Click-through-based Cross-view Learning for Image Search*

Yingwei Pan [1], Ting Yao [2,3], Tao Mei [4], Houqiang Li [1], Chong-Wah Ngo [2,3], Yong Rui [4]
[1] University of Science and Technology of China, Hefei, China
[2] City University of Hong Kong, Kowloon, Hong Kong
[3] Shenzhen Research Institute, City University of Hong Kong, Shenzhen, China
[4] Microsoft Research, Beijing, China
{panyw, tingyao}.ustc@gmail.com; lihq@ustc.edu.cn;
cscwngo@cityu.edu.hk; {tmei, yongrui}@microsoft.com

ABSTRACT

One of the fundamental problems in image search is to rank image documents according to a given textual query. Existing search engines highly depend on surrounding texts for ranking images, or leverage the query-image pairs annotated by human labelers to train a series of ranking functions. However, there are two major limitations: 1) the surrounding texts are often noisy or too few to accurately describe the image content, and 2) the human annotations are resourcefully expensive and thus cannot be scaled up.

We demonstrate in this paper that the above two fundamental challenges can be mitigated by jointly exploring the cross-view learning and the use of click-through data. The former aims to create a latent subspace with the ability in comparing information from the original incomparable views (i.e., textual and visual views), while the latter explores the largely available and freely accessible click-through data (i.e., "crowdsourced" human intelligence) for understanding query. Specifically, we propose a novel cross-view learning method for image search, named Click-through-based Cross-view Learning (CCL), by jointly minimizing the distance between the mappings of query and image in the latent subspace and preserving the inherent structure in each original space. On a large-scale click-based image dataset, CCL achieves the improvement over Support Vector Machine-based method by 4.0% in terms of relevance, while reducing the feature dimension by several orders of magnitude (e.g., from thousands to tens). Moreover, the experiments also demonstrate the superior performance of CCL to several state-of-the-art subspace learning techniques.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Retrieval models*

General Terms

Algorithm, Experimentation.

Keywords

Image search, cross-view learning, subspace learning, click-through data, DNN image representation.

1. INTRODUCTION

Keyword-based image search has received intensive research attention since the early of 1990s [20]. The significance of the topic can be partly reflected from the huge volume of published papers, particularly for addressing the problems of learning the rank or similarity functions. Despite these efforts, the fact that the queries (texts) and search targets (images) are of two different modalities (or views) has resulted in the open problem of "semantic gap." Specifically, a query in the form of textual keywords is not directly comparable with the visual content of images. The commercial search engines to date primarily reply on textual features extracted from the surrounding texts of images. However, the text description might not fully depict the salient aspect of visual content, not to mention that some images actually do not come along with any text description. One feasible solution is learning image rankers from the query-image pairs labeled by human subjects. However, the labeling process is generally time consuming, and in practice difficult to ensure the quality of labels. Furthermore, as the user search intents are not likely to always align with these pre-defined labels, image rankers used to suffer from the poor generalization performance.

Inspired by the success of multi-view embedding [31], this paper studies the cross-view (i.e., text to image views) search problem by learning a common latent subspace that allows direct comparison of text queries and images. Specifically, by mapping to the latent subspace, the relevance or similarity between a textual query and an image can be directly measured between their projections, making the information from the original incomparable cross-view space comparable in the shared subspace. In addition, the dimensionality of the latent subspace is significantly reduced compared with

* This work was performed when Yingwei Pan and Ting Yao were visiting Microsoft Research as research interns. The first two authors contributed equally to this work.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
SIGIR'14, July 6-11, 2014, Gold Coast, Australia.
Copyright 2014 ACM 978-1-4503-2257-7/14/07 ...$15.00.
http://dx.doi.org/10.1145/2600428.2609568.

that of any input view, making the memory costs much saved for existing search systems.

Moreover, we consider exploring user click-through data, aiming to bridge the user intention gap for image search. In general, image rankers obtain training data by manually labelling the relevance of query-image pairs. However, it is difficult to fathom user intent based on the queries, especially for those ambiguous queries. For example, given the query "mustang cobra," experts tend to label the images of animal "mustang" and "cobra" as highly relevant. However, empirical evidence suggests that most users wish to retrieve images of a car of brand "mustang cobra." The experts' labels therefore could be erroneous. This will bias the training set and the ranker will be learned sub-optimal. On the other hand, the click-through data provide an alternative to address this problem. In an image search engine, users browse image search results before clicking a specific image. The decision to click is likely dependent on the relevance of an image. Therefore, the click data can serve as a reliable and implicit feedback for image search. We hypothesize that, most of the clicked images are relevant to the given query judged by the real users.

By jointly integrating cross-view learning and click-through data, this paper presents a novel Click-through-based Cross-view Learning (CCL) approach to image search, as shown in Figure 1. Specifically, a bipartite graph between the user queries and images is constructed based on the search logs from a commercial image search engine. An edge between a query and an image is established when the users who issued the query clicked the image. Moreover, the textual and visual space is formed by constructing a graph on each view, respectively. The link between every two nodes in each space represents the query or image similarity. The spirit of CCL is to learn a latent subspace in the way of minimizing the distance between the mappings of query and image, while preserving the inherent structure in each original space. After the optimization of subspace learning, the relevance score between a query and an image in the original spaces can be directly computed based on their mappings. For any query, the image search list will be returned by sorting their relevance scores with the query.

In summary, this paper makes the following contributions:

- We study the problem of keyword-based image search by jointly exploring cross-view learning and the use of click-through data. To the best of knowledge, this paper represents one of the first efforts towards this target in the information retrieval research community.
- We propose a novel click-through-based cross-view learning (CCL), which aims to learn a latent subspace by simultaneously minimizing the distance between the mappings of query and image in the latent subspace, and preserving the structure in each original space. By mapping to the subspace, text queries and visual images can be directly compared.
- We evaluate the proposed click-through based image search approach on a large-scale click-based image dataset with over 23 millions of log records, which were sampled from one-year click data of a commercial image search engine.

The remaining sections are organized as follows. Section 2 describes related work on multi-view embedding and the use of click data, while Section 3 presents our click-through-based cross-view learning method. Section 4 provides empirical evaluations, followed by the discussions and conclusions in Section 5.

2. RELATED WORK

We briefly group the related work into two categories: multi-view embedding, and search by using click data. The former draws upon research in integrating multiple views to improve learning performance by exploiting either the consensus or the complementary principle, while the latter investigates Web search by mining click-through data.

2.1 Multi-view Embedding

The research in this direction has proceeded along three dimensions: co-training [16][22][33], subspace learning [2][9][25], and multi-kernel learning [5][14][17].

