PL-ranking: A Novel Ranking Method for Cross-Modal Retrieval

Liang Zhang[1,3], Bingpeng Ma[1,2,3]*, Guorong Li[1,3], Qingming Huang[1,2,3], Qi Tian[4]
[1] University of Chinese Academy of Sciences, China
[2] Key Lab of Intell. Info. Process., Inst. of Comput. Tech., CAS, China
[3] Key Laboratory of Big Data Mining and Knowledge Management, CAS, China
[4] Department of Computer Science, University of Texas at San Antonio, TX, 78249, USA
zhangliang14@mails.ucas.ac.cn {bpma, liguorong, qmhuang}@ucas.ac.cn
qitian@cs.utsa.edu

ABSTRACT

This paper proposes a novel method for cross-modal retrieval named **P**airwise-**L**istwise **ranking** (PL-ranking) based on the low-rank optimization framework. Motivated by the fact that optimizing the top of ranking is more applicable in practice, we focus on improving the precision at the top of ranked list for a given sample and learning a low-dimensional common subspace for multi-modal data. Concretely, there are three constraints in PL-ranking. First, we use a pairwise ranking loss constraint to optimize the top of ranking. Then, considering that the pairwise ranking loss constraint ignores class information, we further adopt a listwise constraint to minimize the intra-neighbors variance and maximize the inter-neighbors separability. By this way, class information is preserved while the number of iterations is reduced. Finally, low-rank based regularization is applied to exploit the correlations between features and labels so that the relevance between the different modalities can be enhanced after mapping them into the common subspace. We design an efficient low-rank stochastic subgradient descent method to solve the proposed optimization problem. The experimental results show that the average MAP scores of PL-ranking are improved 5.1%, 9.2%, 4.7% and 4.8% than those of the state-of-the-art methods on the Wiki, Flickr, Pascal and NUS-WIDE datasets, respectively.

Keywords

Multi-modal analysis; Cross-modal retrieval; Subspace learning; Learning to rank

1. INTRODUCTION

With the rapid growth of multi-modal data, including image, text, video and audio, cross-modal retrieval has been widely studied in recent years [7, 9, 13, 15, 17, 18, 20, 21, 22, 26, 27]. The key problem for cross-modal matching is how to push relevant samples from another modality appearing at the top of the ranked list when we give a query sample from one modality. This motivates that the techniques of the learning to rank have become increasingly popular, which can exploit the correlation shared by different modalities. These methods optimize the top of ranking by maximizing a criterion (*e.g.*, MAP or NDCG) related to the ultimate retrieval performance.

The most successful ranking method in cross-modal retrieval may be bi-directional cross-media semantic representation model (Bi-CMSRM), which optimizes ranking performance directly [27]. Bi-CMSRM is based on the structural SVM and optimized by using the 1-Slack cutting plane algorithm, and it has shown good performance in cross-modal retrieval. However, despite using an efficient convex method to solve the dual problem,Bi-CMSRM has shown the weaknesses on the scalability to large, high-dimensional dataset [16]. Besides, Bi-CMSRM only focuses on learning optimal mappings but ignores the structure of mappings such that it can not further exploit the label relevance between the different modalities.

In this paper, we propose an efficient ranking method for cross-modal retrieval named PL-ranking. PL-ranking integrates the weighted approximate rank pairwise (WARP) loss[1], listwise loss[2] and low-rank constraint into a generic minimization formulation, and then is optimized by extending the recently proposed FAST-SSGD [8]. By this way, PL-ranking not only optimizes the top of ranking, but also effectively captures the label correlations as well as scales to high-dimensional and large datasets. Thus, we can effectively retrieve relevant samples by searching in a small neighborhood of the query sample. Specifically, there are three important components contained in PL-ranking.

We first extend WARP to bi-directional WARP (bWARP) such that the learned model can be applied to image-query-texts and text-query-images simultaneously. Since both the directions of retrieval are optimized in the training period, bWARP ensures that the different modalities are projected

*Corresponding author.

[1] The pairwise ranking method takes the sample pairs as the training instances and formulates the ranking as a task of learning a classification or regression model from the collection of the pairwise instances of samples.

[2] The listwise information reflects the class relation of multiple samples, *e.g.*, intra-class and inter-class relations.

MM '16, October 15-19, 2016, Amsterdam, Netherlands
© 2016 ACM. ISBN 978-1-4503-3603-1/16/10...$15.00
DOI: http://dx.doi.org/10.1145/2964284.2964336

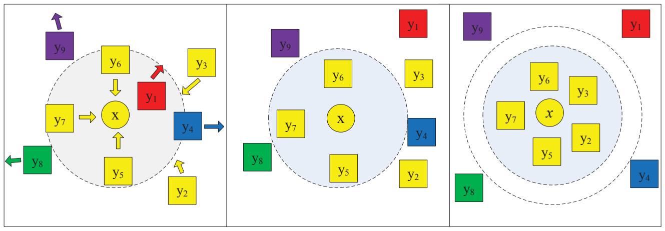
into the same semantic space such that the generalization performance is improved. What's more, by assigning the larger values for the first few larger losses, bWARP guarantees that the relevant samples appear at the top of the ranked list for the two directions of retrieval.

Furthermore, we adopt a listwise constraint for preserving the class information and reducing the iteration number of optimization. WARP uses the pairwise ranking loss to approximate the total ranking loss, and the paired samples are obtained by a random sampling trick. This will lose the class information due to the randomness in sampling. By calculating the nearest intra-neighbors and inter-neighbors in each sampling, the listwise constraint preserves the class information. Considering that the class information is beneficial to the retrieval task, the performance of PL-ranking can be improved. Besides, compared with WARP, the iteration number can be reduced because more positive samples and violators[3] are found in each iteration.

In Fig.1, we provide a simple matching illustration of pairwise ranking method and the proposed PL-ranking. The yellow circle represents an image query $\mathbf{x}$, and the red square represents $\mathbf{x}$'s first violator from the text modality. The yellow squares represent $\mathbf{x}$'s six-nearest intra-neighbors from the text modality, and the other squares represent $\mathbf{x}$'s nearest inter-neighbors. Fig.1(b) shows the matching result of pairwise ranking method [10]. Since the image query $\mathbf{x}$ and its violator $\mathbf{y}_1$ are used to optimize the pairwise ranking loss, they can be separated as far as possible. However, this method ignores the class information, so the nearest intra-neighbors $\mathbf{y}_2$ and $\mathbf{y}_3$ still have the large separation with the image query $\mathbf{x}$ while the distance among the different classes like $\mathbf{y}_2$ and $\mathbf{y}_4$ can not be enlarged. As shown in Fig.1(c), PL-ranking also enlarges the margin between the image query $\mathbf{x}$ and its violator $\mathbf{y}_1$ by using the pairwise ranking constraint. Moreover, the margin between the different classes can be enlarged since the listwise constraint minimizes the variance between $\mathbf{x}$ and its five nearest intra-neighbors and maximizes the separability between $\mathbf{x}$ and the other four nearest inter-neighbors.

To effectively exploit the semantic information embedded in the high-dimensional feature space, we further adopt a nuclear norm based the regularization to learn the low-rank mapping matrices. The low-rank based regularization of the mappings can effectively exploit the correlations between labels and features because the label/feature relevance vectors can be explicitly described via the left/right singular vectors of the mapping [11]. Then the relevance between the heterogeneous features can be enhanced in the low-dimensional subspace. Besides, the low-rank constraint also enhances the mappings' scalability to the high-dimensional data.

Finally, a novel low-rank optimization framework is designed to efficiently solve the complex minimization problem. PL-ranking is optimized by extending FAST-SSGD [8]. FAST-SSGD combines the low-rank stochastic subgradients with the efficient incremental SVD so that the number of iterations is greatly reduced. Extensive experiments have been performed on four public datasets, Wiki[20], Flickr[2], Pascal VOC 2007 [4] and NUS-WIDE [2]. Experimental results show that PL-ranking outperforms several state-of-the-art methods.



(a) ORIGINAL DISTRIBUTION (b) PAIRWISE RANKING (c) PROPOSED METHOD

Figure 1: A simple illustration of the common subspace learnt by pairwise ranking method and the proposed PL-ranking. Shapes represent modalities. The same color indicates relevant samples. The circle represents an image query, and squares denote the retrieved text documents. The different shapes with the yellow color belong to same semantic class, and the squares with other colors (*e.g.*, green and purple) represent the text documents from the other classes. The red square represents the first violator found in the sampling stage.

2. RELATED WORK

Since the problem of cross-modal retrieval is considered as an important component in some real applications, extensive research have been proposed to learn a common subspace for image and text modalities in the recent past.

