation between image, sound and location information, which is efficient and scalable to large-scale multi-modal data. However, the associated tag information is neglected, which may degrade the performance. In addition, it does not utilize the inter-correlation.

Second, some methods explore IIR to propagate tags among images. Probabilistic modeling methods, such as CMRM [10], CRM [11] and MBRM [12], attempt to model such correlations over all tagged images with different representations of ITR and IIR. Wang et al. [13] proposed to assign labels to unlabeled data based on a fraction of labeled data to obtain density estimator. Graph-based inference with integrated multiple/single instance representations [14] is to propagate tags by considering two types of image similarity corresponding to the global representations and the region representations of the image sets. Nearest neighbor based models are local learning techniques. Examples of such models contain using label diffusion over a similarity graph of labeled and unlabeled images [15,16] and learning discriminative models in neighborhoods of testing images [4]. Image annotation is treated as a retrieval problem in a nearest-neighbor keyword transfer mechanism [17]. It linearly combines several distances computed from different visual features to determine nearest neighbors. TagProp [18] is a trained nearest neighbor model that predicts tags of test images using a weighted combination of the tag absence/presence among neighbors. Different from them, the best ranking of tags to an image is designed as our targets. And multiple correlations are exploited simultaneously in a semi-supervised framework. Our method can also explore the information underlying the huge amount of unlabeled data.

In addition, there are some efforts considering tag correlation in the annotation process, such as Coherent Language Model [19], Correlated Label Propagation [20], annotation refinement using random walk [21] and WordNet-based method [22]. Liu et al. [23] proposed a semi-supervised multi-label learning method to exploit unlabeled data and category correlation based on constrained non-negative matrix factorization algorithm. But its computational cost is prohibitive. To annotate images, we jointly exploit available tagging information (i.e., partial image–tag correlation), image correlation and tag correlation to learn an efficient ranking function in a semi-supervised framework.

To effectively annotate an image, some efforts are conducted on jointly exploring ITR, IIR and TTR [24,25,7,26]. Liu et al. [7]...
explicitly demonstrated such an idea by proposing a graph-learning framework for image annotation, in which two sequential steps of learning processes are conducted, namely image-based graph learning for “basic image annotation” and word-based graph learning for “annotation refinement”. Li et al. [26] proposed a Multi-correlation Probabilistic Matrix Factorization (MPMF) model, which factorizes the image-tag relation matrix, image similarity matrix and tag correlation matrix simultaneously by the shared latent matrices and improves the performance of image annotation.

The above methods mainly focus on learning semantics of an image, while the best ranking of tags to an image is rarely considered as their designing targets. Our method is distinct from the traditional methods in that we will design to learn an explicit tag ranking function for image annotation while fusing the image-tag relation information, image similarity information and tag correlation information simultaneously.

2.2. Learning to rank

The importance of the ranking problem has inspired numerous approaches to handle it, especially in the context of information retrieval. One typical direction of rank learning is formulated as ordinal regression [27]. Another popular direction is investigated to transform the problem of ranking into that of classification and apply existing classification techniques to perform the task, such as RankSVM [28], RankBoost [29] and RankNet [30]. More recently, directly defining a loss function on a list of objects and directly optimizing the loss function has been proposed [31–33]. This approach tends to be more effective because it formalizes the ranking problem in a more straight way. As in [31–33], they assume that the objects for ranking are independent. In practice, this is not the true case, because the ranking documents (images or tags) are usually correlated on content (semantics) or possible links. Such correlation should be considered in the ranking process.

Some other learning to rank methods have been studied for image retrieval and annotation. A ranking-based distance metric learning method [34] is proposed based on the rich textual information of web images and applied to content based image retrieval and search-based image annotation. Clickthrough data collected from search logs is exploited to learn concept classifiers for image annotation [35]. A learning to rerank algorithm [36] is proposed to employ Gaussian process regression to predict the normalized click count for each image, and combine it with the original ranking score to rerank the baseline results. Parikh et al. [37] proposed to learn a linear ranking function for each attribute and used it to predict the relative strength of each attribute in images. Different from them, our method focuses on designing a new learning to rank model for image annotation in a semi-supervised framework. It is novel to jointly exploit multi-correlations to learn a ranking function.

Although the idea of learning to rank has been extensively exploited in information retrieval, related work in image annotation are still limited. Moreover, the typical learning to rank in retrieval is unsuitable to the case of image annotation, since visual similarity among images and semantic relevance among tags are in neglectable attributes in image annotation. Towards this end, we propose a novel semi-supervised learning to rank model, which incorporates visual similarity and semantic relevance simultaneously.

3. Multi-correlation Learning to Rank

In this paper, we formulate the image annotation as a problem of tag ranking for an image, develop a novel method, referred to as Multi-correlation Learning to Rank, which jointly integrates inter- and intra-correlations among images and tags, and introduce a semi-supervised learning framework to leverage easily available unlabeled data.

Suppose we have \( n \) images and \( m \) tags. Consider the data \((x_i, y_i)\), where \( x_i = (x_{id}, \ldots, x_{id}) \) and \( y_i = (y_{1i}, \ldots, y_{mi}) \) are the regressors and responses for the \( i \)-th observation. Let \( X \in \mathbb{R}^{n \times d} \) be the feature matrix with each row \( x_i \) corresponding to the \( d \) dimensional feature vector of the \( i \)-th image. Let \( Y = (y_1, \ldots, y_m) \in \mathbb{R}^{n \times m} \) denote the relationships between images and tags with \( y_{im} \in \{0, 1\} \) denoting the absence/presence of tag \( w \) for image \( i \). Let \( S \in \mathbb{R}^{n \times n} \) denote the visual similarity matrix for these \( n \) images and \( C \in \mathbb{R}^{m \times m} \) represent the correlation matrix between these \( m \) tags.