Co-training seeks consensus on two distinct views of the data. Muslea *et al.* combined active learning with co-training and proposed robust semi-supervised learning algorithms [22]. Yu *et al.* developed a Bayesian undirected graphical model for co-training and a novel co-training kernel for Gaussian process classifiers [33]. Kumar *et al.* advanced co-training for data clustering and designed effective algorithms for multi-view data [16]. The idea of subspace learning is similar to co-training except the consensus is solved by learning a latent subspace shared by multiple views by assuming that the input views are generated from this latent subspace. Canonical correlation analysis (CCA) [9], a classical technique, explored the mapping matrices by maximizing the correlation between the projections in the subspace. Similarly, Partial Least Squares (PLS) also aims to model the relations between two or more sets of data by projecting them into the latent subspace [25]. The difference between CCA and PLS is that CCA utilizes cosine as the similarity function while PLS learns dot product. Later in [2], polynomial semantic indexing (PSI) is performed by learning two low-rank mapping matrices in a learning to rank framework, and then a polynomial model is considered to measure the relevance between query and document.

Different from co-training and subspace learning, multi-kernel learning exploits different kernels to different views and fuses them either linearly or non-linearly for exploring complementary properties of different views. In [17], a linear (or convex) combination of a set of predefined kernels were learned to identify a good target kernel for the applications. Later in [5], Kernel target alignment was proposed to learn the entries of a kernel matrix by using the outer product of the label vector as the ground-truth. Kloft *et al.* extended the multi-kernel learning framework to arbitrary l_p-norm by adding a regularizer over the mixing coefficients [14].

In summary, our work belongs to subspace learning. Different from these aforementioned subspace learning methods, our approach contributes by studying not only forming the shared latent subspace with the standard objective of subspace learning (i.e., the consensus between views is maximized) but also preserving the inherent structure in each original space.

2.2 Search by Using Click Data

Click-through data has been studied and analyzed widely with different Web mining techniques for improving the efficacy and usability of search engines. The use of the click-

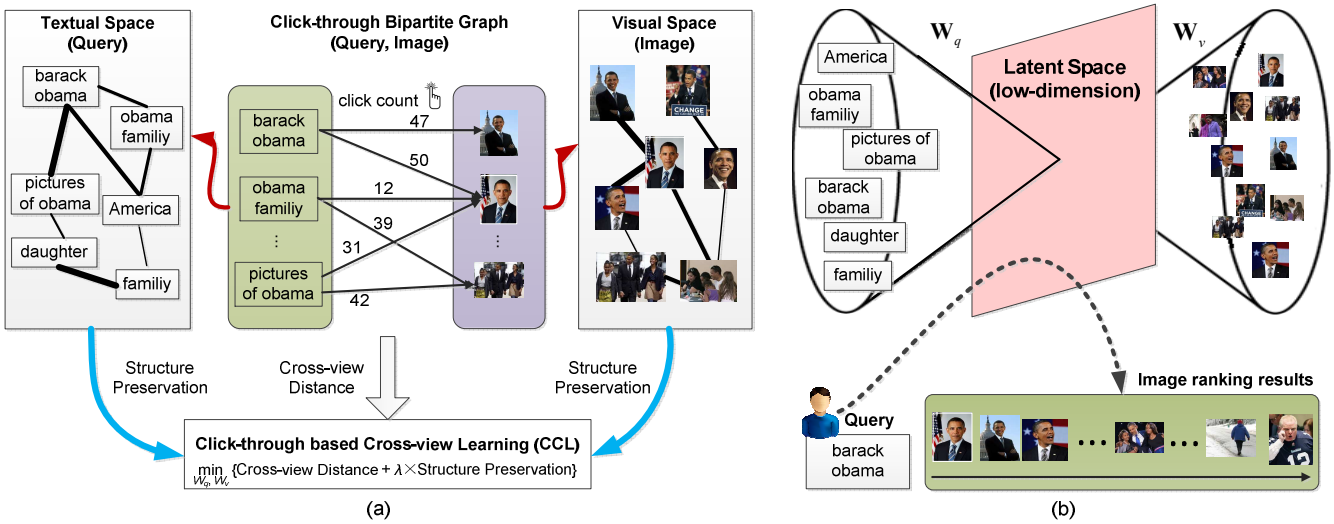

Figure 1: Click-through-based image search framework (better viewed in color). (a) Latent subspace learning between textual query and visual image: click-through-based cross-view learning by simultaneously minimizing the distance between the query and image mappings in the latent subspace (weighted by their clicks) and preserving the inherent structure in each original feature space. (b) With the learned mapping matrices $\mathbf{W}_q$ and $\mathbf{W}_v$, queries and images are projected into this latent subspace and then the distance in the latent subspace is directly taken as the relevance of query-image.

through data for query clustering was suggested by Beffer-man and Berger [3], who proposed an agglomerative clustering technique to identify related queries and Web pages. Wen *et al.* combined query content information and click-through information and applied a density-based method to cluster queries [28]. Mei *et al.* proposed an approach to query suggestion by computing the hitting time on a click graph [19]. Li *et al.* presented the use of click graphs in improving query intent classifiers [18].

There are also several approaches that have tried to model the representation of queries or documents on the click-through bipartite. In [1], the authors introduced another vectorial representation for the queries without considering the content information. Queries were represented as points in a high dimensional space, where each dimension corresponds to a unique URL. The weight assigned to each dimension was equal to the click frequency. Poblete *et al.* proposed the query-set document model by mining frequent query patterns to represent documents rather than the content information of the documents [24].

In addition, click-through data have also been used to learn the rank function [12]. Joachims *et al.* observed the relationship between clicked links and the relevance of the target pages by an eye tracking experiment [13]. Wu *et al.* formalized the learning of similarity as learning of mappings that maximize the similarities of query-documents pairs from the click-through bipartite graph [30]. For image search, click-through data has been found to be very reliable [6][11]. In [6], Craswell *et al.* built a query-image click graph and performed backward random walks to determine a probability distribution over images conditioned on the given query. In [11], Jain *et al.* reranked the image search results so as to promote images that are likely to be clicked to the top of the ranked list. Later in [27], an in-depth analysis of several ranking algorithms was performed on Flickr user log data to investigate the importance of many factors, including internal and external image popularity, the overall attentions, diversity, semantic categories and visual appearance. In [23], Pan *et al.* employed neighborhood graph search to find the nearest neighbors on an image similarity graph and further aggregated their clicked queries/click counts to get the labels of the new image. In another work by Yao *et al.* [32], by combining click-through and video document features for deriving a latent subspace, the dot product of the mappings in the latent subspace is taken as the similarity between videos and the similarity is further applied for video tagging tasks.