One popular direction of methods obtain the common subspace by learning the maximal correlations among different modalities [5, 9, 12, 17, 18, 19, 20, 21]. The most classic method is canonical correlation analysis (CCA) [9], which learns paired linear transformations to maximize the correlations between two modalities. As a supervised extension of CCA, generalized multiview analysis (GMA) [21] solves a joint, relaxed quadratic constrained program over the different feature spaces to obtain a single subspace.

An alternative direction of methods learn the common subspace by manifold learning [14, 15, 25]. The manifold learning methods project the different modalities into a common manifold by learning their underlying manifold representation due to that high dimensional data may embed in a lower dimensional intrinsic space. Mao *et al.*[15] project the relevance between two different modalities into a latent semantic space during the process of manifold alignment.

Different from the aforementioned categories of approaches which do not maximize a criterion related to the ultimate retrieval performance, another direction of methods use the technique of learning to rank [1, 6, 24, 27, 28] for learning the common subspace. These methods obtain the common subspace by the large margin learning with certain ranking criteria. Passive-aggressive model for image retrieval (PAMIR) [6] learns a discriminative retrieval model by using certain ranking criterion. Different from the uni-directional ranking method like PAMIR, Wu *et al.*[27] propose a general cross-modal ranking algorithm Bi-CMSRM, which obtains a latent space embedding via learning the structural large margin to optimize the bi-directional listwise ranking loss.

In this paper, we propose a novel ranking method, which integrates pairwise ranking loss, listwise loss and low-rank constraint into a generic minimization formulation. The proposed method jointly optimizes the ultimate retrieval performance directly and scales to large, high-dimensional dataset.

[3] Violator denotes the first negative sample which is more relevant than the positive sample in the sampling stage.

3. PAIRWISE-LISTWISE RANKING

In this section, we first present our proposed framework of PL-ranking for cross-modal problem and then introduce an efficient method to solve the optimization. Lastly, we provide the convergence analysis.

3.1 Formulation of PL-ranking

Based on the idea of latent space embedding, we focus on learning two linear mappings $\mathbf{U}$ and $\mathbf{V}$. They can map the image and text modalities into a common subspace, in which the samples from the different modalities are much closer to each other within the same class while the samples from the different classes are separated as far as possible.

We are given a training set of s paired samples, with s images from $\mathbf{X}$ and s texts from $\mathbf{Y}$. It is assumed that m and n are the dimension of image and text samples, respectively. Then the correlation between an image $\mathbf{x} \in \mathbb{R}^m$ and a text $\mathbf{y} \in \mathbb{R}^n$ can be measured by a linear mapping function as:

$$f(\cdot) = (\mathbf{U}^T\mathbf{x})^T(\mathbf{V}^T\mathbf{y}), \tag{1}$$

where T denotes the transpose of a matrix or vector, $\mathbf{U} \in \mathbb{R}^{m\times c}$ is the linear mapping matrix which maps the image $\mathbf{x}$ from the image space into the common subspace, and $\mathbf{V} \in \mathbb{R}^{n\times c}$ is the mapping which maps text $\mathbf{y}$ from the text space into the common subspace. c is the dimensionality of the common subspace. It is to be noted that c is equal to the rank of $\mathbf{U}$ and $\mathbf{V}$. In the following, $f_{\mathbf{x}}(\mathbf{y})$ measures the correlation between an image query $\mathbf{x}$ and its retrieved text $\mathbf{y}$, and $f_{\mathbf{y}}(\mathbf{x})$ measures the correlation between a text query $\mathbf{y}$ and its retrieved image $\mathbf{x}$.

To learn the optimal mappings, we formulate our objective function by integrating the pairwise ranking loss, listwise loss and nuclear based regularization into a generic minimization problem as follows:

$$\min_{\mathbf{U}\succeq 0, \mathbf{V}\succeq 0} L(\mathbf{U}, \mathbf{V}) + \lambda\Psi(\mathbf{U}, \mathbf{V}) + \gamma\Omega(\mathbf{U}, \mathbf{V}), \tag{2}$$

where $L(\mathbf{U}, \mathbf{V})$ is the sum of the pairwise losses, which adopts a sampling trick to optimize the top of ranking; $\Psi(\mathbf{U}, \mathbf{V})$ is the sum of the listwise losses, which is applied to preserve the class information by constraining the nearest intra-neighbors and nearest inter-neighbors; $\Omega(\mathbf{U}, \mathbf{V})$ is a low-rank constraint, which is used to enforce the relevance of mapped data by exploring the low-dimensional subspace structures embedded in data. $\lambda > 0$ and $\gamma > 0$ are the trade-off parameters to balance the weights of the different constraints. Since the left/right singular vectors of the larger singular values are more effective than those of the small values on strengthening the label correlation and feature correlation, we set $\mathbf{U} \succeq 0$ and $\mathbf{V} \succeq 0$ to ensure the non-negative singular values.

Pairwise ranking loss constraint: Recently, the ranking based methods obtain the good performance in information retrieval [3, 10, 23]. These methods validate that the top of ranking can be obtained by giving the bigger weights to the larger losses. Especially, Ordered Weighted Pairwise Classification (OWPC) [23] is proposed to optimize a few larger loss functions for ranking. To make the training time efficient, WARP [10] adopts a novel sampling trick to approximate the ranking loss of OWPC.

Motivated by the fact that the top of ranking is more applicable in practice, we first focus on improving the precision of the top of ranked list for a given sample. Similar to WARP, we use the pairwise ranking loss to ensure that the relevant samples appear at the top of ranked list. However, since cross-modal retrieval includes image-query-texts and text-query-images, it is imperative to design a single learned model which can be applied to two directions of retrieval at the same time. Thus, we extend WARP to bWARP for ensuring that image and text modalities are projected into the same semantic space. In this paper, we define the bi-directional pairwise ranking loss as:

$$L(\mathbf{U}, \mathbf{V}) = \sum_{\mathbf{x}\in\mathbf{X}} L(\mathbf{x}, \mathbf{Y}; \mathbf{U}, \mathbf{V}) + \sum_{\mathbf{y}\in\mathbf{Y}} L(\mathbf{y}, \mathbf{X}; \mathbf{U}, \mathbf{V}), \tag{3}$$

where $(\mathbf{x}, \mathbf{y})$ is a sample pair from two different modalities, and they belong to the same class. $L(\mathbf{x}, \mathbf{Y}; \mathbf{U}, \mathbf{V})$ denotes the ranking texts from an image query $\mathbf{x}$, and $L(\mathbf{y}, \mathbf{X}; \mathbf{U}, \mathbf{V})$ denotes the ranking images from a text query $\mathbf{y}$.

For each sample $\mathbf{x}$, $L(\mathbf{x}, \mathbf{Y}; \mathbf{U}, \mathbf{V})$ is defined as the sum of the WARP losses:

$$L(\mathbf{x}, \mathbf{Y}; \mathbf{U}, \mathbf{V}) = \sum_{\mathbf{x}\in\mathbf{X}, \mathbf{y}^+\in\mathbf{Y}_x^+} E[\mathcal{L}(\lfloor \frac{s-1}{s_{\mathbf{x}}} \rfloor) \times [g_{\mathbf{x}}(\mathbf{y})]_+, \tag{4}$$

where $g_{\mathbf{x}}(\mathbf{y})$=$1 + f_{\mathbf{x}}(\bar{\mathbf{y}}^-) - f_{\mathbf{x}}(\mathbf{y}^+)$; $\mathbf{Y}_{\mathbf{x}}^+$ and $\mathbf{Y}_{\mathbf{x}}^-$ denote the relevant and irrelevant text sets of the image query $\mathbf{x}$, respectively; $E[\cdot]$ denotes the mathematical expectation; $\lfloor\cdot\rfloor$ denotes the floor function and $[\cdot]_+ := \max(0, \cdot)$; s_x is the number of sampling that we obtain the first negative sample $\bar{\mathbf{y}}^- \in \mathbf{Y}_{\mathbf{x}}^-$ which satisfies $1 + f_{\mathbf{x}}(\bar{\mathbf{y}}^-) > f_{\mathbf{x}}(\mathbf{y}^+)$, and $\bar{\mathbf{y}}^-$ denotes a violator for the given pair $(\mathbf{x}, \mathbf{y}^+)$; $1 + f_{\mathbf{x}}(\bar{\mathbf{y}}^-) > f_{\mathbf{x}}(\mathbf{y}^+)$ is used to penalize the negative samples so that the margin between the positive and negative samples is larger than 1; $\mathcal{L}(\cdot) : \mathbb{Z}^+ \to \mathbb{R}^+$ is the mapping function that transforms the rank (denotes the order relation) into a loss:

$$\mathcal{L}(k) = \sum_{i=1}^{k} \alpha_i, \alpha_1 > \alpha_2 \cdots \geq 0, \tag{5}$$

where α_i is set to $1/i$, which has shown the good MAP and precision-at-k performance in image retrieval [3] and image annotation [10]. More details about the WARP loss can be found in [10].