In the following sections, we first present the basic formulation, and then elaborate how to learn the correlation constraint ranking function. Finally, we present how to define the matrices \( S \) and \( C \). The solution for the optimization problem will be detailed in Section 4.

3.1. Problem formulation

We propose a novel semi-supervised learning to rank method for image annotation considering all possible information. The learned ranking function is used to predict tags for any untagged image.

First, we specify the notations. The feature matrix \( X \) contains labeled image features and unlabeled image features. The ground truth matrix is \( Y = [Y_1^{\text{labeled}}, Y_1^{\text{unlabeled}}] \), where \( Y_1^{\text{labeled}} \) is the ground truth of labeled images and \( Y_1^{\text{unlabeled}} \) is defaulted to be a zero matrix.

We define the ranking function as \( f(h(X), C, S) \), indicating the ranking relevance of tags to each image, by exploring image visual features \( X \), image similarity matrix \( S \) and tag relevance matrix \( C \). And \( h \) is the ranking function only with content information \( X \). Simply, the ranking based on relations is defined in \( f \), while the ranking based on contents is defined in \( h \). In practice, there are three cases for learning the nested ranking function:

1. inner function \( h \) and outer function \( f \) are unknown and to be learned;
2. inner function \( h \) is predefined and outer function \( f \) is to be learned;
3. inner function \( h \) is to be learned and outer function \( f \) is predefined.

In this paper, we focus on the case 3. The goal of learning is to obtain the best function \( f \) using all possible information. Specifically, we define the ranking function as

\[
Y = f(h(X; W), S, C).
\] (1)

Obviously we make use of visual content information of images and correlations within images and tags in the ranking function. There are many forms for the inner function \( h \). For simplicity, we consider the linear function \( h(X) = XW \), where \( W \) is an unknown parameter matrix. Then, the learning problem is defined as the following optimization with respect to \( W \):

\[
\min_W l(Z, Y),
\] (2)

where \( Y \) denotes the ground truth, \( L \) represents the loss function, and \( Z = f(h(X; W), C, S) \) is the ranking function. The loss function \( L \) has many forms. In this paper, for simplicity, we define the loss function as

\[
L(Z, Y) = \frac{1}{2} \| Z - Y \|_F^2.
\] (3)

Here \( \| \cdot \|_F \) is the Frobenius norm. It is noted that \( Y \) contains the unlabeled data, we only need to fit the labeled data. Hence, we
change Eq. (3) to
\[
I(Z, Y) = \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{m} \ell_{ij}(Y) - Y_{ij}^2 = \frac{1}{2} \|Y \odot (Z - Y)\|_F^2. 
\]
(4)
where \(\odot\) indicates the point-wise product of matrices and \(\ell_{ij}^p\) is the indicator function that is equal to 1 if image \(i\) is labeled and 0 otherwise. In order to avoid overfitting, a regularization term is added into Eq. (4). Hence we have
\[
\min_w E(X, C, S, W) = \frac{1}{2} \|Y \odot (Z - Y)\|_F^2 + \frac{\lambda_w}{2} \|W\|_F^2, 
\]
(5)
where \(\lambda_w > 0\) is a nonnegative trade-off parameter. To solve the above optimization problem, the ranking function \(Z\) should be defined. As aforementioned, in the ranking function \(Z = f(h(X, W), C, S)\), the outer function of \(f\) is predefined and the inner function of \(h\) with the parameter matrix \(W\) is not. The central problem is how to define the outer function of \(f\) to integrate the multiple correlations.

3.2. Correlation constraint ranking function

As mentioned in case 3, we should define the outer ranking function \(f\) with some reasonable constraints. We observe that semantically related or co-occurring tags (generally called as “relevant tags”) are often used to annotate similar images containing similar scenes or objects. The observation defined as the tag-bias consistency can be incorporated as a constraint for the ranking function of \(f\). In converse, visually similar images often describe similar schemes and typically share similar tags. As the dual version of the tag-bias consistency, the image-bias consistency is also reasonable to constraint the ranking function of \(f\). Apart from the two consistencies, the ranking difference between the upper ranking and the inner ranking should not be too large. Integrating these considerations, we define the outer function \(f\) as the solution to minimize the linear combination of the three objectives.

\[
f(h(X, W), C, S) = \arg \min_Z l_1(h(X, W), Z) + \lambda C l_2(Z, C) + \lambda S l_3(Z, S),
\]
(6)
where \(Z\) denotes ranking score matrix with each row \(z\), corresponding to the relational scores between the \(i\)-th image and all tags, the first objective \(l_1(h(X, W), Z)\) measures the ranking difference between \(Z\) and \(h\), \(l_2(Z, C)\) and \(l_3(Z, S)\) measure the tag-bias consistency and the image-bias consistency, respectively. The parameters \(\lambda_C\) and \(\lambda_S\) are non-negative, representing the trade-off among these three objectives.

Simply, we define the first objective \(l_1(h(X, W), Z)\) as
\[
l_1(h(X, W), Z) = \sum_{i=1}^{n} \|h(X, W) - Z_i\|^2 = \|h(X, W) - Z\|_F^2.
\]
(7)
where \(\| \cdot \|\) denotes \(L_2\) norm.

