A Bregman Divergence Optimization Framework for Ranking on Data Manifold and Its New Extensions

Bin Xu, Jiajun Bu, Chun Chen
Zhejiang Provincial Key Laboratory of Service
Robot, College of Computer Science
Zhejiang University, Hangzhou, China
xbzju,bjj,chenc@zju.edu.cn

Deng Cai
State Key Lab of CAD&CG
College of Computer Science
Zhejiang University, Hangzhou, China
dengcai@cad.zju.edu.cn

Abstract

Recently, graph-based ranking algorithms have received considerable interests in machine learning, computer vision and information retrieval communities. Ranking on data manifold (or manifold ranking, MR) is one of the representative approaches. One of the limitations of manifold ranking is its high computational complexity ($O(n^3)$, where n is the number of samples in database). In this paper, we cast the manifold ranking into a Bregman divergence optimization framework under which we transform the original MR to an equivalent optimal kernel matrix learning problem. With this new formulation, two effective and efficient extensions are proposed to enhance the ranking performance. Extensive experimental results on two real world image databases show the effectiveness of the proposed approach.

Introduction

Graph-based ranking algorithms have received considerable interests in machine learning, computer vision and information retrieval communities recently. Ranking on data manifold (or manifold ranking, MR) (Zhou et al. 2004a; 2004b) is one of the representative approaches and has been widely applied in various information retrieval and machine learning applications.

The core idea of MR is to rank the data with respect to the intrinsic geometric structure collectively revealed by a large amount of (unlabeled) data. By taking both the labeled (the query) and unlabeled data (the database) into account, MR assigns each data point a relative ranking score which can be regarded as the relevance degree to the query. Unlike the pairwise similarities or distances used in many traditional methods, the ranking score is more meaningful to measure the semantic relevance expressed within the underlying geometric structure of the data set. A number of works have shown that MR has excellent performance and feasibility on a variety of data types, such as image (He et al. 2004), text (Wan, Yang, and Xiao 2007), and vedio (Yuan et al. 2006). Moreover, it has been shown that for the task of ranking data on a connected and undirected graph without queries, MR yields the same ranking list with the famous PageRank (Brin and Page 1998) algorithm (Zhou et al. 2004b).

One of the main drawbacks of manifold ranking is its high computational complexity. Given a query, MR constructs an affinity graph and propagate the ranking scores on the graph which leads to a $O(n^3)$ complexity, where n is the number of samples in the database. If the query is already in the database, MR can use off-line pre-computation to reduce the on-line cost. However, for a query out of the database, the expensive ranking score propagation step needs to be performed in the on-line stage which is usually referred as the out-of-sample problem (Bengio et al. 2004). Recently, many MR extensions have been proposed to improve the algorithm in different aspects. For example, He *et al*. (He et al. 2006) proposed to use the nearest neighbors of the query as the input to avoid the out-of-sample problem. Xu *et al*. (Xu et al. 2011) introduced a low-rank adjacency matrix approximation and leverage its properties to reduce the computational cost. Cheng *et al*. (Cheng et al. 2011) tried to increase the ranking diversity by turning ranked objects into sink points to prevent redundant objects receiving a high rank.

In this paper, we cast the manifold ranking algorithm into a Bregman divergence optimization framework under which we obtain a new understanding and viewpoint of the algorithm. That is, the optimal ranking function can be built by learning an optimal kernel matrix under the Bregman matrix divergence metric. With this new formulation, we propose two efficient and effective extensions to improve the ranking performance. Meanwhile, our extensions have a closed form solution and is friendly to the out-of-sample data. We applied our method to the content-based image retrieval (CBIR) application. Extensive experiment results on two real world image databases demonstrate the effectiveness of our proposed algorithms.

The main contributions of this paper include: (1) We are the first to formulate the manifold ranking algorithm as a Bregman divergence optimization problem; (2) With which, we get a new understanding of MR's object – to learn an optimal kernel matrix; (3) With this new formulation, we propose two efficient and effective extensions, named DMR$_E$ and DMR$_C$, to improve the performance of traditional MR. (4) In the second extension DMR$_C$, we are able to utilize the information of pairwise constraints inducted from user feedbacks to guide the ranking, which is a promising way for semi-supervised ranking algorithms.

Preliminaries

In this section, we briefly review some preliminary knowledge which are highly related to our work.

Manifold Ranking

Given a data set $X = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n] \in \mathbb{R}^{m \times n}$ where each column is a sample vector. The manifold ranking (MR) first builds an affinity graph on the data (*e.g.*, kNN graph). Let $W \in R^{n \times n}$ denote the weight matrix of the graph with w_{ij} saving the weight of each edge. A common way to compute the weight is using the heat kernel $w_{ij} = \exp[-d^2(\mathbf{x}_i, \mathbf{x}_j)/2\sigma^2)]$ if there is an edge linking $\mathbf{x}_i$ and $\mathbf{x}_j$, otherwise $w_{ij} = 0$. Function d is a distance metric, such as the Euclidean distance.