Similarly, $\mathbf{X}_{\mathbf{y}}^+$ and $\mathbf{X}_{\mathbf{y}}^-$ represent the relevant and irrelevant image sets of the text query $\mathbf{y}$, respectively. The pairwise ranking loss for the text query $\mathbf{y}$ is defined as:

$$L(\mathbf{y}, \mathbf{X}; \mathbf{U}, \mathbf{V}) = \sum_{\mathbf{y}\in\mathbf{Y}, \mathbf{x}^+\in\mathbf{X}_{\mathbf{y}}^+} E[\mathcal{L}(\lfloor \frac{s-1}{s_{\mathbf{y}}} \rfloor) \times [g_{\mathbf{y}}(\mathbf{x})]_+, \tag{6}$$

where $g_{\mathbf{y}}(\mathbf{x})$=$1 + f_{\mathbf{y}}(\bar{\mathbf{x}}^-) - f_{\mathbf{y}}(\mathbf{x}^+)$.

Listwise loss constraint: The WARP-based methods have some problems in practice by using the pairwise ranking loss to approximate the total ranking loss. First, some rare occurrence queries may not be sampled in the large training set due to the randomness in the limited iterations. Thus, it is hard to enhance the semantic information of these queries. Second, in each sampling stage, one positive sample only finds 1-nearest inter-neighbor, and then the ranking loss is approximated by the loss of the paired samples. Since query samples have a lot of nearest inter-neighbors in a large dataset, it may need more iterations to converge.

To alleviate these problems, we use a listwise loss to improve the performance of pairwise ranking loss. Generally speaking, the listwise information represents the correlation of intra-class and inter-class. We can use the class information to minimize the variance of the samples within the same semantic class and maximize the separability of the samples from the different classes. In PL-ranking, the bi-directional listwise constraint is defined as:

$$\Psi(\mathbf{U},\mathbf{V}) = \sum_{\mathbf{x}\in\mathbf{X}} \Psi(\mathbf{x},\mathbf{Y};\mathbf{U},\mathbf{V}) + \sum_{\mathbf{y}\in\mathbf{Y}} \Psi(\mathbf{y},\mathbf{X};\mathbf{U},\mathbf{V}), \quad (7)$$

where $\Psi(\mathbf{x},\mathbf{Y};\mathbf{U},\mathbf{V})$ is used to enhance the listwise correlation between the image query $\mathbf{x}$ and a text training set, and it is computed as follows:

$$\Psi(\mathbf{x},\mathbf{Y};\mathbf{U},\mathbf{V}) = \sum_{\mathbf{y}^-\in\mathbf{Y}_{\mathbf{x}}^{\text{inter}}} f_{\mathbf{x}}(\mathbf{y}^-) - \sum_{\mathbf{y}^+\in\mathbf{Y}_{\mathbf{x}}^{\text{intra}}} f_{\mathbf{x}}(\mathbf{y}^+), \quad (8)$$

where $\mathbf{Y}_{\mathbf{x}}^{\text{intra}} \in \mathbf{Y}_{\mathbf{x}}^+$ and $\mathbf{Y}_{\mathbf{x}}^{\text{inter}} \in \mathbf{Y}_{\mathbf{x}}^-$ represent the k_1 nearest intra-neighbors and k_2 nearest inter-neighbors of $\mathbf{x}$, respectively. We calculate the nearest neighbors by k-nearest-neighbor which is performed after mapping features into the common subspace. Since the dimension of this subspace is very low, PL-ranking is still efficient when it handles the high-dimensional features.

Similarly, $\Psi(\mathbf{y},\mathbf{X};\mathbf{U},\mathbf{V})$ enhances the correlation between a text query $\mathbf{y}$ and an image training set and is defined as:

$$\Psi(\mathbf{y},\mathbf{X};\mathbf{U},\mathbf{V}) = \sum_{\mathbf{x}^-\in\mathbf{X}_{\mathbf{y}}^{\text{inter}}} f_{\mathbf{y}}(\mathbf{x}^-) - \sum_{\mathbf{x}^+\in\mathbf{X}_{\mathbf{y}}^{\text{intra}}} f_{\mathbf{y}}(\mathbf{x}^+). \quad (9)$$

In the listwise loss, the usage of k_1 nearest intra-neighbors and k_2 nearest inter-neighbors can overcome the above problems of the WARP-based methods. First, the queries can find more relevant samples and irrelevant samples in each sampling. Some rare occurrence samples can increase the semantic information by building the semantic relation with other sampled data. Second, the iterations are reduced since a sample can find more positive samples and violators in each iteration.

Low-rank constraint: In cross-modal retrieval, the multi-modal data are usually represented as the high-dimensional vectors. Since the intrinsic dimensionality of a semantic space is usually much lower than that of original feature space, we use the low-rank constraint to exploit the low-dimensional subspace structures embedded in data. In PL-ranking, the low-rank constraint is defined as:

$$\Omega(\mathbf{U},\mathbf{V}) = \|\mathbf{U}\|_* + \|\mathbf{V}\|_*, \quad (10)$$

where $\|\mathbf{U}\|_*$ and $\|\mathbf{V}\|_*$ denote the nuclear norm based regularization. This regularization has been successfully applied to various applications such as multi-label learning [11] and cross-modal matching [26].

Based on singular value decomposition, a low-dimensional mapping matrix can be represented as the product of the left/right singular vectors. The left/right singular vectors of the mapping explicitly describe the label/feature relevance vectors, so the low-rank based regularization can effectively exploit label and feature correlations. For example, the linear mapping $\mathbf{U}$ can be characterized as:

$$\mathbf{U}^T = \sum_{i=1}^{r} \mathbf{M}_i(U)\mathbf{\Sigma}_i(U)\mathbf{N}_i^T(U), \quad (11)$$

where $r = \min\{m, c\}$, $\mathbf{M}_i(U) \in \mathbb{R}^{c\times r}$ and $\mathbf{N}_i(U) \in \mathbb{R}^{m\times r}$ are left/right singular vectors of $\mathbf{U}$, $\mathbf{\Sigma}_i(U)$ is the i-th singular value of $\mathbf{U}$ and all the singular values are real and non-negative. Assuming $\mathbf{\Sigma}_1(U) \geq \mathbf{\Sigma}_2(U) \geq \cdots \mathbf{\Sigma}_r(U)$, we can measure the complexity of $\mathbf{U}$ by summing its singular values, *i.e.*, how many singular values of $\mathbf{U}$ or singular vectors $\mathbf{M}_i(U)$ and $\mathbf{N}_i(U)$ we should keep. When we keep the largest r' singular values of $\mathbf{U}$, the rank of $\mathbf{U}$ is equal to r'. In this paper, we adopt the nuclear norm regularization to measure the complexity of $\mathbf{U}$:

$$\|\mathbf{U}\|_* = \sum_{i=1}^{r} \mathbf{\Sigma}_i(U). \quad (12)$$

Moreover, each image sample $\mathbf{x}_i$ can be mapped based on the minimization of nuclear norm:

$$\mathbf{U}^T\mathbf{x}_i = \sum_{j=1}^{r'} \mathbf{\Sigma}_j(U)[\mathbf{N}_j^T(U)\mathbf{x}_i]\mathbf{M}_j(U). \quad (13)$$

We note that the resulting vector is a linear combination of label-relevance vectors: $\mathbf{M}_1(U), \mathbf{M}_2(U), \cdots, \mathbf{M}_{r'}(U)$, so it is relevant to the label space. Therefore, the label relevance can be exploited by optimizing these label-relevance vectors. The feature correlations can be reflected by the mapped feature value $\mathbf{N}_j^T(U)\mathbf{x}_i$ which is determined by r' feature-relevance vectors $\{\mathbf{N}_j(U)\}_{j=1}^{r'}$. By using these label-relevance and feature-relevance vectors, both the feature correlations and label correlations can be exploited.

Similarly, each text sample $\mathbf{y}_i$ can also be mapped as a linear combination. Then, minimizing the pairwise ranking loss, listwise loss and low-rank constraint can learn two sets of label-relevance vectors: $\{\mathbf{M}_j(U)\}_{j=1}^{r'}$ and $\{\mathbf{M}_j(V)\}_{j=1}^{r'}$, and feature-relevance vectors $\{\mathbf{N}_j(U)\}_{j=1}^{r'}$ and $\{\mathbf{N}_j(V)\}_{j=1}^{r'}$. Based on these vectors, mapped samples from the different modalities are close to each other within the same class and far away between different classes.