The \((i, j)\) element of \(Z_{ij}\), \(Z_{ij,p}\), is the ranking score of the \(i\)-th image with respect to the \(j\)-th tag. The ranking matrix \(Z\) has a two-view interpretation. Each row vector of \(Z\) is the ranking vector over the set of tags and each column of \(Z\) is the ranking vector over the set of images. The constraints \(l_2(Z, C)\) and \(l_3(Z, S)\) coincide with the local geometrical structures in the concept space and visual space, respectively. That is, if two images are visually similar, their ranking vectors over tags should be similar, and vice versa.

Tag-bias correlation consistency: The tag-bias correlation consistency is that if two tags are very relevant semantically, their ranking vectors over images are supposed to be similar. For the \(l\)-th and \(p\)-th tags, the entry \(C_{lp}\) denotes their semantic relevancy. Their difference between the ranking vectors over image is computed by \(\|Z_l - Z_p\|_2\). The tag-bias correlation consistency between them is defined as \(\min C_{lp} \|Z_l - Z_p\|_2^2\). The rationale behind is that we can guarantee that if two tags are semantic relevant, their ranking vectors over images should be similar. That is to say, the larger the value of \(C_{lp}\) is, the closer the relational values \(Z_l\) and \(Z_p\) are.

The tag-bias correlation consistency over all tags is defined as
\[
\min \frac{1}{2} \sum_{i=1}^{m} \sum_{p=1}^{m} C_{lp} \|Z_l - Z_p\|_2^2 = \text{Tr}(ZC^TZ^T).
\]
(8)
where \(L^C = D^C - C\) is the Laplacian matrix of the tag relationship graph and \(\text{Tr}[]\) denotes the trace operation on a matrix. \(D^C\) is a diagonal matrix with \(D^C_{ii} = \sum C_{ip}\). Thus, the second term corresponding to tag correlation in Eq. (6) is stated as
\[
l_2(C, Z, S) = \text{Tr}(ZL^CZ^T).
\]
(9)

Image-bias correlation consistency: Similarly, the image-bias correlation consistency among images can be constrained by minimizing the following objective function:
\[
\frac{1}{2} \sum_{i=1}^{n} \sum_{p=1}^{m} S_{ij} \sum_{p=1}^{m} (Z_{ij,p} - Z_{ij,p})^2 = \text{Tr}(Z^T L^S Z).
\]
(10)
where \(L^S = D^S - S\) is the Laplacian matrix of the image similarity graph. \(D^S\) is defined similar to \(D^C\). Thus, the third objective in Eq. (6) can then be defined as follows based on Eq. (10):
\[
l_3(S, Z, L) = \text{Tr}(Z^T L^S Z).
\]
(11)

With Eqs. (7), (9) and (11) above mentioned, the optimization problem Eq. (6) can be rewritten as
\[
f(h(X, W), C, S, W) = \arg \min_{Z} \|h(X, W) - Z\|_F^2 + \lambda C \text{Tr}(ZL^CZ^T) + \lambda S \text{Tr}(Z^T L^S Z).
\]
(12)
We will detail the optimization of the above problem in Section 4.

3.3. Correlation estimation

In this subsection, we introduce how to estimate the image correlation and tag correlation.

Tag correlation matrix \(C\): There are many forms to define the tag correlation, such as tag co-occurrence in the labeled dataset, WordNet-based correlation (WNC) [38] and the approach in [7]. In this work, we combine the global correlation and local correlation to estimate the tag correlation. For the global correlation \(C^G\), we adopt a concurrence based method to estimate the tag similarity, which is analogous to the principle of the normalized Google distance [39]. We first calculate the semantic distance \(d_{lp}\) between tags \(t_l\) and \(t_p\) as
\[
d_{lp} = \max(\log q_l, \log q_p) - \log q_{lp},
\]
(13)
where \(q_l\) is the number of images containing tag \(l\) and \(d_{lp}\) is the number of images containing both tags \(l\) and \(p\). Such numbers are obtained by searching “tags only” on Google image search website using the tags as queries. In addition, \(T\) is the total number of the images in Google image. [1]. Then, the global semantic similarity matrix \(C^G\) is defined as
\[
C^G_{lp} = e^{-d_{lp}/s_\varepsilon},
\]
(14)
where \(s_\varepsilon\) is empirically set as the median value of all the \(d_{lp}, l, p = 1, \ldots, m, m\). Generally speaking, two tags with high co-occurrence in the labeled dataset will lead to high probability to annotate certain
image jointly, such as “sea” and “beach”, “sky” and “cloud”. Therefore, the tag co-occurrence becomes an informative representation of tag correlation. In this paper, we define the local correlation $C_{lp}$ as the tag co-occurrence in the labeled dataset.

$$C_{lp} = \frac{2q_{lp}}{q_{l} + q_{p}}.$$  \hspace{1cm} (15)

where $q_{l}$, $q_{p}$ and $q_{lp}$ are obtained similar to the normalized Google distance but on the labeled dataset.

Now we have obtained the two types of tag correlations, i.e., the global correlation and the local correlation. They have different characteristics. The local correlation provides relatively precise statistical description, but it depends on the labeled dataset. The global correlation is universal and independent on any corpus, but it may contain some noises due to the diversity of the web data. To make them complement each other, we first normalize them into $[0, 1]$ respectively and then unify them in a linear form as

$$C_{lp} = \alpha C_{lp}^g + (1-\alpha)C_{lp}^l,$$  \hspace{1cm} (16)

where $\alpha$ is a parameter within the interval $[0, 1]$. In this paper, we set $\alpha = 0.4$ empirically.