Most of the above approaches focus on leveraging both the click data and the features only from the textual view. Our work is different that we aim to compute the distance between the textual query and visual features from two different views on the observed query-image pairs and apply the learned distance for image search purpose.

3. CLICK-THROUGH-BASED CROSS-VIEW LEARNING

The main goal of click-through-based cross-view learning is to construct a latent common subspace with the ability of directly comparing textual query and image content. The training of CCL is performed simultaneously by minimizing the distance between query and image mappings in the latent subspace weighted by their clicks, and preserving the structure relationships between the training examples in the original feature space. In particular, the objective function of CCL is composed of two components, i.e., distance between views in the latent subspace, and the structure preservation in the original space. After we obtain the latent subspace, the relevance between query and image is directly measured by their mappings. The approach overview is shown in Figure 1.

In the following, we will first define the bipartite graph that naturally encodes user actions in the query log, followed

by constructing the two learning components of CCL. Then the joint overall objective and its optimization strategy are provided. Finally, the whole algorithm for image search is presented. It is worth noticing that although the two views here are visual (image) and textual (query), our approach is applicable to any other domain.

3.1 Notation

Let $\mathscr{G} = (\mathcal{V}, \mathcal{E})$ denote a click-through bipartite. $\mathcal{V} = Q \cup V$ is the set of vertices, which consists of a query set Q and an image set V. $\mathcal{E}$ is the set of edges between the query and image vertices. The number associated with the edge represents the clicked times in the image search results of the query. Suppose there are n triads $\{q_i, v_i, c_i\}_{i=1}^n$ generated from the click-through bipartite in total, where c_i is the click counts of image v_i in response to query q_i. Let $\mathbf{Q} = \{\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_n\}^\top \in \mathbb{R}^{n \times d_q}$ and $\mathbf{V} = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\}^\top \in \mathbb{R}^{n \times d_v}$ denote the query and image feature matrix, where $\mathbf{q}_i$ and $\mathbf{v}_i$ are the textual and visual feature of query q_i and image v_i, and d_q and d_v are the feature dimensionality, respectively. The click matrix $\mathbf{C}$ is a diagonal $n \times n$ matrix with its diagonal elements as c_i. Please note that the query q_i and image v_i may not be unique in each view as one single query can correspond to multiple clicked images.

3.2 Cross-view Distance

We assume that a low-dimensional common subspace exists for the representation of query and image. The linear mapping function can be derived from this subspace by

$$f(\mathbf{q}_i) = \mathbf{q}_i \mathbf{W}_q, \quad \text{and} \quad f(\mathbf{v}_i) = \mathbf{v}_i \mathbf{W}_v, \tag{1}$$

where d is the dimensionality of the common subspace, and $\mathbf{W}_q \in \mathbb{R}^{d_q \times d}$ and $\mathbf{W}_v \in \mathbb{R}^{d_v \times d}$ are the transformation matrices that project the query textual semantics and image content into the common subspace, respectively.

To measure the relations between the textual query and image visual content, one natural way is to measure the distance between their mappings in the latent subspace as

$$\begin{aligned} \min_{\mathbf{W}_q, \mathbf{W}_v} \; & tr\big((\mathbf{Q}\mathbf{W}_q - \mathbf{V}\mathbf{W}_v)^\top \mathbf{C} (\mathbf{Q}\mathbf{W}_q - \mathbf{V}\mathbf{W}_v)\big) \\ s.t. \quad & \mathbf{W}_q^\top \mathbf{W}_q = \mathbf{I}, \quad \mathbf{W}_v^\top \mathbf{W}_v = \mathbf{I} \end{aligned} \tag{2}$$

where $tr(\bullet)$ denotes the trace function. The matrices $\mathbf{W}_q$ and $\mathbf{W}_v$ have orthogonal columns, i.e., $\mathbf{W}_q^\top \mathbf{W}_q = \mathbf{W}_v^\top \mathbf{W}_v = \mathbf{I}$, where $\mathbf{I}$ is an identity matrix. The constrains restrict $\mathbf{W}_q$ and $\mathbf{W}_v$ to converge to reasonable solutions rather than go to $\mathbf{0}$ which is meaningless in practice.

Specifically, we view the click number of a query and an image as an indicator of their relevance. As most image search engines display results as thumbnails. The users can see the entire image before clicking on it. As such, barring distracting images and intent changes, users predominantly tend to click on images that are relevant to their query. Therefore, click data can serve as a reliable connection between the queries and images. The underlying assumption is that the higher the click number, the smaller the distance between the query and the image in the latent subspace. To learn this shared latent subspace, we intuitively incorporate the distance as a regularization on the mapping matrices $\mathbf{W}_q$ and $\mathbf{W}_v$ weighted by the click numbers.

3.3 Structure Preservation

Structure preservation or manifold regularization has been shown effective for semi-supervised learning [21] and multi-view learning [7]. This regularizer indicates that similar points in the original space should be mapped to the positions closely in the shared latent subspace. The estimation of the underlying structure can be measured by the appropriate pairwise similarity between the training samples. Specifically, it can be given by

$$\sum_{i,j=1}^{n} \mathbf{S}_{ij}^q \|\mathbf{q}_i \mathbf{W}_q - \mathbf{q}_j \mathbf{W}_q\|^2 + \sum_{i,j=1}^{n} \mathbf{S}_{ij}^v \|\mathbf{v}_i \mathbf{W}_v - \mathbf{v}_j \mathbf{W}_v\|^2, \tag{3}$$

where $\mathbf{S}^q \in \mathbb{R}^{n \times n}$ and $\mathbf{S}^v \in \mathbb{R}^{n \times n}$ denote the affinity matrices defined on the queries and images, respectively. Under the structure preservation criterion, it is reasonable to minimize Eq.(3), since it will incur a heavy penalty if two similar examples are mapped far away.

There are many ways of defining the affinity matrices $\mathbf{S}^q$ and $\mathbf{S}^v$. Inspired by [7], the elements are computed by Gaussian functions in this work, i.e.,

$$S_{ij}^t = \begin{cases} e^{-\frac{\|\mathbf{t}_i - \mathbf{t}_j\|^2}{\sigma_t^2}} & if\ \mathbf{t}_i \in N_k(\mathbf{t}_j)\ or\ \mathbf{t}_j \in N_k(\mathbf{t}_i) \\ 0 & otherwise \end{cases}, \tag{4}$$

where $t \in \{q, v\}$ for simplicity, i.e., t can be replaced by any one of q and v. σ_t is the bandwidth parameters. $N_k(\mathbf{t}_i)$ represents the set of k nearest neighbors of $\mathbf{t}_i$.