3.2 Optimization via bFAST-SSGD

It is difficult to optimize the objective function in Eq. (2) directly due to that as a matrix function, the nuclear norm is not differentiable.

The recently proposed FAST-SSGD method [8] combines the low-rank stochastic subgradients with the incremental SVD. FAST-SSGD maintains a low-rank factorization in each iteration, and it makes the objective function more easily convergent. Since only one matrix variable is optimized in [8], we extend FAST-SSGD to the bi-variable FAST-SSGD (bFAST-SSGD). bFAST-SSGD can simultaneously learn two optimal low-rank mappings $\mathbf{U}$ and $\mathbf{V}$. Furthermore, we set $\|\mathbf{U}\|_* = \|\mathbf{V}\|_*$ to enforce the interaction of $\mathbf{U}$ and $\mathbf{V}$, by which the number of iterations will be greatly reduced.

In Algo. 1, we show the optimization procedure for learning $\mathbf{U}$ and $\mathbf{V}$. First, $\mathbf{U}$ and $\mathbf{V}$ are initialized with a normal distribution with mean zero and standard deviation one. Then, given the current $\mathbf{U}^{(t)}$ and $\mathbf{V}^{(t)}$, the optimization is conducted as follows:

(1) We sample a paired samples $(\mathbf{x},\mathbf{y})$ randomly, and conduct the bi-directional sampling to obtain $(\mathbf{x},\mathbf{y}^+,\bar{\mathbf{y}}^-,s_{\mathbf{x}})$ and $(\mathbf{y},\mathbf{x}^+,\bar{\mathbf{x}}^-,s_{\mathbf{y}})$;

(2) We calculate the subgradients $\mathbf{G}_U^{(t)}$ and $\mathbf{G}_V^{(t)}$ for $\mathbf{U}$ and $\mathbf{V}$;

(3) We set the step size $\eta_Z^{(t)}$ (Z denotes U or V) to ensure the convergence in $O(c/r)$ iterations:

$$\eta_Z^{(t)} = \beta\frac{\sqrt{r}\Delta_Z}{\sqrt{c}(G_Z + \gamma\sqrt{c})}, \quad (14)$$

Algorithm 1 PL-ranking

```
Input: $\mathbf{U}^{(t)} \in \mathbb{R}^{m\times c^{(t)}}$, $\mathbf{V}^{(t)} \in \mathbb{R}^{n\times c^{(t)}}$, image data matrix
    $\mathbf{X} \in \mathbb{R}^{m\times M}$, text data matrix $\mathbf{Y} \in \mathbb{R}^{n\times M}$
Output: Optimal $\mathbf{U}$ and $\mathbf{V}$
repeat
    Pick a pair of labeled samples $(\mathbf{x}, \mathbf{y})$ randomly
    Draw $\mathbf{y}^+$ from $\mathbf{Y}_{\mathbf{x}}^+$; $s_{\mathbf{x}} \leftarrow 0$;
    Calculate the nearest intra-neighbors $\mathbf{Y}_{\mathbf{x}}^{\text{intra}} \in \mathbf{Y}_{\mathbf{x}}^+$
    and inter-neighbors $\mathbf{Y}_{\mathbf{x}}^{\text{inter}} \in \mathbf{Y}_{\mathbf{x}}^-$
    Draw $\mathbf{x}^+$ from $\mathbf{X}_{\mathbf{y}}^+$; $s_{\mathbf{y}} \leftarrow 0$;
    Calculate the nearest intra-neighbors $\mathbf{X}_{\mathbf{y}}^{\text{intra}} \in \mathbf{X}_{\mathbf{y}}^+$
    and inter-neighbors $\mathbf{X}_{\mathbf{y}}^{\text{inter}} \in \mathbf{X}_{\mathbf{y}}^-$
    repeat
        Sample $\mathbf{y}^-$ from $\mathbf{Y}_{\mathbf{x}}^-$
    until $1 + f_{\mathbf{x}}(\bar{\mathbf{y}}^-) > f_{\mathbf{x}}(\mathbf{y}^+)$ or $s_{\mathbf{x}} > |\mathbf{Y}_{\mathbf{x}}^-|$
    repeat
        Sample $\mathbf{x}^-$ from $\mathbf{X}_{\mathbf{y}}^-$
    until $1 + f_{\mathbf{y}}(\bar{\mathbf{x}}^-) > f_{\mathbf{y}}(\mathbf{x}^+)$ or $s_{\mathbf{y}} > |\mathbf{X}_{\mathbf{y}}^-|$
    if $1 + f_{\mathbf{x}}(\bar{\mathbf{y}}^-) > f_{\mathbf{x}}(\mathbf{y}^+)$ or $1 + f_{\mathbf{y}}(\bar{\mathbf{x}}^-) > f_{\mathbf{y}}(\mathbf{x}^+)$ then
        bFAST-SSGD:
        1). Generate a probing matrix $\mathbf{P}^{(t)} \in c^{(t)} \times r^{(t)}$
        2). $\mathbf{U}^{(t+1)} \leftarrow$ FAST-SSGD$(\mathbf{U}^{(t)}, \mathbf{P}^{(t)}, \mathbf{G}_U^{(t)}, \eta_U^{(t)})$
        3). $\mathbf{V}^{(t+1)} \leftarrow$ FAST-SSGD$(\mathbf{V}^{(t)}, \mathbf{P}^{(t)}, \mathbf{G}_V^{(t)}, \eta_V^{(t)})$
        4). Ensure $\|\mathbf{U}^{(t+1)}\|_* = \|\mathbf{V}^{(t+1)}\|_*$
        end bFAST-SSGD
    else
        continue
    end if
until validation performance does not improve or max
    iterations exceeded
```

Algorithm 2 FAST-SSGD

```
Input: $\mathbf{Z}^{(t)}$, $\mathbf{P}^{(t)}$, $\mathbf{G}_Z^{(t)}$,  $\eta_Z^{(t)} > 0$
SVD computation: $\mathbf{Z}^{(t)} = \mathbf{M}^{(t)}(Z)\mathbf{\Sigma}^{(t)}(Z)(\mathbf{N}^{(t)}(Z))^T$
$\mathbf{S}^{(t)} \leftarrow \mathbf{G}_Z^{(t)}\mathbf{Y}^{(t)}$
$\hat{\mathbf{M}}^{(t+1)} \leftarrow [\mathbf{M}^{(t)}(Z)\mathbf{\Sigma}^{(t)}(Z) \;\; \mathbf{S}^{(t)}]$
$\hat{\mathbf{N}}^{(t+1)} \leftarrow [\mathbf{N}^{(t)}(Z) \;\; -\eta_Z^{(t)}\mathbf{P}^{(t)}]$
Factorize:$\hat{\mathbf{M}}^{(t+1)} = \mathbf{Q}_M\mathbf{R}_M$
Factorize:$\hat{\mathbf{N}}^{(t+1)} = \mathbf{Q}_N\mathbf{R}_N$
$\mathbf{D} \leftarrow \mathbf{R}_M\mathbf{R}_N^T$
SVD computation: $\mathbf{D} = \mathbf{M}(D)\mathbf{\Sigma}^{(t+1)}(D)\mathbf{N}^T(D)$
$\mathbf{M}^{(t+1)}(Z) \leftarrow \mathbf{Q}_M\mathbf{M}(D)$, $\mathbf{N}^{(t+1)}(Z) \leftarrow \mathbf{Q}_N\mathbf{N}(D)$
$\mathbf{Z}^{(t+1)} \leftarrow \mathbf{M}^{(t+1)}(Z)\mathbf{\Sigma}^{(t+1)}(D)\mathbf{N}^{(t+1)}(Z)$
return $\mathbf{Z}^{(t+1)}$
```

where $\beta > 0$, Δ_Z and G_Z denote the upper bounds of $\|\mathbf{Z}\|_F$ and $\|\mathbf{G}_Z^{(t)}\|_F$.

(4) $\mathbf{U}^{(t+1)}$ and $\mathbf{V}^{(t+1)}$ are obtained via Algo. 2.

In Algo. 1, $\mathbf{P} \in \mathbb{R}^{c\times r}$ is a probing matrix and $E[\mathbf{P}\mathbf{P}^T] = \mathbf{I}_{c\times c}$, where $r \leq c$ is the adjustable parameter, and $\mathbf{I}_{c\times c}$ is the identity matrix. By the linearity of the expectation, for any subgradient $\mathbf{G}^{(t)}$, we have $E[\mathbf{G}^{(t)}\mathbf{P}\mathbf{P}^T] = \mathbf{G}^{(t)}E[\mathbf{P}\mathbf{P}^T] = \mathbf{G}^{(t)}$. Thus, $\mathbf{G}^{(t)}\mathbf{P}\mathbf{P}^T$ can be used as an unbiased estimator of $\mathbf{G}^{(t)}$. Since $\mathbf{G}^{(t)}\mathbf{P}\mathbf{P}^T$ has the rank at most r, this estimator is a low-rank unbiased estimator, by which $\mathbf{U}$ and $\mathbf{V}$ are guaranteed as the low-rank mappings in each iteration.