Let $\mathbf{f}$ be a ranking function which assigns to each point $\mathbf{x}_i$ a ranking score f_i. MR defines an initial vector $\mathbf{f}^0 = [f_1^0, \ldots, f_n^0]^T$, in which $f_i^0 = 1$ if x_i is a query and $f_i^0 = 0$ otherwise. The cost function associated with f in MR is defined to be (Zhou et al. 2004a)

$$O(\mathbf{f}) = \frac{1}{2}(\sum_{i,j=1}^{n} w_{ij}(\frac{f_i}{\sqrt{D_{ii}}} - \frac{f_j}{\sqrt{D_{jj}}})^2 + \mu\|\mathbf{f} - \mathbf{f}^0\|^2), \quad (1)$$

where $\mu > 0$ is the regularization parameter and D is a diagonal matrix with $D_{ii} = \sum_j w_{ij}$. The algorithm can also be designed as an iterative form as follows

$$\mathbf{f}(t+1) = \beta S \mathbf{f}(t) + (1-\beta)\mathbf{f}^0, \quad (2)$$

where $S = D^{-1/2} W D^{-1/2}$ and $\beta = \mu/(1+\mu)$. During one iteration t, each data point receives information from its neighbors and retains its initial assignment. At last, the algorithm converges to

$$\mathbf{f}^* = (I_n - \beta S)^{-1}\mathbf{f}^0 = K\mathbf{f}^0. \quad (3)$$

It is important to note that the parameter β should stay in the range of $[0, 1)$[1], otherwise the algorithm cannot converge.

The advantages of manifold ranking (MR) are as follows:

- In Eq.(3), if the query is in the database (*i.e.*, the query is one of the column of X), the matrix K can be precomputed, which makes the online computation process light weight;
- MR's input is a query list which makes it quite suitable for the *relevance feedback*, *i.e.*, a user provides some relevant/irrelevant samples in the initial results returned by the system and the system ranks the database again. It is very convenient for MR to take the advantages of relevance feedback – just simply change the query vector $\mathbf{f}^0$.

Meanwhile the algorithm has some disadvantages:

- The pre-computation is useful for the queries in the database. However, for the queries out of the database (out-of-sample), we have to re-compute the matrix K, which leads to $O(n^3)$ computational complexity. Thus, MR is not friendly to the new data samples.
- It is not clear for MR that how to accumulate the users feedbacks for future searches. Actually, the feedbacks provided by the users can be regarded as relevance constraints of the database and naturally can help to improve the ranking for a future query (Si et al. 2006).

Bregman Matrix Divergence

The Bregman divergence (Bregman 1967) can be used to measure the closeness of two vectors. Let $\varphi : \Delta \to \mathbb{R}$ be a continuously differentiable, real-valued strictly convex function defined on a convex set Δ, The Bregman divergence associated with φ is defined as

$$D_\varphi(\mathbf{x}, \mathbf{x}_0) = \varphi(\mathbf{x}) - \varphi(\mathbf{x}_0) - (\mathbf{x} - \mathbf{x}_0)^T \nabla\varphi(\mathbf{x}_0). \quad (4)$$

Intuitively, this definition can be regarded as the difference between the value of φ at point x and the value of the first-order Taylor expansion of φ around point x_0 evaluated at point x. For example, if $\varphi(\mathbf{x}) = \mathbf{x}^T\mathbf{x}$, then the corresponding Bregman divergence is the squared Euclidean distance: $D_\varphi(\mathbf{x}, \mathbf{x}_0) = \|\mathbf{x} - \mathbf{x}_0\|^2$. The definition of Bregman divergence can be naturally extended to real-valued symmetric $n \times n$ matrices (Kulis, Sustik, and Dhillon 2006). Given a continuously differentiable, strictly convex function φ, the Bregman matrix divergence is defined as

$$D_\varphi(X, X_0) = \varphi(X) - \varphi(X_0) - \langle \nabla\varphi(X_0), X - X_0 \rangle. \quad (5)$$

Similarly, if $\varphi(X) = \|X\|_F$, the Frobenius norm of a matrix, then $D_\varphi(X, X_0) = \|X - X_0\|_F^2$.

In this paper, we focus on the convex log-determinant function $\varphi(X) = -\log\det X$, which is the Burg entropy of X's eigenvalues, *i.e.*, $\varphi(X) = -\sum_i \log\lambda_i$. The resulting Bregman matrix divergence is

$$D_{\ell d}(X, X_0) = \mathrm{tr}(XX_0^{-1}) - \log\det(XX_0^{-1}) - n, \quad (6)$$

which is called the Log-Determinant divergence (Kulis, Sustik, and Dhillon 2006).

A New Derivation of Manifold Ranking

In this section, we will derive the MR algorithm from a Bregman divergence optimization framework. Based on the new formulation, some extensions can be naturally derived to overcome the shortcomings of the traditional MR approach.