By defining the graph Laplacian $\mathbf{L}^t = \mathbf{D}^t - \mathbf{S}^t$ for $t \in \{q, v\}$, where $\mathbf{D}^t$ is a diagonal matrix with its elements defined as $\mathbf{D}_{ij}^t = \sum_j \mathbf{S}_{ij}^t$, Eq.(3) can be rewritten as

$$tr\big((\mathbf{Q}\mathbf{W}_q)^\top \mathbf{L}^q (\mathbf{Q}\mathbf{W}_q)\big) + tr\big((\mathbf{V}\mathbf{W}_v)^\top \mathbf{L}^v (\mathbf{V}\mathbf{W}_v)\big). \tag{5}$$

By minimizing this term, the similarity between examples in the original space can be preserved in the learned latent subspace. Therefore, we add this regularizer in our framework for optimization.

3.4 Overall Objective

The overall objective function integrates the distance between views in Eq.(2) and structure preservation in Eq.(5). Hence we get the following optimization problem

$$\begin{aligned} \min_{\mathbf{W}_q, \mathbf{W}_v} \; & tr\big((\mathbf{Q}\mathbf{W}_q - \mathbf{V}\mathbf{W}_v)^\top \mathbf{C} (\mathbf{Q}\mathbf{W}_q - \mathbf{V}\mathbf{W}_v)\big) \\ & + \lambda \big(tr\big((\mathbf{Q}\mathbf{W}_q)^\top \mathbf{L}^q (\mathbf{Q}\mathbf{W}_q)\big) + tr\big((\mathbf{V}\mathbf{W}_v)^\top \mathbf{L}^v (\mathbf{V}\mathbf{W}_v)\big) \big) \\ s.t. \quad & \mathbf{W}_q^\top \mathbf{W}_q = \mathbf{I}, \quad \mathbf{W}_v^\top \mathbf{W}_v = \mathbf{I} \end{aligned}, \tag{6}$$

where λ is the tradeoff parameter. The first term is the cross-view distance, while the second term represents structure preservation.

For simplicity, we denote $L(\mathbf{W}_q, \mathbf{W}_v)$ as the objective function in Eq.(6). Thus, the optimization problem can be rewritten as

$$\min_{\{\mathbf{W}_q, \mathbf{W}_v\}} L(\mathbf{W}_q, \mathbf{W}_v), \quad s.t. \;\; \mathbf{W}_q^\top \mathbf{W}_q = \mathbf{I}, \;\; \mathbf{W}_v^\top \mathbf{W}_v = \mathbf{I}. \tag{7}$$

The optimization above is a non-convex problem. Nevertheless, the gradient of the objective function with respect to $\mathbf{W}_q$ and $\mathbf{W}_v$ can be easily obtained as follows:

$$\begin{cases} \nabla_{\mathbf{W}_q} L(\mathbf{W}_q, \mathbf{W}_v) = 2\mathbf{Q}^\top \mathbf{C}(\mathbf{Q}\mathbf{W}_q - \mathbf{V}\mathbf{W}_v) + 2\lambda \mathbf{Q}^\top \mathbf{L}^q \mathbf{Q}\mathbf{W}_q \\ \nabla_{\mathbf{W}_v} L(\mathbf{W}_q, \mathbf{W}_v) = 2\mathbf{V}^\top \mathbf{C}(\mathbf{V}\mathbf{W}_v - \mathbf{Q}\mathbf{W}_q) + 2\lambda \mathbf{V}^\top \mathbf{L}^v \mathbf{V}\mathbf{W}_v \end{cases}. \tag{8}$$

3.5 Optimization

To address the difficult non-convex problem in Eq.(7) due to the orthogonal constrains, we use a gradient descent optimization procedure with curvilinear search [29] for a local optimal solution in this work.

In each iteration of the gradient descent procedure, given the current feasible mapping matrices $\{\mathbf{W}_q,\mathbf{W}_v\}$ and their corresponding gradients $\{\mathbf{G}_q = \nabla_{\mathbf{W}_q} L(\mathbf{W}_q,\mathbf{W}_v),\ \mathbf{G}_v = \nabla_{\mathbf{W}_v} L(\mathbf{W}_q,\mathbf{W}_v)\}$, we define the skew-symmetric matrices $\mathbf{P}_q$ and $\mathbf{P}_v$ as

$$\mathbf{P}_q = \mathbf{G}_q\mathbf{W}_q^\top - \mathbf{W}_q\mathbf{G}_q^\top,\ \ \mathbf{P}_v = \mathbf{G}_v\mathbf{W}_v^\top - \mathbf{W}_v\mathbf{G}_v^\top. \quad (9)$$

The new point can be searched as a curvilinear function of a step size τ, such that

$$\begin{aligned}\mathbf{F}_q(\tau) &= (\mathbf{I}+\frac{\tau}{2}\mathbf{P}_q)^{-1}(\mathbf{I}-\frac{\tau}{2}\mathbf{P}_q)\mathbf{W}_q,\\ \mathbf{F}_v(\tau) &= (\mathbf{I}+\frac{\tau}{2}\mathbf{P}_v)^{-1}(\mathbf{I}-\frac{\tau}{2}\mathbf{P}_v)\mathbf{W}_v.\end{aligned} \quad (10)$$

Then, it is easy to verify that $\mathbf{F}_q(\tau)$ and $\mathbf{F}_v(\tau)$ lead to several characteristics. The matrices $\mathbf{F}_q(\tau)$ and $\mathbf{F}_v(\tau)$ satisfy $(\mathbf{F}_q(\tau))^\top\mathbf{F}_q(\tau) = (\mathbf{F}_v(\tau))^\top\mathbf{F}_v(\tau) = \mathbf{I}$ for all $\tau \in R$. The derivatives with respect to τ are given as

$$\begin{cases}\mathbf{F}'_q(\tau) = -(\mathbf{I}+\frac{\tau}{2}\mathbf{P}_q)^{-1}\mathbf{P}_q(\frac{\mathbf{W}_q+\mathbf{F}_q(\tau)}{2})\\ \mathbf{F}'_v(\tau) = -(\mathbf{I}+\frac{\tau}{2}\mathbf{P}_v)^{-1}\mathbf{P}_v(\frac{\mathbf{W}_v+\mathbf{F}_v(\tau)}{2})\end{cases}. \quad (11)$$