To effectively learn the label-relevance and feature-relevance vectors, we set $\|\mathbf{U}\|_* = \|\mathbf{V}\|_*$ after each iteration, which we briefly reason below. From Eq. (13), we conclude:

$$\begin{aligned} \mathbf{U}^T\mathbf{x}_i &= \sum_{j=1}^{r'} \mathbf{\Sigma}_j(U)[\mathbf{N}_j^T(U)\mathbf{x}_i]\mathbf{M}_j(U) \\ \mathbf{V}^T\mathbf{y}_i &= \sum_{j=1}^{r'} \mathbf{\Sigma}_j(V)[\mathbf{N}_j^T(V)\mathbf{y}_i]\mathbf{M}_j(V) \end{aligned} . \tag{15}$$

It is obvious that the singular values are the coefficients of the above linear combinations. We hope to learn the optimal singular vectors $\mathbf{M}(U)$, $\mathbf{N}(U)$, $\mathbf{M}(V)$ and $\mathbf{N}(V)$ such that the correlations between two modalities can be exploited. If $\sum_{j=1}^{r'} \mathbf{\Sigma}_j(U)$ is equal to $\sum_{j=1}^{r'} \mathbf{\Sigma}_j(V)$, we can truly optimize these singular vectors, otherwise the correlations may be resulted from these singular values rather than the optimal label-relevance and feature-relevance vectors. Besides, $\|\mathbf{U}\|_* = \|\mathbf{V}\|_*$ can construct certain relations between two mappings to some extent.

Thus, in bFAST-SSGD, $\mathbf{U}$ and $\mathbf{V}$ are forced to be multiplied by a constant respectively to ensure $\|\mathbf{U}\|_* = \|\mathbf{V}\|_*$ after each subgradient descent. Suppose after the t-th iteration, we obtain $\mathbf{U}^{(t)} = \mathbf{M}^{(t)}(U)\mathbf{\Sigma}^{(t)}(U)\mathbf{N}^{(t)}(U)$, $\mathbf{V}^{(t)} = \mathbf{M}^{(t)}(V)\mathbf{\Sigma}^{(t)}(V)\mathbf{N}^{(t)}(V)$. Then the update is defined as:

$$\begin{aligned} \mathbf{U}^{(t)} &= \alpha(\mathbf{U}^{(t)}./\mathrm{Tr}(\mathbf{\Sigma}^{(t)}(U))) \\ \mathbf{V}^{(t)} &= \alpha(\mathbf{V}^{(t)}./\mathrm{Tr}(\mathbf{\Sigma}^{(t)}(V))) \end{aligned} , \tag{16}$$

where $\alpha = \sqrt{\mathrm{Tr}(\mathbf{\Sigma}^{(t)}(U)) \times \mathrm{Tr}(\mathbf{\Sigma}^{(t)}(V))}$.

The computational complexity of PL-ranking scales as $O(d(c^{(t)}+r^{(t)})^2)$ for each iteration [8], where $d = \max(m, n)$. Since $c^{(t)}$ is far smaller than m and n, PL-ranking results in large computational savings on learning $\mathbf{U}$ and $\mathbf{V}$. Generally, the current metric learning approaches usually scale as $O(d^2)$ or $O(d^3)$. The complete derivation of PL-ranking is given in the supplementary material.

3.3 Convergence analysis

In this section, we present the convergence analysis of the proposed framework. We prove that Algo. 1 will monotonically decrease the objective function in Eq. (2).

According to the definition of the objective function, the pairwise ranking loss has the following equation:

$$\begin{aligned} &L(\mathbf{U}^{(t+1)}, \mathbf{V}^{(t+1)}) - L(\mathbf{U}^{(t)}, \mathbf{V}^{(t)}) = \\ &\{L(\mathbf{U}^{(t+1)}, \mathbf{V}^{(t+1)}) - L(\mathbf{U}^{(t)}, \mathbf{V}^{(t+1)})\} + . \\ &\{L(\mathbf{U}^{(t)}, \mathbf{V}^{(t+1)}) - L(\mathbf{U}^{(t)}, \mathbf{V}^{(t)})\} \end{aligned} \tag{17}$$

The two terms at the right side of Eq. (17) are less than or equal to zero. Taking one sampling as example, we have:

$$\begin{aligned} &L(\mathbf{x}, \mathbf{Y}, \mathbf{U}^{(t+1)}, \mathbf{V}^{(t+1)}) - L(\mathbf{x}, \mathbf{Y}, \mathbf{U}^{(t)}, \mathbf{V}^{(t+1)}) \\ &= -\mathbf{x}^T\mathbf{G}^{(t)}(U)\mathbf{V}^{(t+1)^T}(\bar{\mathbf{y}}^- - \mathbf{y}^+) \leq 0 \end{aligned} , \tag{18}$$

since $\mathbf{x}^T\mathbf{G}_U^{(t)}\mathbf{V}^{(t+1)^T}\bar{\mathbf{y}}^-$ is greater than $\mathbf{x}^T\mathbf{G}_U^{(t)}\mathbf{V}^{(t+1)^T}\mathbf{y}^+$ before convergence, the above inequality is satisfied.

Similarly, both the listwise loss and the low-rank constraint also satisfy this property. Therefore, the overall objective function will monotonically decrease after each iteration and can converge to a global optimum eventually.

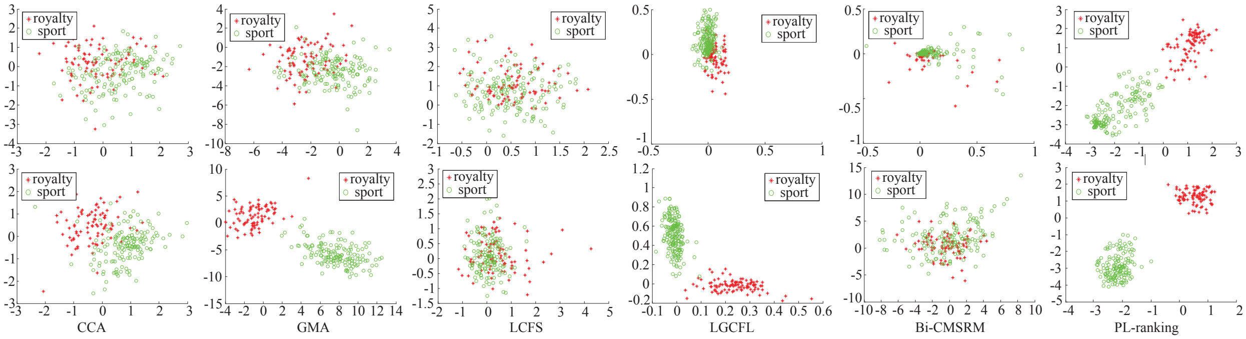


Figure 2: Low-dimensional mapping of images and texts from 'royalty' and 'sport' classes of Wiki. The top row shows the mapping for image modality, and the bottom shows the mapping for text modality.

4. EXPERIMENTS

In this section, we present the experimental results of PL-ranking and the other methods for image-text retrieval, *i.e.*, image-query-texts and text-query-images. We evaluate and compare different methods on four public datasets-Wiki [20], Flickr [2], Pascal VOC 2007 [4] and NUS-WIDE [2]. There are many distinct properties of these datasets. The text modality of the four datasets are article, sentences and tags, respectively. The size of these datasets ranges from 2k to 26k and the number of the classes ranges from 10 to 20.

4.1 Experiment settings

The proposed PL-ranking is compared with several related methods, such as CCA [9], GMA [21], LCFS [26], Bi-CMSRM [27] and LGCFL [13]. Both CCA and PL-ranking focus on learning a latent space. Comparing with CCA aims to test PL-ranking's ability to learn a useful latent space. PL-ranking is a supervised learning method like GMA. The comparison between PL-ranking and GMA can validate the effectiveness of PL-ranking on exploiting the label information. LCFS also uses the nuclear norm as the low-rank constraint. This comparison can test PL-ranking's ability to learn the low-rank mappings. We further highlight the advantages of low-rank optimization based ranking method by comparing PL-ranking with Bi-CMSRM. LGCFL uses the valuable class information to learn the consistent features and achieves the state-of-the-art performance. Additionally, since the listwise loss and nuclear norm constraints are used to improve the performance of the pairwise ranking in PL-ranking, we simplify PL-ranking as PL-ranking$_p$ to validate their impacts on the performance. In other words, PL-ranking$_p$ only uses the pairwise ranking loss constraint.