Let $Y = [\mathbf{y}_1, \ldots, \mathbf{y}_n] \in \mathbb{R}^{p \times n}$ be the data representation in a new feature space of the data samples. That is to say $\mathbf{y}_i = \Psi(\mathbf{x}_i)$, for $i = 1, \ldots, n$, where Ψ is a transformation function of the data to the new feature space. We define the matrirx K as

$$K = Y^T Y. \quad (7)$$

It is easy to check that the matrix K is always *positive semi-definite*, since for any vector $\mathbf{v}$, $\mathbf{v}^T K \mathbf{v} = \|Y\mathbf{v}\|^2 \geq 0$ holds. Now we present our main theorem as follows:

Theorem 1 *The matrix K in the manifold ranking formulation (Eq.(3)) is the solution of the following optimization problem:*

$$\begin{aligned} \min_{K} \quad & D_{\ell d}(K, I) \\ \textit{s.t.} \quad & \sum_{i,j} \|\tfrac{1}{\sqrt{D_{ii}}}\mathbf{y}_i - \tfrac{1}{\sqrt{D_{jj}}}\mathbf{y}_j\|^2 w_{ij} \leq \delta, \\ & K \succeq 0, \end{aligned} \quad (8)$$

where δ is a parameter controlling the smoothness constraint which makes the nearby samples having close distances in the new space.

[1] when $\beta = 0$, $\mathbf{f}^*$ is always equal to the initial $\mathbf{f}^0$

The optimization problem (8) finds a K closest to the identity matrix measured by the Log-Determinant divergence with a normalized Laplacian smoothness constraint. Note that the first constraint can be written as a matrix form (Zhou et al. 2004a):

$$\sum_{i,j} \| \frac{1}{\sqrt{D_{ii}}}\mathbf{y}_i - \frac{1}{\sqrt{D_{jj}}}\mathbf{y}_j \|^2 w_{ij} = \text{tr}(YLY^T), \qquad (9)$$

where $L = I - S$ ($S = D^{-1/2}WD^{-1/2}$) is the normalized graph Laplacian (Chung 1997).

Substituting the objective function with Eq. (6) and using Lagrange multiplier, the optimization problem (8) has the equivalent formulation as follows:

$$\min_{K \succeq 0} \quad \text{tr}(KI^{-1}) - \log\det(K) + \alpha\text{tr}(YLY^T), \qquad (10)$$

where $\alpha \geq 0$ is the Lagrange multiplier for the constraint. Since $\text{tr}(YLY^T) = \text{tr}(Y^TYL) = \text{tr}(KL)$, we can further simplify the optimization problem (10) as

$$\min_{K \succeq 0} \quad \text{tr}(KB) - \log\det(K), \qquad (11)$$

where $B = I + \alpha L$ is a *positive-definite* matrix. The optimal solution K^* of the above optimization problem is (Hoi, Liu, and Chang 2010):

$$K^* = B^{-1} = (I + \alpha L)^{-1}. \qquad (12)$$

Note that $B = I + \alpha L = (1+\alpha)(I - \frac{\alpha}{(1+\alpha)}S)$, then

$$K^* = (I - \beta S)^{-1}, \qquad (13)$$

where $\beta = \frac{\alpha}{1+\alpha}$. The positive constant part $(1+\alpha)^{-1}$ is ignored because it is simply a scaling factor and does not influence the ranking. The parameter $\beta = \frac{\alpha}{1+\alpha}$ is in the range of $[0, 1)$, when $\alpha \geq 0$. Thus, K^* is exactly equal to the matrix K in Eq.(3). So we have our Theorem 1.

New Extensions of Manifold Ranking

The above Bregman divergence optimization view of manifold ranking shows that MR essentially learns an optimal matrix K "closest" to the identity matrix under certain constraints. We name this formulation as **DMR** (divergence view of MR) for distinguish. In this section, we propose two effective extensions.

Extension 1: An Efficient Extension

MR learns an optimal matrix K "closest" to the identity matrix, which is not informative. We can naturally use other matrices (derived from the data) instead of the identity matrix. In this paper, we use a *Gaussian kernel* matrix and we want the optimal matrix to be close to the Gaussian kernel matrix under certain constraints. Each element of the Gaussian kernel matrix is calculated by

$$K^G_{ij} = \exp\left(-d^2(\mathbf{x}_i, \mathbf{x}_j)/2\sigma^2\right), \qquad (14)$$

where σ is the window size parameter and $d(\mathbf{x}_i, \mathbf{x}_j)$ returns the Euclidean distance between $\mathbf{x}_i$ and $\mathbf{x}_j$. Replacing the matrix I in Eq.(10) by K^G, the object of DMR becomes

$$\min_{K \succeq 0} \quad \text{tr}(KH) - \log\det(K) + \alpha\text{tr}(YLY^T), \qquad (15)$$

where $H = K^{G^{-1}}$. The solution of the above problem

$$K^* = (H + \alpha L)^{-1}. \qquad (16)$$

If matrix $B_E \doteq H + \alpha L$ is not positive-definite, one can simply add a ridge term as $\tilde{B}_E = B_E + r_I I$ to replace it, where r_I is a positive constant.