In particular, we can obtain $\mathbf{F}_q{}'(0) = -\mathbf{P}_q\mathbf{W}_q$ and $\mathbf{F}_v{}'(0) = -\mathbf{P}_v\mathbf{W}_v$. Then, $\{\mathbf{F}_q(\tau),\mathbf{F}_v(\tau)\}_{\tau\geq 0}$ is a descent curve. We use the classical Armijo-Wolfe based monotone curvilinear search algorithm [26] to determine a suitable step τ as one satisfying the following conditions:

$$\begin{aligned}L(\mathbf{F}_q(\tau),\mathbf{F}_v(\tau)) &\leq L(\mathbf{F}_q(0),\mathbf{F}_v(0))\\ &\quad +\rho_1\tau L_\tau{}'(\mathbf{F}_q(0),\mathbf{F}_v(0)),\\ L_\tau{}'(\mathbf{F}_q(\tau),\mathbf{F}_v(\tau)) &\geq \rho_2 L_\tau{}'(\mathbf{F}_q(0),\mathbf{F}_v(0)),\end{aligned} \quad (12)$$

where ρ_1 and ρ_2 are two parameters satisfying $0<\rho_1<\rho_2<1$. $L_\tau{}'(\mathbf{F}_q(\tau),\mathbf{F}_v(\tau))$ is the derivative of L with respect to τ and is calculated by

$$\begin{aligned}&L_\tau{}'(\mathbf{F}_q(\tau),\mathbf{F}_v(\tau)) =\\ &-\sum_{t\in\{q,v\}} tr\Big(\mathbf{R}_t(\tau)^\top(\mathbf{I}+\frac{\tau}{2}\mathbf{P}_t)^{-1}\mathbf{P}_t(\frac{\mathbf{W}_t+\mathbf{F}_t(\tau)}{2})\Big),\end{aligned} \quad (13)$$

where $\mathbf{R}_t(\tau) = \nabla_{\mathbf{W}_t} L(\mathbf{F}_q(\tau),\mathbf{F}_v(\tau))$ for $t\in\{q,v\}$. In particular, we have

$$\begin{aligned}L_\tau{}'(\mathbf{F}_q(0),\mathbf{F}_v(0)) &= -\sum_{t\in\{q,v\}} tr\Big(\mathbf{G}_t^\top(\mathbf{G}_t\mathbf{W}_t^\top - \mathbf{W}_t\mathbf{G}_t^\top)\mathbf{W}_t\Big)\\ &= -\frac{1}{2}\|\mathbf{P}_q\|_F^2 - \frac{1}{2}\|\mathbf{P}_v\|_F^2\end{aligned}. \quad (14)$$

Please refer to [29] for the theoretical proof details of curvilinear search algorithm.

3.6 CCL Algorithm

After the optimization of $\mathbf{W}_q$ and $\mathbf{W}_v$, we can obtain the linear mapping functions defined in Eq.(1). With this, original incomparable textual query and visual image become comparable. Specifically, given a test query-image pair $(\hat{\mathbf{q}}\in\mathbb{R}^{d_q},\hat{\mathbf{v}}\in\mathbb{R}^{d_v})$, we compute the distance value between the pair as

$$r(\hat{\mathbf{q}},\hat{\mathbf{v}}) = \|\hat{\mathbf{q}}\mathbf{W}_q - \hat{\mathbf{v}}\mathbf{W}_v\|_2. \quad (15)$$

This value reflects how relevant the query could be used to describe the given image, with lower numbers indicating higher relevance. For any query, sorting by its corresponding values for all its associated images gives the retrieval ranking for these images. The algorithm is given in Algorithm 1.

Algorithm 1 Click-through-based Cross-view Learning (CCL)

Input: $0<\mu<1$, $0<\rho_1<\rho_2<1$, $\varepsilon\geq 0$, and initial $\mathbf{W}_q$ and $\mathbf{W}_q$.
for *iter* $= 1$ to T_{max} **do**
  compute gradients $\mathbf{G}_q$ and $\mathbf{G}_v$ via Eq.(8).
  if $\|\mathbf{G}_q\|_F^2 + \|\mathbf{G}_v\|_F^2 \leq \varepsilon$ **then**
    exit.
  end if
  compute $\mathbf{P}_q$ and $\mathbf{P}_v$ by using Eq.(9).
  compute $L_\tau{}'(\mathbf{F}_q(0),\mathbf{F}_v(0))$ according to Eq.(14).
  set $\tau = 1$.
  repeat
    $\tau = \mu\tau$
    compute $\mathbf{F}_q(\tau)$ and $\mathbf{F}_v(\tau)$ via Eq.(10).
    compute $L_\tau{}'(\mathbf{F}_q(\tau),\mathbf{F}_v(\tau))$ via Eq.(13).
  until Armijo-Wolfe conditions in Eq.(12) are satisfied
  update the transformation matrices:
    $\mathbf{W}_q = \mathbf{F}_q(\tau)$
    $\mathbf{W}_v = \mathbf{F}_v(\tau)$
end for
Output:
distance function: $\forall\hat{\mathbf{q}},\hat{\mathbf{v}},\ \ r(\hat{\mathbf{q}},\hat{\mathbf{v}}) = \|\hat{\mathbf{q}}\mathbf{W}_q - \hat{\mathbf{v}}\mathbf{W}_v\|_2$.

3.7 Complexity Analysis

The time complexity of CCL mainly depends on the computation of $\mathbf{G}_q$, $\mathbf{G}_v$, $\mathbf{P}_q$, $\mathbf{P}_v$, $\mathbf{F}_q(\tau)$, $\mathbf{F}_v(\tau)$, and $L_\tau{}'(\mathbf{F}_q(\tau),\mathbf{F}_v(\tau))$. Obviously, the computation complexity of $\mathbf{G}_q$ and $\mathbf{G}_v$ is $\mathcal{O}(n^2\times d_q)$ and $\mathcal{O}(n^2\times d_v)$, respectively. $\mathbf{P}_q$ and $\mathbf{P}_v$ take $\mathcal{O}(d_q^2\times d)$ and $\mathcal{O}(d_v^2\times d)$.