For the evaluation, we use mean average precision (MAP) [20] as the performance measure. MAP@R measures MAP at fixed number of the retrieved samples, and we set $R = 10$ for the top 10 retrieved samples and $R = all$ for all the samples. Besides, we also display the precision-recall curve [20] for all methods to pictorially demonstrate performance.

Since CCA and GMA just focus on the common subspace learning, principal component analysis (PCA) is performed on the original features before learning the latent space. By this way, the redundant information of the original features is greatly reduced. For the compared methods, we use the parameters' optimal settings tuned by a parameter validation process except for the specified values. PL-ranking uses the following parameter setting: $\lambda = 0.001$, $\gamma = 0.1$, $\beta = 0.01$, $k_1 = 20$, $k_2 = 200$, in all experiments. Specially, we set r to c to avoid an additional tunable parameter. For fair comparison, we conduct experiments 10 times by randomly selecting training/validating/testing combinations, and compare the average performance for all methods.

4.2 Results on mappings

In this section, we show the data distribution after the low-dimensional mapping. To demonstrate this, we construct a toy dataset using the 'royalty' and 'sport' classes of the Wiki dataset. The texts are represented as 5,000-dimensional Bag-of-Words (BoW) features, and images use the 1,000-dimensional Bag-of-Visual-Words (BoVW) features. For both modalities, the first and second most correlated components of the different methods are shown in Fig. 2. The red color represents the data distribution of the 'royalty' class, and green color represents the 'sport' class. From this figure, we can see that PL-ranking can unit the same-class samples and separates the different classes for both directions of retrieval, but the second best results (GMA and LGCFL) only unit the same-class samples and separate the different classes for text query. Furthermore, both image and text distributions of PL-ranking are in the same coordinate range, while the other methods' coordinate ranges are quite different. These results validate that PL-ranking is able to map the different features into a discriminative subspace such that the low-dimensional representations are enforced with the high correlation.

4.3 Results on Wiki

The Wiki dataset [20] is generated from Wikipedia's "featured articles". It consists of 2,866 image-text pairs with 10 semantic categories. For the texts, we extract the 5,000-dimensional features using the BoW representations with the TF-IDF weighting scheme. For the images, we use the 1,000-dimensional BoVW features. We randomly choose 1,500 pairs of the data for training, 500 pairs for validating and 866 pairs for testing. In Table 1, we report the MAP scores for image query, text query and their average. From this table, we draw the following conclusions:

First, compared to PL-ranking$_p$, PL-ranking gains significant improvements. For example, under R=*all*, the MAP scores of PL-ranking$_p$ are 0.1715, 0.2042 and 0.1879 for text query, image query and their average, respectively, while PL-ranking achieves 0.2221, 0.2625 and 0.2423, respectively. The improvements are 29.50%, 28.55% and 28.95%, respec-

Table 1: The performance comparison in terms of MAP@R on Wiki dataset.

Tasks Methods	$R = 10$			$R = all$		
	Text query	Image query	Average	Text query	Image query	Average
CCA	0.3081	0.2425	0.2753	0.1415	0.1595	0.1505
GMA	0.3805	0.2284	0.3045	0.1726	0.2305	0.2016
LCFS	0.4632	0.2637	0.3635	0.1917	0.2481	0.2199
LGCFL	0.4472	0.2849	0.3661	0.2156	0.2615	0.2386
Bi-CMSRM	0.4599	0.2732	0.3666	0.2123	0.2528	0.2326
PL-ranking$_p$	0.3840	0.2487	0.3164	0.1715	0.2042	0.1879
PL-ranking	**0.4823**	**0.2882**	**0.3853**	**0.2221**	**0.2625**	**0.2423**

Table 2: The performance comparison in terms of MAP@R on Flickr dataset.

Tasks Methods	$R = 10$			$R = all$		
	Text query	Image query	Average	Text query	Image query	Average
CCA	0.3106	0.2000	0.2553	0.1438	0.1455	0.1447
GMA	0.4437	0.2951	0.3694	0.1884	0.2657	0.2271
LCFS	0.4797	0.2600	0.3699	0.1868	0.2103	0.1986
LGCFL	0.4734	0.3028	0.3881	0.2234	0.2767	0.2501
Bi-CMSRM	0.4209	0.2774	0.3492	0.2040	0.2791	0.2416
PL-ranking$_p$	0.4255	0.3101	0.3678	0.1856	0.2156	0.2006
PL-ranking	**0.5376**	**0.3098**	**0.4237**	**0.2323**	**0.2851**	**0.2587**

Table 3: The performance comparison in terms of MAP@all on Pascal and NUS-WIDE datasets.

Methods Tasks		CCA	GMA	LCFS	LGCFL	Bi-CMSRM	PL-ranking
Pascal	Image query	0.163	0.308	0.344	0.401	0.292	**0.418**
	Text query	0.150	0.244	0.267	0.320	0.204	**0.338**
	Average	0.157	0.276	0.306	0.361	0.248	**0.378**
NUS-WIDE	Image query	0.248	0.374	0.413	0.424	0.338	**0.443**
	Text query	0.201	0.241	0.274	0.307	0.217	**0.323**
	Average	0.225	0.308	0.344	0.366	0.278	**0.383**

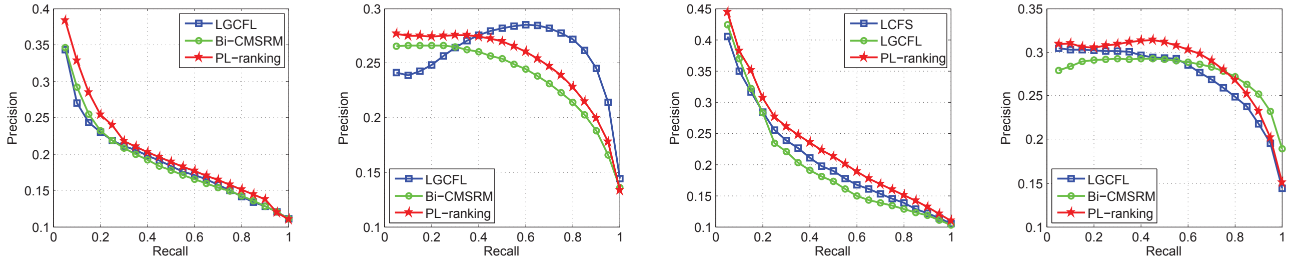
tively. Since PL-ranking adds the listwise loss and nuclear norm constraints on PL-ranking$_p$, the superior performance of PL-ranking indicates that the combination of the listwise loss and nuclear norm can improve the performance of the pairwise ranking. Besides, we also do experiments to validate that the nuclear norm is superior to F-norm. The MAP scores of image query and text query are 0.2198 and 0.1929 in R=all, and 0.2316 and 0.4237 in R=10 by adopting F-norm. This is mainly because F-norm leads to high-rank gradient such that the mappings' structures are destroyed. Thus, its generalization performance cannot be ensured.

Second, the MAP scores of PL-ranking outperform those of Bi-CMSRM on both retrieval tasks. The MAP scores of Bi-CMSRM are 0.2123 and 0.2528 under R=all. Though Bi-CMSRM and PL-ranking are both the ranking-based methods, Bi-CMSRM only optimizes the listwise ranking loss by learning the structural large margin. But PL-ranking integrates the pairwise ranking loss, the listwise loss and nuclear norm constraints. The advantage of PL-ranking demonstrates that it is reasonable to use the pairwise ranking loss and low-rank constraints for ranking.

Third, the performance of PL-ranking is also better than that of LCFS. The MAP scores of LCFS are 0.1917 and 0.2481 for text query and image query when under R=all. Although the nuclear norm is also adopted to optimize the low-rank mappings, LCFS mainly focuses on preserving the correlations between the paired samples. In PL-ranking, the pairwise ranking loss and listwise loss are also adopted except for the nuclear norm. The improvement of PL-ranking indicates that PL-ranking can improve the precision at the top of ranked list by using the pairwise ranking loss, and the listwise loss can preserve the class information.

Finally, the improvement of PL-ranking under R=10 is as significant as that under R=all,. For example, compared with Bi-CMSRM, the average MAP of PL-ranking improves about 4.17% under R=all and 5.1% under R=10. These results show that PL-ranking optimizes the top of ranking, which has the practical significance since the relevant samples are pushed at the top of ranked list.