So far, we still need to inverse a $n \times n$ matrix $H + \alpha L$. Assume the mapping from X to Y is linear, *i.e.*, $Y = U^TX$, where U is a $m \times p$ matrix, we have

$$\begin{aligned} & \text{tr}(KH) - \log\det(K) + \alpha\text{tr}(YLY^T) \\ = \; & \text{tr}(X^TUU^TXH) - \log\det(X^TUU^TX) \\ & \qquad + \alpha\text{tr}(U^TXLX^TU) \\ = \; & \text{tr}(AXHX^T) - \log\det(XX^T) - \log\det(A) \\ & \qquad + \alpha\text{tr}(AXLX^T), \end{aligned}$$

where $A = UU^T \succeq 0$ and the second equality use the property that $\det(AB) = \det(A)\det(B)$.

Thus, the optimization problem (15) becomes

$$\min_{A \succeq 0} \quad \text{tr}(A(XHX^T + \alpha XLX^T)) - \log\det(A), \qquad (17)$$

where A is a $m \times m$ matrix. To solve this problem, we only need to inverse a matrix of size $m \times m$, which remains the same as the size of the data set (n) grows.

It is not hard to see that we actually learned a distance metric d_A:

$$\begin{aligned} d^2_A(\mathbf{x}_i, \mathbf{x}_j) &= \|U^T\mathbf{x}_i - U^T\mathbf{x}_i\|^2 \\ &= (\mathbf{x}_i - \mathbf{x}_j)^T UU^T (\mathbf{x}_i - \mathbf{x}_j) \\ &= (\mathbf{x}_i - \mathbf{x}_j)^T A (\mathbf{x}_i - \mathbf{x}_j). \end{aligned} \qquad (18)$$

With the learned metric matrix A, K can be computed as $K = Y^TY = X^TAX$. However, the linear mapping from X to Y is too restricted. To narrow down the gap between the linear and nonlinear mapping, we use the learned distance metric d_A to estimate a new Gaussian kernel matrix. This is reasonable because our goal is finding a matrix "close to" the Gaussian kernel matrix. Then each element of the final matrix K^* is computed by

$$K^*_{ij} = \exp\left(-d^2_A(\mathbf{x}_i, \mathbf{x}_j)/2\sigma^2\right). \qquad (19)$$

Similar to Eq.(3), we can use the learned K^* for ranking as follows:

$$\mathbf{f}^* = K^*\mathbf{f}^0. \qquad (20)$$

where $\mathbf{f}^0$ is the query vector and $\mathbf{f}^*$ is the ranking score.

Given a sample point out of the database (*e.g.*, a new query), instead of updating the entire matrix K^*, we only need to compute a new column (and row) of the matrix. Without loss of generality, let $\widehat{X} = [X \; \mathbf{x}_t] \in \mathbb{R}^{m \times (n+1)}$ be the new data matrix where $\mathbf{x}_t$ is the new sample. Then the new optimal matrix $\widehat{K}^* \in \mathbb{R}^{(n+1)\times(n+1)}$ has the form of

$$\widehat{K}^* = \begin{bmatrix} K^* & \mathbf{k}_{nt} \\ \mathbf{k}^T_{nt} & 1 \end{bmatrix}. \qquad (21)$$

And we only need to compute the vector $\mathbf{k}_{nt} \in \mathbb{R}^n$ based on Eq. (19).

In MR, the most computational expensive step is the inversion of a $n \times n$ matrix, which has a complexity of $O(n^3)$.

While in our extension, we optimize a much smaller matrix A of size $m \times m$ ($m \ll n$). Then we use A to estimate the K^*. As a result, the complexity of the proposed extension is $O(m^3) + O(n^2)$. Moreover, given a new query, the traditional MR still needs $O(n^3)$ to propagate the ranking score. Our extension only needs $O(n)$ to update the matrix K^* as in Eq. (21).

The proposed extension is much more efficient than the traditional MR algorithm and we use **DMR**$_E$ (stands for Efficient DMR) to denote it.

Extension 2: Incorporating Pairwise Constraints

Our first extension uses a distance metric learning formulation to speed up traditional MR. It inspires us to further extend our algorithm to utilize the *pairwise constraints*, which have been fully studied in semi-supervised clustering and metric learning works (Xing et al. 2002; Yang et al. 2006; Davis et al. 2007; Hoi, Liu, and Chang 2008; 2010).

Assume that we are given two sets of pairwise constraints among the data set X:

$$\begin{aligned} \mathcal{S} &= \{(i,j) \mid \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ are judged to be similar}\} \\ \mathcal{D} &= \{(i,j) \mid \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ are judged to be dissimilar}\} \end{aligned}$$

where $\mathcal{S}$ is the set of similar pairwise constraints and $\mathcal{D}$ is the set of dissimilar pairwise constraints. In a retrieval system, it is not hard to accumulate these knowledge. For example, we can get the pairwise constraints from the click-through data or the relevance feedback.