**Image similarity matrix $S$:** We represent the similarity relationship between images in an undirected graph. The graph is constructed with $n$ vertices, where each vertex corresponds to a data point $x_i$. For each data point $x_i$, we find its $k$ nearest neighbors and put edges between $x_i$ and its neighbors. Such kind of graph is widely used in semi-supervised learning and clustering. There are also many choices to define the similarity matrix $S$ based on the similarity graph. Three of the most commonly approaches to define $S$ on the graph are as follows:

1. **0–1 similarity:** $S_{ij} = 1$ if and only if nodes $i$ and $j$ are connected by an edge. This is the simplest method and is very easy to compute.

2. **Heat kernel similarity:** It is a popular approach utilized for image data. Heat kernel is based on Euclidean distance and has an intrinsic connection to the Laplace Beltrami operator on differentiable function on a manifold [40]. It is defined as

$$S_{ij} = \left\{ \begin{array}{ll} e^{-\|x_i-x_j\|^2/\sigma^2} & \text{if nodes } i \text{ and } j \text{ are connected,} \\ 0 & \text{otherwise} \end{array} \right.$$  \hspace{1cm} (17)

Here the $\sigma$ is the median value of all the Euclidean distances.

3. **Dot-product similarity:** If nodes $i$ and $j$ are connected, we set

$$S_{ij} = x_i^T x_j.$$  \hspace{1cm} (18)

Note that if $x_i$, $i = 1, \ldots, n$ is normalized to 1, the dot product of two vectors is equivalent to the cosine similarity of the two vectors.

Since in this paper the $S_{ij}$ is employed to measure the closeness of two points, we do not treat the different weighting schemes separately and use the heat kernel similarity for constructing the $k$-nearest neighbor graph for simplicity.

### 4. Problem optimization

In this section, we first introduce the optimization of the ranking function problem (as in Eq. (12)) and then describe the process of solving $W$ (as in Eq. (5)) and the computational complexity analysis.

#### 4.1. Optimization for ranking function

First, for clarity, let $l(Z)$ denote the objective function in Eq. (12).

$$l(Z) = \|h(X; W) - Z\|^2 + \lambda_c \text{Tr}(ZL^c Z^T) + \lambda_s \text{Tr}(ZL^s Z).$$  \hspace{1cm} (19)

To minimize $l(Z)$, we calculate the derivative $l(Z)$ with respect to $Z$ and obtain

$$\frac{\partial l(Z)}{\partial Z} = -2(h(X; W) - Z) + 2\lambda_c L^c + 2\lambda_s L^s Z.$$  \hspace{1cm} (20)

We set $\frac{\partial l(Z)}{\partial Z} = 0$ and get

$$Z[Z^T + \lambda_c L^c] + [L^s + \lambda_s L^s] Z = h(X; W),$$  \hspace{1cm} (21)

where $C$ and $F$ are $m \times m$ and $n \times n$ identity matrices, respectively. The above equation is a classical matrix equation (Sylvester equation).

To the best of our knowledge, it cannot be solved to obtain the explicit form of $Z = AWB$, where $A$ and $B$ are matrices with the suitable size.

As above mentioned, it leads to a complex problem when we explore the image-bias consistency and tag-bias consistency simultaneously to learn the ranking model. Therefore, we have to relax the optimization problem of the ranking model by considering the tag-bias consistency and image-bias consistency in a sequential manner. Then we obtain two kinds of relaxation problems. The first relaxation is to consider the ranking difference as the first item in Eq. (12) and the tag-bias consistency to get an initial ranking result, and then append the image-bias consistency to further update the ranking result. The second relaxation has the similar pattern, but with exchanged roles of the two kinds of correlation consistency.

Firstly, we consider the first relaxation problem. In the ranking model, we first learn an initial ranking function $Z_1 = f_1(h(X; W), C)$ by exploring the ranking difference between $h$ and $Z_1$ and the tag-bias consistency, and then update $Z_1$ to obtain the final ranking function $Z = f(h(X; W), C, S)$ by deliberating the difference between $Z$ and $Z_1$ and the image-bias consistency. Thus, the initial ranking function is formulated as

$$f_1(h(X; W), C) = \arg \min_{Z_1} l(h(X; W), Z_1) + \lambda_c l_c(Z_1).$$  \hspace{1cm} (22)

Through the similar inference as above, setting the derivative of the above objective with respect to $Z_1$ to 0, we obtain

$$2(Z_1 - h(X; W)) + 2\lambda_s Z_1 = 0.$$  \hspace{1cm} (23)

Thus, we have

$$Z_1 = (\Gamma^c + \lambda_c L^c)^{-1} h(X; W).$$  \hspace{1cm} (24)

Similarly, because $\lambda_c$ is non-negative, matrix $\Gamma^c + \lambda_c L^c$ is diagonally dominant and $\Gamma^c + \lambda_c L^c$ is invertible according to the Levy–Desplanques theorem. Therefore, the explicit form of the initial ranking model is obtained as follows:

$$f_1(h(X; W), C) = Z_1 = h(X; W)(\Gamma^c + \lambda_c L^c)^{-1}. $$  \hspace{1cm} (25)

Then, we update the initial ranking model by solving the following optimization problem:

$$\min_Z \|Z - Z_1\|^2_2 + \lambda_s l_s(S, Z).$$  \hspace{1cm} (26)

Setting the derivative of the above objective with respect to $Z$ to 0, we have

$$2(Z - Z_1) + 2\lambda_s L^s Z = 0.$$  \hspace{1cm} (27)

$$(\Gamma^c + \lambda_c L^c)^{-1} Z = Z_1.$$  \hspace{1cm} (28)