The matrix inverse $(\mathbf{I}+\frac{\tau}{2}\mathbf{P}_q)^{-1}$ and $(\mathbf{I}+\frac{\tau}{2}\mathbf{P}_v)^{-1}$ dominate the computation of $\mathbf{F}_q(\tau)$ and $\mathbf{F}_v(\tau)$ in Eq.(10). By forming $\mathbf{P}_q$ and $\mathbf{P}_v$ as the outer product of two low-rank matrices, the inverse computation cost decreases a lot. As defined in Eq.(9), $\mathbf{P}_q = \mathbf{G}_q\mathbf{W}_q^\top - \mathbf{W}_q\mathbf{G}_q^\top$ and $\mathbf{P}_v = \mathbf{G}_v\mathbf{W}_v^\top - \mathbf{W}_v\mathbf{G}_v^\top$, $\mathbf{P}_q$ and $\mathbf{P}_v$ can be equivalently rewritten as $\mathbf{P}_q = \mathbf{X}_q\mathbf{Y}_q^\top$ and $\mathbf{P}_v = \mathbf{X}_v\mathbf{Y}_v^\top$, where $\mathbf{X}_q = [\mathbf{G}_q,\mathbf{W}_q], \mathbf{Y}_q = [\mathbf{W}_q,-\mathbf{G}_q]$ and $\mathbf{X}_v = [\mathbf{G}_v,\mathbf{W}_v], \mathbf{Y}_v = [\mathbf{W}_v,-\mathbf{G}_v]$. According to Sherman-Morrison-Woodbury formula, i.e.,

$$(\mathbf{A}+\alpha\mathbf{X}\mathbf{Y}^\top)^{-1} = \mathbf{A}^{-1} - \alpha\mathbf{A}^{-1}\mathbf{X}(\mathbf{I}+\alpha\mathbf{Y}^\top\mathbf{A}^{-1}\mathbf{X})^{-1}\mathbf{Y}^\top\mathbf{A}^{-1},$$

the matrix inverse $(\mathbf{I}+\frac{\tau}{2}\mathbf{P}_q)^{-1}$ can be re-expressed as

$$(\mathbf{I}+\frac{\tau}{2}\mathbf{P}_q)^{-1} = \mathbf{I} - \frac{\tau}{2}\mathbf{X}_q(\mathbf{I}+\frac{\tau}{2}\mathbf{Y}_q^\top\mathbf{X}_q)^{-1}\mathbf{Y}_q^\top.$$

Furthermore, $\mathbf{F}_q(\tau)$ can be rewritten as

$$\mathbf{F}_q(\tau) = \mathbf{W}_q - \tau\mathbf{X}_q(\mathbf{I}+\frac{\tau}{2}\mathbf{Y}_q^\top\mathbf{X}_q)^{-1}\mathbf{Y}_q^\top\mathbf{W}_q.$$

For $\mathbf{F}_v(\tau)$, we can get the corresponding conclusion. Since we typically have $d \ll d_q$, the cost of inverting $(\mathbf{I}+\frac{\tau}{2}\mathbf{Y}_q^\top\mathbf{X}_q)\in\mathbb{R}^{2d\times 2d}$ is much lower than inverting $(\mathbf{I}+\frac{\tau}{2}\mathbf{P}_q)\in\mathbb{R}^{d_q\times d_q}$. The inverse of $(\mathbf{I}+\frac{\tau}{2}\mathbf{Y}_q^\top\mathbf{X}_q)^{-1}$ takes $\mathcal{O}(d^3)$, thus

the computation complexity of $\mathbf{F}_q(\tau)$ is $\mathcal{O}(d_q d^2) + \mathcal{O}(d^3)$. Similarly, $\mathbf{F}_v(\tau)$ is $\mathcal{O}(d_v d^2) + \mathcal{O}(d^3)$. The computation of $L_\tau'(\mathbf{F}_q(\tau), \mathbf{F}_v(\tau))$ has a cost of $\mathcal{O}(n^2 \times d_q) + \mathcal{O}(n^2 \times d_v) + \mathcal{O}(d_q d^2) + \mathcal{O}(d_v d^2) + \mathcal{O}(d^3)$.

As $d \ll d_q, d_v \ll n$, the overall complexity of the Algorithm 1 is $T_{max} \times T \times \mathcal{O}(n^2 \times max(d_q, d_v))$, where T is the number of searching for appropriate τ which satisfies the Armijo-Wolfe conditions and it is usually less than ten in our experiments. Take the training of $\mathbf{W}_q$ and $\mathbf{W}_v$ on one million {query, image, click} triads with $d_v = 1,024$ and $d_q = 10,000$ for example, our algorithm takes about 32 hours on a server with 2.40GHz CPU and 128GB RAM.

3.8 Extensions

Although we only present the distance function between query and image on the learned mapping matrices in the Algorithm 1, the optimization actually can also help learning of query-query and image-image distance. Similar to the distance function between query and image, the distance between query and query, image and image, is computed as $(\forall \hat{\mathbf{q}}, \bar{\mathbf{q}},\ \ r(\hat{\mathbf{q}}, \bar{\mathbf{q}}) = \|\hat{\mathbf{q}}\mathbf{W}_q - \bar{\mathbf{q}}\mathbf{W}_q\|_2)$ and $(\forall \hat{\mathbf{v}}, \bar{\mathbf{v}},\ \ r(\hat{\mathbf{v}}, \bar{\mathbf{v}}) = \|\hat{\mathbf{v}}\mathbf{W}_v - \bar{\mathbf{v}}\mathbf{W}_v\|_2)$, respectively. Furthermore, the obtained distance can be applied for several IR applications, e.g., query suggestion, query expansion, image clustering, image classification, and so on.

4. EXPERIMENTS

We conducted our experiments on the Clickture dataset [10] and evaluated our approaches for image search.

4.1 Dataset

The dataset, Clickture, is a large-scale click based image dataset [10]. It was collected from one year click-through data of one commercial image search engine. The dataset comprises two parts, i.e., the training and development (dev) sets. The training set consists of 23.1 million $\{query, image, click\}$ triads, where *query* is a textual word or phrase, *image* is a base64 encoded JPEG image thumbnail, and *click* is an integer which is no less than one. There are 11.7 millions distinct queries and 1.0 million unique images of the training set. Figure 2 shows a few exemplary images with their clicked queries and click counts in the Clickture. For example, users clicked the first image 146 times in the search results when submitting query "obama" in total. It is worth noting that there is no surrounding text or description of images provided in the Clickture.

In the dev dataset, there are 79,926 $\langle query, image \rangle$ pairs generated from 1,000 queries, where each image to the corresponding query was manually annotated on a three point ordinal scale: Excellent, Good, and Bad. In the experiments, the training set is used for learning the latent subspace, while the dev set is used for performance evaluation.

4.2 Experimental Settings

Task. We investigate whether our proposed approach can be used to improve image search in this work. Specifically, we use Clickture as "labeled" data for semantic queries and train the ranking model. The task is to estimate the relevance of the image and the query for each test query-image pair, and then for each query, we order the images based on the prediction scores returned by our trained ranking model.