The pairwise constraints require the pairs in $\mathcal{S}$ having shorter distances and the pairs in $\mathcal{D}$ having longer distances under the distance metric d_A. Following (Si et al. 2006; Hoi, Liu, and Chang 2008), we formulate the optimization problem (17) with pairwise constraints as:

$$\begin{aligned} \min_{A \succeq 0} \quad & \mathrm{tr}(A(XHX^T + \alpha XLX^T)) - \log\det(A) \\ & + \gamma_s \sum_{(i,j)\in\mathcal{S}} d_A(\mathbf{x}_i, \mathbf{x}_j)^2 - \gamma_d \sum_{(i,j)\in\mathcal{D}} d_A(\mathbf{x}_i, \mathbf{x}_j)^2, \end{aligned} \tag{22}$$

where parameters $\gamma_s \geq 0$ and $\gamma_d \geq 0$ control the strength of the constraints. Similar as before, the solution can be computed as

$$A^* = (XHX^T + \alpha XLX^T + \gamma_s M - \gamma_d C)^{-1}. \tag{23}$$

where the matrices M and C are defined as:

$$\begin{aligned} M &= \sum_{(i,j)\in\mathcal{S}} (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T \\ C &= \sum_{(i,j)\in\mathcal{D}} (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T. \end{aligned} \tag{24}$$

If the matrix to be inversed in Eq. (23) is not positive-definite, one can simply add a ridge term $r_I I$ (Hoi, Liu, and Chang 2010)

Similar to Extension 1, we can use Eq.(19-21) to estimate the matrix K and ranking the database given a query. We use **DMR**$_C$ (stands for DMR with Constraints) to denote the second extension.

Table 1: Basic statistics of the two image data sets.

	COREL	Caltech101
# of images	5,000	9,094
# of categories	50	101
# of images per category	100	(avg)90

Experimental Results

In this section, we use content-based image retrieval as an example to demonstrate the effectiveness of the proposed approach. The proposed **DMR**$_E$ is compared with several state-of-the-art unsupervised (without constraints) ranking methods and the proposed **DMR**$_C$ is compared with several state-of-the-art semi-supervised (with constraints) ranking methods.

Data Set and Experimental Setup

Our experiments are performed on two real world image data sets: COREL and Caltech101. The COREL data set is widely used in many CBIR works (Cai, He, and Han 2007a; 2007b; He et al. 2007; He, Cai, and Han 2008; Hoi, Liu, and Chang 2008; He 2010; Xu et al. 2011). We use a subset which consists of 5,000 images in 50 semantic categories, and each category has exactly 100 images. The Caltech101 data set[2] has 9,094 images belonging to 101 categories with about 40 to 800 images per category. Table 1 shows some important statistics of the data sets.

Image Features Image feature extraction is a key step for CBIR. A wide variety of global features were proposed in the past decades. In our experiments, we extract four kinds of effective features, and as a result, a 297-dimensional vector is used for each image (Zhu et al. 2008).

- Grid Color Moment: Each image is partitioned into 3×3 grids. For each grid, the color moments: mean, variance and skewness are extracted in each color channel (R, G, and B) respectively. Finally, we have an 81-dimensional grid color moment vector for each image.
- Edge: The Canny edge detector (Canny 1986) is used to obtain the edge map for the edge orientation histogram, which is quantized into 36 bins of 10 degrees each. An additional bin is to count the number of pixels without edge information. Hence, a 37-dimensional vector is used.
- Gabor Wavelets Texture: Each image is first scaled to 64×64 pixels. The Gabor wavelet transform (Lades et al. 1993) is then applied on the scaled image with 5 levels and 8 orientations, which results in 40 subimages. For each subimage, 3 moments are calculated: mean, variance and skewness. Thus, a 120-dimensional vector is used.
- Local Binary Pattern: The LBP (Ojala, Pietikäinen, and Harwood 1996) is a gray-scale texture measure derived from a general texture definition in a local neighborhood. A 59-dimensional LBP histogram vector is adopted.

[2] http://www.vision.caltech.edu/Image_Datasets/Caltech101

Evaluation Metric There are many metrics to evaluate the performance of a CBIR system. In reality, images in top returned pages receive most of the interests and attentions from the users. Thus the Precision at K (P@K) metric is very practical to evaluate the retrieval performance. In addition, we also use NDCG and MAP values.

NDCG is a wildly used metric to evaluate a ranked list (Manning, Raghavan, and Schutze 2008). NDCG@K is defined as:

$$NDCG@K = \frac{1}{IDCG} \times \sum_{i=1}^{K} \frac{2^{r_i - 1}}{log_2(i+1)}, \qquad (25)$$

where r_i is 1 if the item at position i is a relevant item and 0 otherwise. IDCG is chosen so that the perfect ranking has a NDCG value 1.

MAP (Mean Average Precision) provides a single-figure measure of quality across recall levels (Manning, Raghavan, and Schutze 2008). For a single query j, Average Precision is the average of precisions computed at the point of each correctly retrieved item $(d_1, \ldots, d_{mj})$ in the ranked list, and this value is then averaged over the query set U:

$$MAP(U) = \frac{1}{|U|} \sum_{j=1}^{|U|} \frac{1}{m_j} \sum_{k=1}^{m_j} Precision(R_{jk}), \qquad (26)$$

where R_{jk} is the set of ranked results from the top result until you get to item d_k. In our experiments, only the the top 200 returned images for each query are used.