Fig. 3(a) and 3(b) show the precision-recall curves of the different methods on the Wiki dataset. We select 3 maximum curves to make the clear presentation. For two retrieval tasks, PL-ranking achieves the higher precision than the other methods at the low levels of recall, which demonstrate that the top of ranking is optimized in PL-ranking. We also see that LGCFL achieves lower precision at low levels of recall in image query. LGCFL learns coupled mappings by optimizing the labeling approximation error between the given data and labels. Since the class labels apply more directly to texts than images, the image query is more likely to mismatch. As a result, the relevant samples may be pushed in the front of the ranked list but not the top of ranked list.



(a) Wiki Text Query (b) Wiki Image Query (c) Flickr Text Query (d) Flickr Image Query

Figure 3: Precision-Recall curves on Wiki and Flickr datasets.

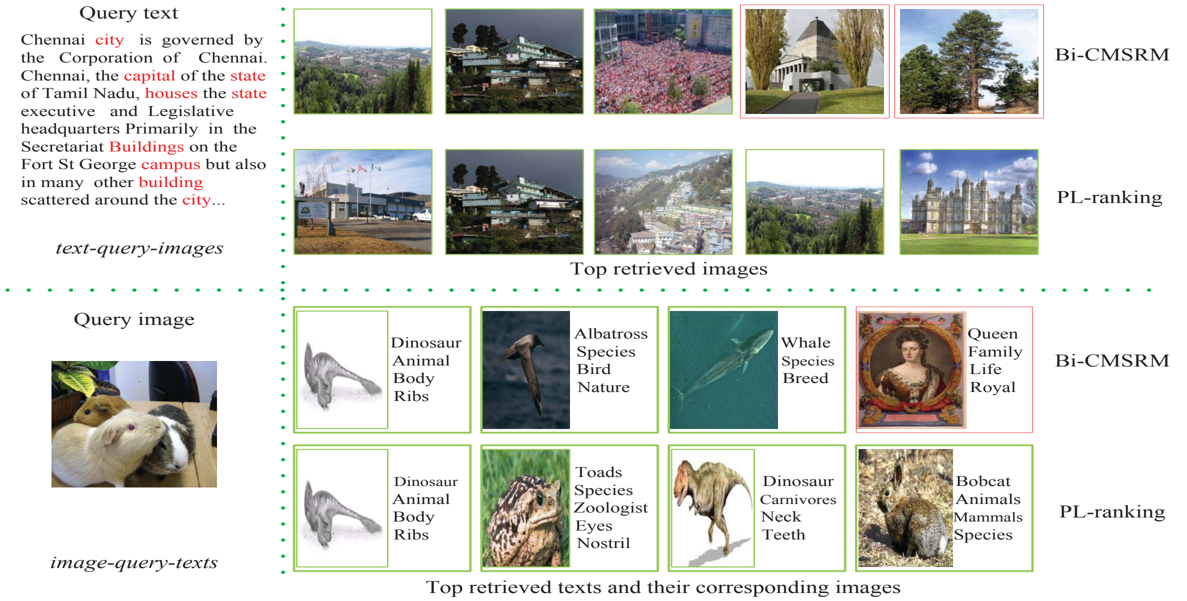

Query text
Chennai city is governed by the Corporation of Chennai. Chennai, the capital of the state of Tamil Nadu, houses the state executive and Legislative headquarters Primarily in the Secretariat Buildings on the Fort St George campus but also in many other building scattered around the city...
text-query-images
Bi-CMSRM
PL-ranking
Top retrieved images
Query image
image-query-texts
Dinosaur Animal Body Ribs
Albatross Species Bird Nature
Whale Species Breed
Queen Family Life Royal
Bi-CMSRM
Dinosaur Animal Body Ribs
Toads Species Zoologist Eyes Nostril
Dinosaur Carnivores Neck Teeth
Bobcat Animals Mammals Species
PL-ranking
Top retrieved texts and their corresponding images

Figure 4: Two examples of text-query-images and image-query-texts on Wiki dataset. For each example, we show the query and its corresponding top retrieved samples by PL-ranking and Bi-CMSRM. In top half figure, we give a text about "city", "building" and etc; and a "Guinea pigs" image belonging to the "biology" class is given in bottom half figure. The incorrect retrieved samples are shown in the red frame.

We demonstrate two examples of retrieved results on two directional retrieval in Fig. 4. In this paper, the retrieved sample is relevant to the query if they belong to the same semantic class. Based on the intuitive judgement, we can observe that the top retrieved samples of PL-ranking are clearly relevant while Bi-CMSRM produces some irrelevant samples for both examples. It is encouraging that the proposed method retrieves more relevant results comparing with Bi-CMSRM.

4.4 Results on Flickr

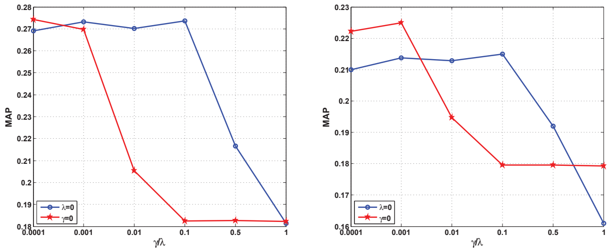
The Flickr dataset is a subset selected from NUS [2]. It consists of 5,730 paired samples that belong to the 10 largest classes (concepts) with each pair having a unique class label (concept). Images are represented by 500-dimensional BoVW based on the SIFT descriptors, and texts are represented by the index vectors of the most frequent 1,000 tags. We randomly choose 75% of the data for training, 10% for validating, and the remaining 15% for testing.

The MAP scores and precision-recall curves of the different methods on the Flickr dataset are shown in Table 2 as well as Fig. 3(c) and 3(d). From Table 2, we conclude that PL-ranking achieves the average MAP scores of 0.2587 and 0.4237 under $R = all$ and $R = 10$ respectively, which are about 3.4% and 9.2 % higher than the second best results (0.2501 and 0.3881 for LGCFL). These results validate that PL-ranking can achieve the best results on Flickr dataset, and the analysis on the Wiki dataset is reasonable.

4.5 Results on Pascal and NUS-WIDE

To validate the generalization performance of PL-ranking, we also do experiments on Pascal [4] and NUS-WIDE [2].

The Pascal dataset consists of 5,011/4,952(testing/training) images-tag pairs, which belongs to 20 semantic classes. We only select the single-labeled data, which results in 2,808 training and 2,841 texting data. We use the publicly available 512-dimensional GIST features for images. For texts, we use the 399-dimensional word frequency features. The NUS-WIDE dataset consists of 15,628/10,437 (training/testing) image-tag pairs. This dataset is pruned from the original



(a) Image Query (b) Text Query

Figure 5: Sensitivity test of λ and γ on Flickr dataset. For the curve of $\lambda = 0$, x-axis means that γ varies from 0.0001 to 1. x-axis means that λ varies from 0.0001 to 1 for the curve of $\gamma = 0$.

train-test split of the NUS dataset by keeping the pairs which belongs to the 10 largest classes. Each image-tag pair belongs to a unique class. Each text is represented by a 1000 dimensional word frequency vector based tag features while each image is represented as a 500 dimensional SIFT feature.

The MAP scores on the two datasets are reported in Table 3. The average MAP scores of PL-ranking on Pascal and NUS-WIDE are 0.378 and 0.383, which are about 4.7% and 4.8% higher than the second best results (0.361 and 0.366 for LGCFL). It is to note that image features of Pascal is different from other datasets, and the sample size of NUS-WIDE is greater than other datasets. The superior results show that PL-ranking can effectively handle the various features and the large-scale dataset.

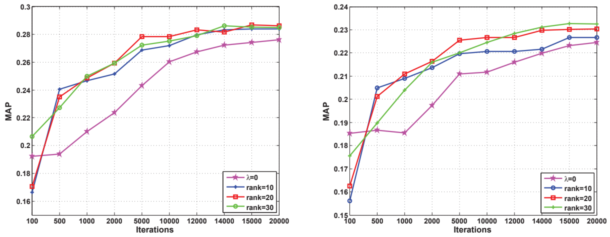
4.6 Parameter sensitivity analysis

There are mainly three parameters in PL-ranking: λ, γ and c. On the Flickr dataset, we conduct the empirical analysis on the parameter sensitivity to show how to select the values for them.

First, Fig. 5 shows the MAP scores on both retrieval tasks in order to explore the contribution of each constraint. $\lambda = 0$ means that the listwise loss constraint is removed from PL-ranking. In this case, we observe that the performance is stable when γ varies from 10^{-4} to 10^{-1} on both directional retrieval, and both maximum values are achieved at 10^{-1}. This figure also shows the MAP scores by varying λ when γ is fixed with 0, which means that the low-rank constraint is removed. From the figure, we can know that PL-ranking obtains the highest MAP score at 10^{-3}. Thus, λ is set to 10^{-3}, and γ is set to 10^{-1} in all experiments.