Since $\lambda_c$ is non-negative, thus matrix $\Gamma^c + \lambda_c L^c$ is diagonally dominant and it is invertible. In this way, we obtain the explicit form of the final ranking model as follows:

$$Z = f(h(X; W), C, S) = (\Gamma^c + \lambda_c L^c)^{-1} Z_1 = (\Gamma^c + \lambda_c L^c)^{-1} h(X; W)(\Gamma^c + \lambda_c L^c)^{-1}. $$  \hspace{1cm} (29)

Second, as to the second relaxed problem, through the similar inference as the first relaxation, we can get

$$Z_2 = (\Gamma^c + \lambda_c L^c)^{-1} h(X; W).$$  \hspace{1cm} (30)
\[ Z = Z_2 (I^2 + \lambda_L L^2)^{-1} = (I^2 + \lambda_L L^2)^{-1} h(X, W) (I^2 + \lambda_L L^2)^{-1}. \] (31)

Compared Eqs. (29) and (31), it is observed that the two relaxation problems lead to the same ranking function. Thus, Eq. (29) is the ranking model to be learned.

4.2. Optimize \( W \)

Next, we solve the optimization problem Eq. (5) with the obtained ranking model. The optimization problem minimizes the sum-of-squared-errors objective function with quadratic regularization terms. Gradient based approaches can be applied to find a local minimum.

\[ \frac{\partial E}{\partial W} = X^T [I^2 + \lambda_L L^2]^{-1} [I^2 + \lambda_L L^2]^{-1} h(X, Y) [I^2 + \lambda_L L^2]^{-1} + \lambda_W W \] (32)

Since \( S \) and \( C \) are both symmetric, \( L^2 \) and \( L^2 \) are also symmetric.

4.3. Computational complexity analysis

In this subsection, we discuss the extra computational cost of our proposed algorithm. The common way to express the complexity of one algorithm is using \( O \) notation.

For \( n \) images with \( d \) dimensional feature and \( m \) tags, the main cost of the computation comes from matrix inversion. First, suppose that the matrix inversion has been obtained. The time cost to compute the ranking model is \( O(n^2d + ndm + nm^2) \). For each iteration, the time complexity is \( O(n^2d + ndm + nm^2) \). Suppose the updates stops after \( t \) iterations, the time cost is \( O(tn^2d + ndm + nm^2) \).

Now let us analyze the time complexity of matrix inversion. The time complexity of straightforwardly computing the ranking model is of order \( O(mn^2 + m^2n) \). The main cost of the computation comes from matrix inversion. In reality, if \( I^2 + \lambda_L L^2 \) is a banded matrix with bandwidth \( k + n \), then \( Z \) in Eq. (28) can be solved with time complexity \( O(n) \). If \( S \) is a banded matrix, \( I^2 + \lambda_L L^2 \) is also a banded matrix with the same bandwidth as that of \( S \). In practice, we only consider the \( k \) nearest neighborhoods of one object. Hence \( S \) becomes a sparse matrix, which has at most \( k \) non-zero values in each row and each column. A sparse matrix can be converted to a banded matrix with linear time by Gibbs–Poole–Stockmeyer [41]. In this way, the time complexity becomes \( O(n + m) \).

To construct the \( k \)-nearest neighbor graph, MLRank also needs \( O(n^2d) \). To compute the tag correlation, it also needs \( O(m^2) \). Thus, the overall cost for MLRank is \( O(tm^2d + ndm + nm^2 + n^2d) \). In our work, we have \( n = m \) and \( n = d \) and then the overall cost becomes \( O(tn^2d) \).

5. Experimental setting

In this section, we describe our experimental settings, including data sets, feature extraction, parameters’ setting and evaluation measures.

5.1. Datasets

We consider three publicly available benchmarks [Corel5K [42], IAPR TC12 [43] and NUS-WIDE [44]] that have been widely used in previous work, and convenient to direct comparison. We present some statistics of the three datasets in Table 1.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Corel5K</th>
<th>IAPR TC12</th>
<th>NUS-WIDE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vocabulary size</td>
<td>260</td>
<td>291</td>
<td>81</td>
</tr>
<tr>
<td># of training images</td>
<td>4493</td>
<td>17,665</td>
<td>15,667</td>
</tr>
<tr>
<td># of testing images</td>
<td>499</td>
<td>1962</td>
<td>15,659</td>
</tr>
<tr>
<td>Words per img.</td>
<td>3.4/5</td>
<td>5.7/23</td>
<td>2.6/15</td>
</tr>
<tr>
<td>Img. per word</td>
<td>58.6/1004</td>
<td>347/4999</td>
<td>511/4373</td>
</tr>
</tbody>
</table>

5.2. Parameter setting

In our method, there are some parameters to be set. To combine the tag global correlation and local correlation in Eq. (16), we set \( \alpha = 0.4 \) empirically. To construct the \( k \)-nearest neighbor graph for image similarity, the number of nearest neighbors \( k \) is set to \( \log(n) \), where \( n \) is the number of the images. The regularization parameter \( \lambda_W \) is set to 0.1 by cross-validation. In addition, the parameters \( \lambda_C \) and \( \lambda_S \) play very important roles in our method. They control how much our method should incorporate the information of the tag correlation and the image similarity. Thus we will discuss the impacts of them based on the experimental evaluations.

5.3. Evaluation measures

To compare with previous work, we evaluate our approach with standard evaluation measures as them. We evaluate the performance per keyword, and then average over keywords.