Experiments without Constraints

We first compare some ranking methods without using pairwise constraints. These methods include dimensionality reduction algorithms, metric learning algorithms, and the original MR approach. Specifically, they are:

- Eud: the simple **baseline** method using Euclidean distance for ranking.
- PCA: the most popular linear unsupervised dimensionality reduction method (Shlens 2005). The reduced dimension is set to 30 in the experiment.
- Mah: a standard distance metric method for which the matrix $A = C^{-1}$, where C is the sample covariance matrix.
- MR: the original manifold ranking which is the most important comparison, since we are just want to improve it.
- DMR$_E$: the first extension proposed in this paper.

Results on COREL and Caltech Each image in the COREL data set is used as a query and the remaining images are served as the database. The retrieval performance is averaged over all the queries. In the left part of Table 3, we record the Precision, NDCG and MAP values of all the approaches. Each figure in the table has two parts: the left part is the mean value and the right part is the relative improvement over the baseline Eud method. The best result in each row for the corresponding group of methods is in bold type. Similarly, for the Caltech101 data set, we use each image as a query and the retrieval performance is averaged. In the left part of Table 4, we record the Precision, NDCG and MAP values of all the approaches. We will provide a detailed analysis in the next subsection.

Table 2: Statistics of the generated constraints.

	COREL	Caltech101
# of queries	300	606
# of judgements per query	20	20
# of total positive judgements	2,147	1,730

Experiments with Constraints

Constraints Generation A real CBIR system can always ask the users to provide feedbacks on their queries. These feedback information can naturally be used as pairwise constraints for future search.

In our experiment, we simulate this procedure to generate the constraints. We randomly select some images from the database as the queries. For each query, we automatically generates 20 feedback images (closest to the query image in Euclidean distance). The candidate images are assigned positive or negative to the query according to their labels. These information are stored in the system as the constraints.

We use 300 random selected images as queries in COREL database and 606 images in Caltech101 database (6 queries for each category). Some statistics of the generated constraints are listed in Table 2.

Compared Methods With constraints information, we compare our proposed approach with several state-of-the-art semi-supervised metric learning algorithms listed as follows:

- Xing: a well known distance matric learning approach, which minimizes the similar pair distances and maximizes the dissimilar pair distances (Xing et al. 2002). The learned distance metric is used for image retrieval.
- ITML: information theoretic metric learning with pairwise constraints (Davis et al. 2007). We run it over various parameters and select the best one. The learned metric distance is used for image retrieval.
- LRML: the laplacian regularized metric learning with pairwise constraints, which utilizes both the labeled and unlabeled data (Hoi, Liu, and Chang 2008; 2010). The learned distance metric is used for image retrieval.
- DMR$_C$: the second extension proposed in this paper.

Results on COREL and Caltech Similar to the settings of the previous experiment, we record the results of all the above methods on COREL and Caltech101 in the right part of Table 3 and Table 4 respectively. The detailed analysis of these results will be provided later.

Parameter Selection

There are three important parameters in our algorithms: α, γ_s and γ_d. They control the balance between unconstrained and constrained data. Since simultaneously tuning three parameters is too complex, we use a simpler strategy. First of all, we find the best α for DMR$_S$ and fix it for DMR$_C$. Then we assume that γ_d is a ratio of γ_s, *i.e.*, $\gamma_d = r \cdot \gamma_s$. Similar to (Si et al. 2006; Hoi, Liu, and Chang 2008), we use $r = \frac{1}{3}$

Table 3: Performance Comparisons on COREL data set for all the algorithms in terms of Precision (P), NDCG (N) and MAP. For each result, we record the mean value (%) of all the queries and the relative improvement over the baseline method Eud. The best result in each row is indicated by the bold font.

	Methods without constraints					Methods with constraints			
Method	Eud	PCA	Mah	MR	DMR$_E$	Xing	ITML	LRML	DMR$_C$
P@10	42.12	40.19 -4.57%	40.05 -4.91%	42.92 +1.90%	**44.55 +5.77%**	33.55 -20.35%	46.23 +9.78%	49.56 +17.67%	**51.21 +21.58%**
P@20	35.98	34.74 -3.46%	33.20 -7.74%	38.00 +5.59%	**38.15 +6.03%**	28.39 -21.10%	41.41 +15.08%	44.03 +22.36%	**45.75 +27.14%**
P@30	32.17	31.25 -2.86%	29.10 -9.53%	**34.67 +7.77%**	34.06 +5.88%	25.17 -21.77%	38.13 +18.54%	40.42 +25.65%	**42.09 +30.83%**
N@10	44.98	42.90 -4.64%	43.34 -3.66%	45.70 +1.59%	**47.51 +5.61%**	36.21 -19.5%	48.60 +8.04%	52.10 +15.83%	**53.81 +19.61%**
N@20	39.67	38.11 -3.92%	37.36 -5.81%	41.26 +4.02%	**41.98 +5.83%**	31.65 -20.20%	44.38 +11.88%	47.32 +19.31%	**49.06 +23.68%**
N@30	36.17	34.92 -3.47%	33.56 -7.22%	38.20 +5.62%	**38.25 +5.76%**	28.67 -20.75%	41.42 +14.52%	44.07 + 21.83%	**45.76 +26.50%**
MAP	33.49	32.38 -3.31%	32.96 -1.59%	**36.07 +7.71%**	35.44 +5.82%	27.67 -17.37%	37.71 +12.58%	40.14 +19.84%	**41.65 +24.35%**

Table 4: Performance Comparisons on Caltech101 data set for all the algorithms in terms of Precision (P), NDCG (N) and MAP. For each result, we record the mean value (%) of all the queries and the relative improvement over the baseline method Eud. The best result in each row is indicated by the bold font.