Then, we repeat the experiments with the different iterations. The MAP scores of PL-ranking under the different iterations are shown in Fig. 6. From the figure, we know that for different ranks, MAP scores are improved with the increase of the iterations when the number of iterations is smaller than 15,000. But they are near to the constant after about 15,000 iterations. This experiment validates that PL-ranking can achieve the stable performance on the different ranks. In our experiment, we find 1.189 seconds are needed for each iteration in MATLAB R2013a, and our hardware configuration comprises a 3.6-GHz CPU and a 16GB RAM.

In Fig. 6, we also show the MAP scores of PL-ranking when λ is set to 0, which means that the listwise loss is removed from PL-ranking. From the figure, we observe that the MAP scores are still increasing after 15,000 iterations. But for PL-ranking, the MAP scores are near to the constant after 15,000 iterations. The iterations of PL-ranking are greatly reduced by using the listwise loss. This phenomenon validates that the listwise loss can reduce the iterations via preserving the class information in each iteration.



(a) Image Query (b) Text Query

Figure 6: The MAP scores of PL-ranking with the different iterations on Flickr dataset.

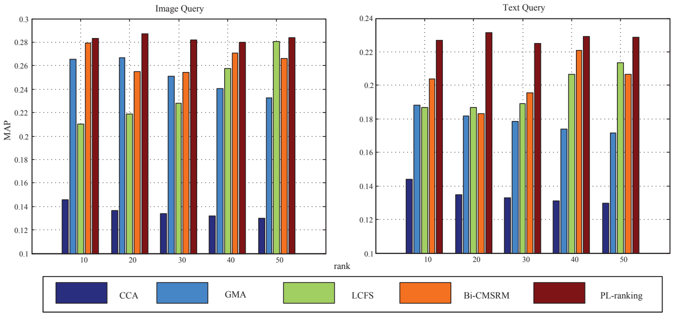


Figure 7: Performance of all methods with different ranks on Flickr dataset.

Furthermore, we show the MAP scores of the different methods on two retrieval tasks with varying ranks (*i.e.*, the dimensionality of the latent space) in Fig. 7. From the figure, we know that compared with other methods, the performance of PL-ranking is stable with the variation of ranks. These results show that the dimension of the latent space has little influence on the performance of PL-ranking.

Finally, we report the MAP scores without the bi-directional constraint. When we only perform the image-query-texts retrieval, the MAP scores of image query and text query are 0.2041 and 0.1694 on Wiki, and 0.2117 and 0.1904 on Flickr. In the other directional retrieval, the MAP scores of image query and text query are 0.1829 and 0.1583 on Wiki, and 0.2186 and 0.1852 on Flickr. These results demonstrate that putting the optimization of two directions of retrieval together can further improve the retrieval performance.

5. CONCLUSION

This paper proposes a novel ranking method named PL-ranking for cross-modal retrieval based on the low-rank optimization framework. In PL-ranking, the pairwise constraint ensures that the relevant samples appear at the top of ranked list. The listwise constraint enhances the separability among the different classes by minimizing the nearest intra-neighbors variance and maximizing the nearest inter-neighbors separability for each sample. The low-rank constraint reveals the latent semantic information. Experimental results on four public cross-modal datasets have shown that the proposed method improves at least 4.7% than that of the state-of-the-art method. Our future work will focus on making our method more efficient and online, so that it can be applied to deal with more practical problems.

6. ACKNOWLEDGMENTS

This work was supported in part by National Basic Research Program of China (973 Program): 2012CB316400 and 2015CB351802, Natural Science Foundation of China (NSFC): 61332016, 61303153, 61572465 and 61429201. This work was supported in part to Dr. Qi Tian by ARO grants W911NF-15-1-0290 and Faculty Research Gift Awards by NEC Laboratories of America and Blippar.

7. REFERENCES

[1] B. Bai, J. Weston, D. Grangier, R. Collobert, K. Sadamsda, Y. Qi, O. Chapelle, and K. Weinberger. Learning to rank with (a lot of) word features. *Information Retrieval*, 13(3):291–314, 2010.

[2] T. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y. Zheng. Nus-wide: A real-world web image database from national university of singapore. In *ACM International Conference on Image and Video Retrieval*, 2009.

[3] L. Daryl and L. Gert. Efficient learning of mahalanobis metrics for ranking. In *International Conference on Machine Learning*, 2014.

[4] M. Everingham, V. Gool, C. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 2010.

[5] Y. Gong, Q. Ke, M. Isard, and S. Lazebnik. A multi-view embedding space for modeling internet images, tags, and their semantics. *International Journal of Computer Vision*, 106(2):210–233, 2014.

[6] D. Grangier and S. Bengio. A discriminative kernel-based approach to rank images from text queries. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(8):1371–1384, 2008.

[7] A. Habibian, T. Mensink, and C. Snoek. Videostory: A new multimedia embedding for few-example recognition and translation of events. In *ACM International Conference on MultiMedia*, 2014.

[8] A. Haim, K. Satyen, P. Shiva, and S. Vikas. Efficient and practical stochastic subgradient descent for nuclear norm regularization. In *International Conference on Machine Learning*, 2012.

[9] D. Hardoon, S. Szedmark, and J. Shawe-Taylor. Canonical correlation analysis: An overview with application to learning methods. *Neural Computation*, 16(12):2639–2664, 2004.

[10] W. Jason, B. Samy, and U. Nicolas. Large scale image annotation: learning to rank with joint word-image embeddings. *Machine Learning*, 81:21–35, 2010.

[11] L. Jing, L. Yang, and J. Yu. Semi-supervised low-rank mapping learning for multi-label classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[12] M. Kan, S. Shan, H. Zhang, S. Lao, and X. Chen. Multi-view discriminant analysis. In *European Conference on Computer Vision*, 2012.

[13] C. Kang, S. Xiang, S. Liao, C. Xu, and C. Pan. Learning consistent feature representation for cross-modal multimedia retrieval. *IEEE Transactions on Multimedia*, 17(3):370–381, 2015.

[14] V. Mahadevan, C. Wong, J. Pereira, T. Liu, N. Vasconcelos, and L. Saul. Maximum covariance unfolding: Manifold learning for bimodal data. In *Advances in Neural Information Processing Systems*, 2011.

[15] X. Mao, B. Lin, D. Cai, X. He, and J. Pei. Parallel field alignment for cross media retrieval. In *ACM International Conference on MultiMedia*, 2013.

[16] B. McFee and G. Lanckriet. Metric learning to rank. In *International Conference on Machine Learning*, 2010.

[17] J. Pereira, E. Coviello, G. Doyle, N. Rasiwasia, G. Lanckriet, R. Levy, and N. Vasconcelos. On the role of correlation and abstraction in cross-modal multimedia retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(3):521–535, 2014.

[18] V. Ranjan, N. Rasiwasia, and C. Jawahar. Multi-label cross-modal retrieval. In *IEEE International Conference on Computer Vision*, 2015.

[19] N. Rasiwasia, D. Mahajan, V. Mahadevan, and G. Aggarwal. Cluster canonical correlation analysis. In *International Conference on Artificial Intelligence and Statistics*, 2014.

[20] N. Rasiwasia, J. Pereira, E. Coviello, G. Doyle, G. Lanckriet, R. Levy, and N. Vasconcelos. A new approach to crossmodal multimedia retrieval. In *ACM International Conference on MultiMedia*, 2010.

[21] A. Sharma, A. Kumar, D. Hal, and D. Jacobs. Generalized multiview analysis: A discriminative latent space. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[22] G. Song, S. Wang, Q. Huang, and Q. Tian. Similarity gaussian process latent variable model for multi-modal data analysis. In *IEEE International Conference on Computer Vision*, 2015.

[23] N. Usunier, D. Buffoni, and P. Gallinari. Ranking with ordered weighted pairwise classification. In *International Conference on Machine Learning*, 2009.

[24] Y. Verma and C. Jawahar. Im2text and text2im: Associating images and texts for cross-modal retrieval. In *British Machine Vision Conference*, 2014.

[25] C. Wang and S. Mahadevan. Heterogeneous domain adaptation using manifold alignment. In *International Joint Conference on Artificial Intelligence*, 2011.

[26] K. Wang, R. He, W. Wang, L. Wang, and T. Tan. Learning coupled feature spaces for cross-modal matching. In *IEEE International Conference on Computer Vision*, 2013.

[27] F. Wu, X. Lu, Z. Zhang, S. Yan, Y. Rui, and Y. Zhuang. Cross-media semantic representation via bi-directional learning to rank. In *ACM International Conference on MultiMedia*, 2013.

[28] Y. Yang, D. Xu, F. Nie, J. Luo, and Y. Zhuang. Ranking with local regression and global alignment for cross media retrieval. In *ACM International Conference on MultiMedia*, 2009.