First, we employ recall, precision and F1 to measure the annotation performance for fixed annotation length. Given a concept \( w \), let \( |\mathcal{C}_w| \) be the number of images annotated with this image descriptors: gist features and color histograms with 16 bins in each color channel for LAB and HSV representations, which are got from [47].

IAPR TC12: It is a collection of 19,627 images of natural scenes that include different sports and actions, photographs of people, animals, cities and many other aspects of contemporary life. It is regarded as the one suitable for testing the scalability of annotation algorithms [46,48]. To compare with previous work [12,17,18,46], we select 17,665 training images and 1962 images for testing. The features utilized for IAPR TC12 are same to Corel5K downloaded from [47].

NUS-WIDE: It is a web image dataset with images from Flickr. There are 55,615 images with 1000 tags, in which 81 tags are provided the ground-truth from [44]. The 81 tags have semantic meanings and are referred to as concepts by the dataset creators. We select a subset, focusing on images containing at least 5 tags and at least 1 concept, and obtain a collection of 31,336 images. We choose 15,677 images as labeled images and the rest 15,659 as unlabeled images. We make use of the 81 concepts to annotate the unlabeled images and evaluate the performance. To verify the performance stability of the proposed method, we vary the percentage of labeled images selected from the labeled images. We denote it as \( r \) with varying it 10% to 100% increased by a step of 10% in implementation. The sampled labeled images are then amalgamated with the whole set of unlabeled images. The visual features are got from [49], which contain four types of global features: 64-D color histogram (LAB), 144-D color auto-correlation (HSV), 73-D edge direction histogram and 128-D wavelet texture, and grid-based local features: 225-D block-wise color moments (LAB). Thus, we sequentially combine these 5 groups into 634-D features.
keyword $w$ in the ground truth, $|RM|$ be the number of annotated images with the same keyword of our algorithm, and $|RC|$ be the number of correct annotations of our annotation algorithm. We employ the $R$ and $P$ to denote the recall and precision respectively. Recall, precision and $F1$ are defined as $R = |RC|/|RC|$, $P = |RC|/|RM|$, $F1 = 2 \times R \times P / (R + P)$. Let $N+$ denote the number of keywords with non-zero recall value.

Second, the ranking order provided with image annotation is very important for image retrieval. To evaluate the ranking order, we analyze results with single-word queries. Average Precision ($\text{AvgP}$) is the standard measure used for retrieval benchmark. It corresponds to the average of the precision at each position where a relevant image appears. Let $P(k)$ measures the percentage of relevant images within the top $k$ positions of the ranking.

$$\text{AvgP} = \frac{1}{|W|} \sum_{w \in W} P(\text{rank}(l, w)),$$

where $\text{rank}(l, w)$ is the rank of image $l$ for concept $w$. The mean average precision (mAP) over concepts is got by averaging $\text{AvgP}$ over all concepts.

6. Experimental analysis

In this section we systematically evaluate the effectiveness of our proposed MLRank. We will present the performance of our method for image annotation and image retrieval as well as compare it with the related method on the three datasets. Over and above, the impact of the tag-bias consistency and the image-bias consistency will also be discussed. Note that for all the compared methods including the proposed method, we use the same experimental protocol.

6.1. Results for the Corel5K dataset

Firstly, how the changes of $\lambda_c$ and $\lambda_s$ can affect the performance for image annotation is analyzed. These two parameters balance the trade-off among image-bias and tag-bias correlation consistency. The impacts of $\lambda_c$ and $\lambda_s$ in terms of $F1$ on the Corel5K dataset are shown in Fig. 2. We observe that the values of $\lambda_c$ and $\lambda_s$ impact the image annotation results significantly, which demonstrates that incorporating the tag-bias consistency and image-bias consistency in the ranking model greatly improve the performance. From the results, we can see that no matter $\lambda_c$ and $\lambda_s$, as the value increases, the $F1$ values increase at first, but when it exceeds a certain threshold, the $F1$ values decrease with further increase of the value of $\lambda_c$ or $\lambda_s$. This phenomenon confirms with the intuition that integrating visual content information, tag correlation and image similarity simultaneously can generate better performance than purely using one or two of them. When $\lambda_c = 0.01$ and $\lambda_s = 1$, our method achieves the best result. Thus, we set $\lambda_c = 0.01$ and $\lambda_s = 1$ in the following experiments on the Corel5K dataset.

Then we conduct experiments to evaluate the performance of MLRank in terms of image annotation and image retrieval on the Corel5K dataset. The performance is measured by $P$, $R$, $F1$, $N+$ and mAP. In order to further demonstrate the effectiveness of our method, we also investigate the performance of some selected previous methods: CRM [11], MBRM [12], SML [45], LASSO [17], JEC [17], MSC [25], TagProp [18], TGLM [7], CLS [8], MPMF [26] and SSK + CBKP [16]. It is worth to note that JEC [17] is a new baseline for image annotation proposed in 2008. For TagProp, we report the result of its two versions as in [18]. ML (integrating metric learning) and its modulated extension $\sigma$ML (introducing word-specific logistic discriminant models). The average results are presented in Table 2.