	Methods without constraints					Methods with constraints			
Method	Eud	PCA	Mah	MR	DMR$_E$	Xing	ITML	LRML	DMR$_C$
P@10	35.50	34.03 -4.15%	37.34 +5.18%	34.98 -1.48%	**38.93 +9.65%**	33.92 -4.45%	37.49 +5.59%	39.82 +12.16%	**41.82 +17.80%**
P@20	32.52	31.27 -3.86%	33.75 +3.77%	32.47 -0.15%	**35.72 +9.85%**	30.75 -5.44%	35.10 +7.93%	36.91 +13.51%	**38.84 +19.44%**
P@30	30.73	29.55 -3.83%	31.54 +2.63%	30.88 +0.47%	**33.73 +9.77%**	28.77 -6.38%	33.50 +9.02%	35.05 +14.07%	**36.96 +20.26%**
N@10	37.28	35.63 -4.43%	39.26 +5.30%	36.79 -1.31%	**40.77 +9.37%**	35.56 -4.62%	38.93 +4.42%	41.53 +11.41%	**43.55 +16.83%**
N@20	34.55	33.12 -4.15%	36.06 +4.35%	34.38 -0.50%	**37.87 +9.60%**	32.76 -5.21%	36.74 +6.33%	38.89 +12.54%	**40.85 +18.21%**
N@30	32.85	31.51 -4.08%	33.99 +3.50%	32.84 -0.02%	**36.00 +9.59%**	30.92 -5.87%	35.26 +7.34%	37.14 +13.08%	**39.08 +18.97%**
MAP	33.25	31.56 -5.07%	34.46 +3.66%	35.23 +5.98%	**36.65 +10.25%**	31.36 -5.68%	35.36 +6.37%	38.34 +15.31%	**39.54 +18.94%**

for COREL. Finally, for COREL data set, we use $\alpha = 5$, $\gamma_s = 2$, and $r = \frac{1}{3}$. For Caltech101 data set, we use $\alpha = 1$, $r_s = 1$ and $r = \frac{1}{4}$. In addition, $k = 5$ for the construction of kNN graph and $\sigma = 1$ for the Gaussian kernel function and $r_I = 100$.

Results Analysis

For the COREL data set, the used image features are very good for discrimination, so the baseline method (simply uses the Euclidean distance) performs reasonably well. In the group of unsupervised methods, we can easily see from Table 3 that DMR$_E$ and MR are better than rest of the three. DMR$_E$ is slightly better than MR. In the group of semi-supervised methods, DMR$_C$ outperforms all the other methods. This is probably because DMR$_C$ utilizes the geometric structure of the data (the graph Laplacian), the constraint information, as well as the initial Gaussian kernel matrix. The relative low performance of Xing and ITML algorithms are mainly caused by the over-fitting problem (Si et al. 2006), since they only learn from the 'labeled' points which are a very small portion of the data in our experiment.

For the Caltech101 data set, the performance of the baseline Eud method becomes worse, simply because of the larger number of categories, the unbalanced category size, and the complicated content in the images. Thus, directly using the Euclidean distance is not good for ranking. Since the graph used in MR is based the Euclidean structure of the data, the performance of MR is also not good. Our approach DMR$_E$ significantly outperforms all the other unsupervised methods. When the constraints are introduced, our method DMR$_C$ archives better performance. And it is better than the rest semi-supervised methods. This demonstrates that the constraints (feedbacks) are really useful in a CBIR system.

Conclusion and Future Work

We present a new viewpoint of manifold ranking – it learns an optimal kernel matrix under the Bregman matrix divergence metric. Based on this new formulation, two efficient and effective extensions, named DMR$_E$ and DMR$_C$, are proposed. For the sake of efficiency, we transform the kernel matrix learning problem to a metric learning problem. Moreover, the constraints can easily be used in our framework to derive a better ranking algorithm. Extensive experiments on COREL and Caltech101 demonstrate the effectiveness of our approach. In the future work, we will investigate the relationship of our work to many learning to rank literatures (Liu 2009), which is a hot research topic in recent years.

Acknowledgments

This work was supported in part by National Natural Science Foundation of China under Grant 91120302 and 61173186, National Basic Research Program of China (973 Program) under Grant 2011CB302206, Fundamental Research Funds for the Central Universities, Program for New Century Excellent Talents in University (NCET-09-0685), Zhejiang Provincial Natural Science Foundation under Grant No Y1101043 and Foundation of Zhejiang Provincial Educational Department under Grant No Y201018240.

References

Bengio, Y.; Paiement, J.; Vincent, P.; Delalleau, O.; Le Roux, N.; and Ouimet, M. 2004. Out-of-sample extensions for lle, isomap, mds, eigenmaps, and spectral clustering. *Advances in neural information processing systems* 16:177–184.

Bregman, L. 1967. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR computational mathematics and mathematical physics* 7(3):200–217.