Textual and Visual Features. We take the word in queries as "word features." Words are stemmed and stop words are removed. With word features, each query is represented by a tf vector in the query space. In our experiments, we use the top 10,000 most frequent words as the word vocabulary. Inspired by the success of deep neural networks (DNN) [4], we use it to generate image representation in this work, which is a 1024-dimensional feature vector. Specifically, similar to [15], the used DNN architecture is denoted as $Image - C64 - P - N - C128 - P - N - C192 - C192 - C128 - P - F4096 - F1024 - F1000$, which contains five convolutional layers (denoted by C following the number of filters) while the last three are fully-connected layers (denoted by F following the number of neurons); the max-pooling layers (denoted by P) follow the first, second and fifth convolutional layers; local contrast normalization layers (denoted by N) follow the first and second max-pooling layers. The weights of DNN are learned on ILSVRC-2010[1], which is a subset of ImageNet[2] dataset with 1.26 million training images from 1,000 categories. For an image, its representation is the neuronal responses of the layer $F1024$ by input the image into the learned DNN.

Figure 2: Examples in Clickture dataset (upper row: clicked images; lower row: search query with click times on the upper image).

Compared Approaches. We compare the following approaches for performance evaluation:

- N-Gram SVM Modeling (N-Gram SVM). We use all the clicked images of a given query as positive samples and randomly select negative samples from the rest of the training dataset to build a support vector machine (SVM) model for each query, and then use this model to predict the relevance of the query to a new image. In addition, in order to extend the capability of the training data to model queries that are not covered in the dataset, n-gram modeling, which attempts to model each n-gram as a "query," is used. In other words,

[1] http://www.image-net.org/challenges/LSVRC/2010/
[2] http://www.image-net.org/

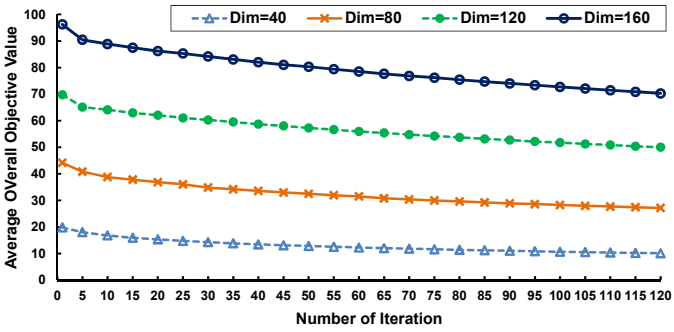

Figure 3: The average overall objective value of Eq. (6) for each query-image pair with the increase of the iteration. The changes of the value are given at different dimensionality of the latent subspace.

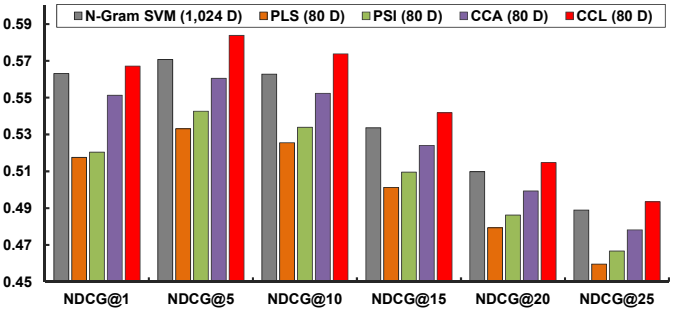

Figure 4: The NDCG of different approaches for image search. The numbers in the brackets represent the feature dimension used in each approach.

if a query is not in the training set, but its n-grams appear in some queries of the training set, we can generate the model by linearly fusing the SVM models of these queries. No latent subspace is learned in this baseline. We name this run as *N-Gram SVM*.

- Canonical Correlation Analysis [8][9] (CCA). A classical and successful approach for mapping visual and textual features into a latent subspace where the correlation between the two views is maximized. This run is named as *CCA*.
- Partial Least Squares [25][30] (PLS). Similar to CCA, PLS aims to learn linear mapping functions to project two views into a common latent subspace as well. But different from CCA, PLS learns dot product as the similarity function while cosine similarity is used in CCA. Deriving from the ideas in [30], the learning of the mappings is performed by maximizing the similarities of the observed query-image pairs on the click-through data here. We name this run as *PLS*.
- Polynomial Semantic Indexing [2][32] (PSI). Similar in spirit, PSI first chooses a low dimensional feature representation space for query and image, and then a polynomial model is discriminatively learned for mapping the query-image pair to a relevance score. This run is named as *PSI*.
- Click-through-based Cross-view Learning (CCL). We designed the run, *CCL*, for our proposed approach described in Algorithm 1.

Parameter Settings. *N-Gram SVM* is a baseline without low-dimensional latent subspace learning, thus the relevance score is predicted on the original visual features. For the other four subspace learning methods, the dimensionality of the latent subspace is in the range of $\{40, 80, 120, 160\}$. The k nearest neighbors preserved in Eq.(4) is chosen within $\{100, 500, 1000, 1500, 2000\}$. The tradeoff parameter λ in the overall objective function is set within $\{0.1, 0.2, ..., 1.0\}$. We set μ=0.3, ρ_1=0.2, and ρ_2=0.9 in the curvilinear search by using a validation set.

Evaluation Metrics. For the evaluation of image search, we adopted Normalized Discounted Cumulative Gain ($NDCG$) which takes into account the measure of multi-level relevancy as the performance metric. Given an image ranked list, the $NDCG$ score at the depth of d in the ranked list is defined by:

$$NDCG@d = Z_d \sum_{j=1}^{d} \frac{2^{r^j} - 1}{\log(1+j)} \quad (16)$$

where $r^j = \{Excellent = 3, Good = 2, Bad = 0\}$ is the manually judged relevance for each image with respect to the query. Z_d is a normalizer factor to make the score for d Excellent results 1. The final metric is the average of $NDCG@d$ for all queries in the test set.

4.3 Optimization Analysis

As we choose the step τ satisfying the Armijo-Wolfe conditions to achieve an approximate minimizer of $L(\mathbf{F}_q(\tau), \mathbf{F}_v(\tau))$ in Algorithm 1 instead of finding the global minimization due to its computationally expense, we depict the average overall objective value of Eq.(6) for one query-image pair versus iterations to illustrate the convergence of the algorithm. As shown in Figure 3, the value does decrease as the iterations increase at all the dimensionality of the latent subspace. Specifically, after 100 iterations, the average objective value between query mapping and image projection is around 10 when the latent subspace dimension is 40. Thus, the experiment verifies that our algorithm can always reach a reasonable local optimum.

4.4 Performance Comparison

Figure 4 shows the NDCG performances on image search of five runs averaged over 1,000 queries in Clickture dev dataset. It is worth noting that the prediction of *N-Gram SVM* is performed on the original image visual features of 1,024 dimensions and for other four methods, the performances are given by choosing 80 as the dimensionality of the latent subspace.