From the results, we can draw the following conclusions. First, MLRank outperforms all annotation algorithms except $\sigma$ML [18] in terms of $P$, $R$, $F1$ and $N+$). It is reasonable since we focus on ranking tags for any image and do not create models for single tag, $\sigma$ML adopts complex visual features and trains word-specific models, which makes it overly dependent on the training data. It can be revealed from the results of ML. If it does not introduce word-specific logistic discriminant models, our method outperforms it since for rare tags, the training data is not sufficient. Additionally, more features and more word models make its higher computational complexity. Second, the performance of the proposed MLRank is better than other graph based methods, i.e., TGLM [7], CLS [8], MPMF [26] and SSK + CBKP [16]. It indicates that our method can better propagate the relations between images and tags. Third, based on the simple feature representation, MLRank is superior to MSC [25] and GS [46], which adopt complex visual feature processing procedure. Finally, MLRank achieves the best performance in terms of mAP, which demonstrates that the ranking order obtained from MLRank is preferable. To the best of our knowledge, the results of TagProp [18] are the best in the literature. However, our method is superior to TagProp in terms of mAP. Our method captures the consistency between visual similarity and tag relevance, which enables to obtain better ranking of tags to any given image.

![Fig. 2. Performance on the Corel5K dataset of our method in terms of F1 with respect to the parameters $\lambda_s$ and $\lambda_c$.](image-url)

Table 2

<table>
<thead>
<tr>
<th>Method</th>
<th>Image annotation</th>
<th>Image retrieval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P$</td>
<td>$R$</td>
<td>$F1$</td>
</tr>
<tr>
<td>CRM [11]</td>
<td>0.16</td>
<td>0.19</td>
</tr>
<tr>
<td>MBRM [12]</td>
<td>0.24</td>
<td>0.25</td>
</tr>
<tr>
<td>SML [45]</td>
<td>0.23</td>
<td>0.29</td>
</tr>
<tr>
<td>LASSO [17]</td>
<td>0.24</td>
<td>0.29</td>
</tr>
<tr>
<td>JEC [17]</td>
<td>0.27</td>
<td>0.32</td>
</tr>
<tr>
<td>MSC [25]</td>
<td>0.25</td>
<td>0.32</td>
</tr>
<tr>
<td>ML [18]</td>
<td>0.31</td>
<td>0.37</td>
</tr>
<tr>
<td>$\sigma$ML [18]</td>
<td>0.33</td>
<td>0.42</td>
</tr>
<tr>
<td>GS [46]</td>
<td>0.30</td>
<td>0.33</td>
</tr>
<tr>
<td>TGLM [7]</td>
<td>0.25</td>
<td>0.29</td>
</tr>
<tr>
<td>CLS [8]</td>
<td>0.29</td>
<td>0.32</td>
</tr>
<tr>
<td>MPMF [26]</td>
<td>0.27</td>
<td>0.34</td>
</tr>
<tr>
<td>SSK + CBKP [16]</td>
<td>0.29</td>
<td>0.33</td>
</tr>
<tr>
<td>MLRank</td>
<td>0.32</td>
<td>0.37</td>
</tr>
</tbody>
</table>
6.2. Results for the IAPR TC12 dataset

For the IAPR TC12 dataset, we also first tune the parameters \( \lambda_C \) and \( \lambda_S \) to find a better trade-off among the visual content information, tag-bias consistency and image-bias consistency. The impacts of \( \lambda_C \) and \( \lambda_S \) in terms of F1 are shown in Fig. 3. From the results, it is observed that these two parameters have the similar effects on F1 for the IAPR TC12 dataset as for the Corel5K dataset. \( \lambda_C = 0.01 \) and \( \lambda_S = 1 \) gain the best result. Hence \( \lambda_C = 0.01 \) and \( \lambda_S = 1 \) are set in the remainder of experiments on the IAPR TC12 dataset.

The performance of the proposed MLRank is tested and compared with MBRM, LASSO, JEC, TagProp, GS and MPMF, since there are no results reported by other methods. Table 3 presents the compared results. For reference we also include our implementation of MPMF [26]. Again we find that our method gain better results for the image annotation task than other methods except sML and the best result for the image retrieval task. The advantages of the proposed MLRank are again demonstrated.

6.3. Results for the NUS-WIDE dataset

Next we present the performance of the proposed MLRank with different values of \( \lambda_C \) and \( \lambda_S \) and the experimental results compared with previous work on the NUS-WIDE dataset. There are results reported of some graph-based semi-supervised learning methods, such as linear neighborhood propagation (LNP) [50], Entropic Graph Semi-Supervised Classification (EGSSC) [51], Sparse Graph-based Semi-Supervised Learning (SGSSL) [52] and Large-Scale Multi-label Propagation (LSMP) [53]. In this paper, we compare MLRank only with LSMP since LSMP and SGSSL are superior to LNP and EGSSC [51,53] and LSMP outperforms ESSSC significantly [53]. Besides, we also compare our method with k-Nearest-Neighbors (KNN) and MPMF [26]. For these methods, we implement them and tune the corresponding parameters to find the optimal values. The results with varying the parameter \( \tau \) are presented in Fig. 4. As done in [53], we conduct the following experiments with \( \tau = 100\% \). The impacts of \( \lambda_C \) and \( \lambda_S \) are analyzed in Fig. 5 and the compared results are shown in Table 4. The APs of these four methods for these 81 concepts are illustrated in Fig. 6.

From the results above, it is observed that MLRank always achieves the best results when selecting different proportions of labeled set and outperforms the other baseline algorithms significantly. The improvement is supposed to stem from the fact that our model integrates both content and relational information among images and tags and introduces the consistency between

---

Table 3
Image annotation and retrieval performance comparison on the IAPR TC12 dataset.