Brin, S., and Page, L. 1998. The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems* 30(1-7):107–117.

Cai, D.; He, X.; and Han, J. 2007a. Regularized regression on image manifold for retrieval. In *Proceedings of the 9th ACM SIGMM International Workshop on Multimedia Information Retrieval*, 11–20.

Cai, D.; He, X.; and Han, J. 2007b. Spectral regression: A unified subspace learning framework for content-based image retrieval. In *Proceedings of the 15th ACM International Conference on Multimedia*, 403–412.

Canny, J. 1986. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (6):679–698.

Cheng, X.; Du, P.; Guo, J.; Zhu, X.; and Chen, Y. 2011. Ranking on data manifold with sink points. *IEEE Transactions on Knowledge and Data Engineering* (99):1–1.

Chung, F. 1997. *Spectral graph theory*. Number 92. Amer Mathematical Society.

Davis, J.; Kulis, B.; Jain, P.; Sra, S.; and Dhillon, I. 2007. Information-theoretic metric learning. In *Proceedings of the 24th international conference on Machine learning*, 209–216.

He, J.; Li, M.; Zhang, H.; Tong, H.; and Zhang, C. 2004. Manifold-ranking based image retrieval. In *Proceedings of the 12th annual ACM international conference on Multimedia*, 9–16.

He, J.; Li, M.; Zhang, H.; Tong, H.; and Zhang, C. 2006. Generalized manifold-ranking-based image retrieval. *IEEE Transactions on Image Processing* 15(10):3170–3177.

He, X.; Min, W.; Cai, D.; and Zhou, K. 2007. Laplacian optimal design for image retrieval. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 119–126.

He, X.; Cai, D.; and Han, J. 2008. Learning a maximum margin subspace for image retrieval. *IEEE Transactions on Knowledge and Data Engineering* 20(2):189–201.

He, X. 2010. Laplacian regularized d-optimal design for active learning and its application to image retrieval. *IEEE Transactions on Image Processing* 19(1):254–263.

Hoi, S.; Liu, W.; and Chang, S. 2008. Semi-supervised distance metric learning for collaborative image retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition, 2008.*, 1–7.

Hoi, S.; Liu, W.; and Chang, S. 2010. Semi-supervised distance metric learning for collaborative image retrieval and clustering. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP)* 6(3).

Kulis, B.; Sustik, M.; and Dhillon, I. 2006. Learning low-rank kernel matrices. In *Proceedings of the 23rd international conference on Machine learning*, 505–512.

Lades, M.; Vorbruggen, J.; Buhmann, J.; Lange, J.; von der Malsburg, C.; Wurtz, R.; and Konen, W. 1993. Distortion invariant object recognition in the dynamic link architecture. *IEEE Transactions on Computers* 42(3):300–311.

Liu, T. 2009. Learning to rank for information retrieval. *Foundations and Trends in Information Retrieval* 3(3):225–331.

Manning, C.; Raghavan, P.; and Schutze, H. 2008. *Introduction to information retrieval*, volume 1. Cambridge University Press Cambridge.

Ojala, T.; Pietikäinen, M.; and Harwood, D. 1996. A comparative study of texture measures with classification based on featured distributions. *Pattern recognition* 29(1):51–59.

Shlens, J. 2005. A tutorial on principal component analysis. *Measurement* 51.

Si, L.; Jin, R.; Hoi, S.; and Lyu, M. 2006. Collaborative image retrieval via regularized metric learning. *Multimedia Systems* 12(1):34–44.

Wan, X.; Yang, J.; and Xiao, J. 2007. Manifold-ranking based topic-focused multi-document summarization. In *Proceedings of IJCAI*, volume 7, 2903–2908.

Xing, E.; Ng, A.; Jordan, M.; and Russell, S. 2002. Distance metric learning, with application to clustering with side-information. *Advances in neural information processing systems* 15:505–512.

Xu, B.; Bu, J.; Chen, C.; Cai, D.; He, X.; Liu, W.; and Luo, J. 2011. Efficient manifold ranking for image retrieval. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information*, 525–534.

Yang, L.; Jin, R.; Sukthankar, R.; and Liu, Y. 2006. An efficient algorithm for local distance metric learning. In *Proceedings of the National Conference on Artificial Intelligence*.

Yuan, X.; Hua, X.; Wang, M.; and Wu, X. 2006. Manifold-ranking based video concept detection on large database and feature pool. In *Proceedings of the 14th annual ACM International Conference on Multimedia*, 623–626.

Zhou, D.; Bousquet, O.; Lal, T.; Weston, J.; and Schölkopf, B. 2004a. Learning with local and global consistency. *Advances in neural information processing systems* 16:321–328.

Zhou, D.; Weston, J.; Gretton, A.; Bousquet, O.; and Schölkopf, B. 2004b. Ranking on data manifolds. *Advances in neural information processing systems* 16:169–176.

Zhu, J.; Hoi, S.; Lyu, M.; and Yan, S. 2008. Near-duplicate keyframe retrieval by nonrigid image matching. In *Proceeding of the 16th ACM international conference on Multimedia*, 41–50.