<table>
<thead>
<tr>
<th>Method</th>
<th>Image annotation</th>
<th>Image retrieval</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( P )</td>
<td>( R )</td>
</tr>
<tr>
<td>MBRM [12]</td>
<td>0.24</td>
<td>0.23</td>
</tr>
<tr>
<td>LASSO [17]</td>
<td>0.28</td>
<td>0.29</td>
</tr>
<tr>
<td>JEC [17]</td>
<td>0.28</td>
<td>0.29</td>
</tr>
<tr>
<td>ML [18]</td>
<td>0.48</td>
<td>0.25</td>
</tr>
<tr>
<td>sML [18]</td>
<td>0.46</td>
<td>0.35</td>
</tr>
<tr>
<td>GS [46]</td>
<td>0.32</td>
<td>0.29</td>
</tr>
<tr>
<td>MPMF</td>
<td>0.31</td>
<td>0.28</td>
</tr>
<tr>
<td>MLRank</td>
<td>0.38</td>
<td>0.32</td>
</tr>
</tbody>
</table>

---

Table 4
Image annotation and retrieval performance comparison on the NUS-WIDE dataset with 81 concepts.

<table>
<thead>
<tr>
<th>Method</th>
<th>Image annotation</th>
<th>Image retrieval</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( P )</td>
<td>( R )</td>
</tr>
<tr>
<td>KNN</td>
<td>0.12</td>
<td>0.17</td>
</tr>
<tr>
<td>MPMF</td>
<td>0.20</td>
<td>0.26</td>
</tr>
<tr>
<td>LSMP</td>
<td>0.21</td>
<td>0.27</td>
</tr>
<tr>
<td>MLRank</td>
<td>0.24</td>
<td>0.28</td>
</tr>
</tbody>
</table>
Fig. 6. The comparison of APs for the 81 concepts using the four methods.

Fig. 7. Convergence curve of MLRank on Corel5K, IAPR TC12 and NUS-WIDE datasets.

Fig. 8. The annotated results versus human annotations for images from Corel5K, IAPR TC12 and NUS-WIDE datasets. The tags are predicted using MLRank and aligned by the ranking values.
visual similarity and tag relevance. Our method enables to assign the best ranking of tags to any given image.

6.4. Convergence study

The updating rules for mining the objective function of MLRank are essentially iterative. Here we investigate how fast these rules can converge. Fig. 7 presents the convergence curves of MLRank on all the three datasets. For each figure, the vertical axis is the value of objective function and the horizontal axis denotes the iteration number. We can see that MLRank converges very fast, usually within 10 iterations.

6.5. Discussion

We have investigated the performance of the proposed MLRank and the reason that it can achieve better results that other methods. It is concluded that the proposed MLRank can successfully leverage the visual content information, the image relation information and the tag relation information. It performs image annotation and retrieval very well simultaneously, which dovetails our intuition that MLRank can obtain the best ranking of tags to an image, and capture the consistency between visual similarity and tag relevance. Besides, the update rule for MLRank converges very fast.

We also present some exemplary results of image annotation from these three datasets in Fig. 8. Each image is shown with the predicted tags and human annotations. The difference in predicted tags are marked in italic font. The tags marked in red italic font are the groundtruth but not predicted by MLRank while the tags marked in blue italic font are predicted by MLRank but not the groundtruth. In all of our experiments, five tags are transferred to each image. Thus there may be some redundancies that may misinterpret images. However, in most cases they can explain the images well, such as “grass” in the first image.

7. Conclusions and future work

We propose an effective Multi-correlation Learning to Rank (MLRank) framework, where the visual similarity among images and the semantic relevance among tags are explored simultaneously. Generally, two key points are addressed in our work. First, we propose a semi-supervised learning to rank algorithm to leverage the labeled data and unlabeled data to estimate the ranking function. Second, we formulate the consistency between visual similarity and tag relevance into the ranking function learning process. Extensive experiments on image annotation and retrieval over three public data sets have demonstrated that the proposed method is competitive with current methods. In future, we will consider to introduce other prior information such as adding some type of regularization on the inner function \( f \) and the inner function \( h \) in training.

Conflict of interest statement

None declared.

Acknowledgments

This work was supported by 973 Program (2010CB327905) and National Natural Science Foundation of China (61272329, 61071014, and 61202325).

References

Zechao Li received the B.E. degree from University of Science and Technology of China (USTC), Anhui, China, in 2008. He is currently pursuing the Ph.D. degree at National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China. His current research interests include correlation mining, social multimedia analysis and applications.

Jing Liu received the B.E. and M.E. degrees from Shandong University, Shandong, in 2001 and 2004, respectively, and the Ph.D. degree from the Institute of Automation, Chinese Academy of Sciences, Beijing, in 2008. She is an Associate Professor with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. Her current research interests include correlation mining, multimedia. She published over 30 papers in those areas.

Changsheng Xu received the Ph.D. degree from Tsinghua University, Beijing, China in 1996. He is a Professor of Institute of Automation, Chinese Academy of Sciences, and Executive Director of China-Singapore Institute of Digital Media. He was with Institute for Infocomm Research, Singapore from 1998 to 2008. He was with the National Lab of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences from 1996 to 1998. His research interests include multimedia content analysis, indexing and retrieval, digital watermarking, computer vision and pattern recognition. He published over 180 papers in those areas.

Hangqing Lu received his B.S. and M.S. from department of computer science and department of electric engineering in Harbin institute of technology in 1982 and 1985. He got his Ph.D. from department of electronic and information science in Huazhong University of sciences and technology. He is a Professor with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. Current research interests include Image similarity measure, Video Analysis, Multimedia Technology and System. He published over 200 papers in those areas.