Overall, our proposed *CCL* consistently outperforms the other runs across different depths of NDCG. In particular, the NDCG@10 of *CCL* can achieve 0.5738, which makes the improvement over *N-Gram SVM* model by 4.0%. More importantly, by learning a low-dimensional latent subspace, the dimension of the mappings of textual query and visual image is reduced by several orders of magnitude. Furthermore, *CCL* by additionally incorporating structure preservation leads to a performance boost against *PLS* and *CCA*. The result basically indicates the advantage of minimizing distance between views in the latent subspace and preserving similarity in the original space simultaneously.

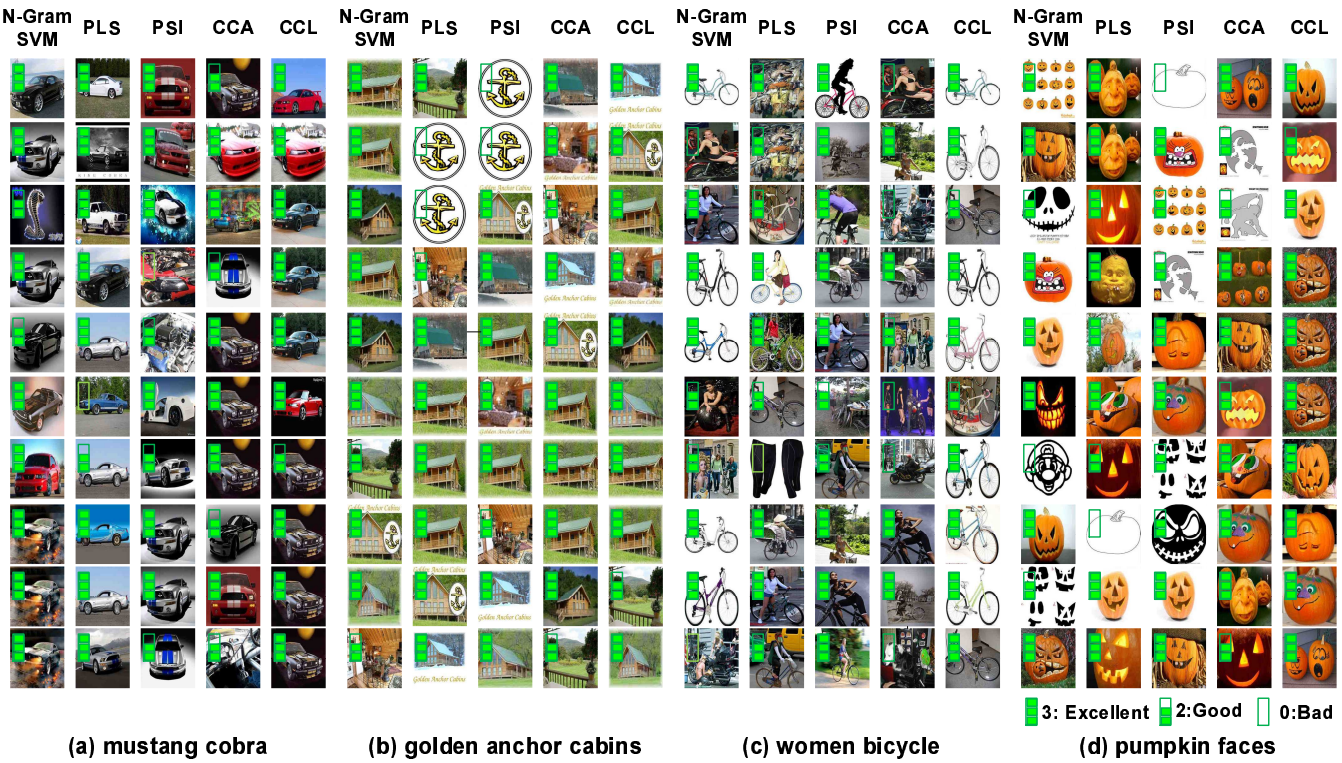

Figure 5: Examples showing the top 10 image search results by different methods of queries "mustang cobra," "golden anchor cabins," "women bicycle," and "pumpkin faces" (better viewed in color). The relevance scale is provided at the top left corner for each image.

There is a performance gap between *CCA* and *PLS*. Though both runs attempt to learn linear mapping functions for forming a subspace, they are different in the way that *CCA* learns cosine as a similarity function, and *PLS* learns dot product instead. As indicated by our results, maximizing the correlation between the mappings in the latent subspace can lead to a better performance. Moreover, *PSI* utilizing click-through data as relative relevance judgements rather than absolute click numbers is superior to *PLS*, but is still lower than *CCL*. Another observation is that the performance gain is almost consistent when going deeper into the list. This further confirms the effectiveness of *CCL*.

Figure 5 shows the top 10 image search results by different approaches for the query "mustang cobra," "golden anchor cabins," "women bicycle," and "pumpkin faces." We can easily see the proposed *CCL* method gets the most satisfying ranking results. Specifically, compared to other baselines, the top images by *CCL* are more visually similar to each other, especially of the query "women bicycle" and "pumpkin faces." That is mainly caused by the effect of structure preservation regularization term in the overall objective, which restricts the similar images in the original space to remain close in the low-dimensional latent subspace. Therefore, the ranks of these group of images are likely to be moved up.

4.5 Effect of the Dimensionality of the Latent Subspace

In order to show the relationship between the performance and the dimensionality of the latent subspace, we compared the results of the dimension in the range of 40, 80, 120, and 160. As the method *N-Gram SVM* performs training and prediction by only using the original features rather than learning a latent subspace, it is excluded in this comparison.

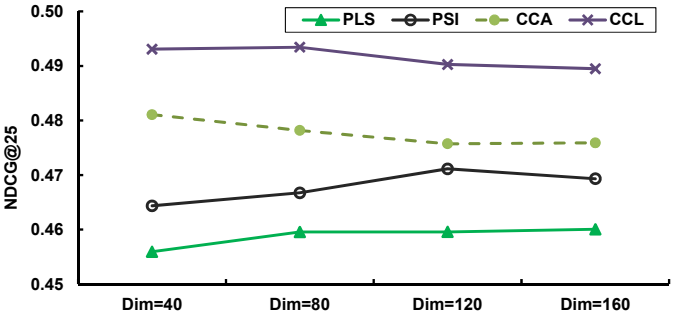

Figure 6: The NDCG@25 performance with different dimensionalities of the latent subspace. We can see that *CCL* achieves the best performance among the four methods.

The results are shown in Figure 6. Compared to the other three runs, performance improvement is consistently observed at each dimensionality of the latent subspace by our proposed *CCL* method. Furthermore, *CCL* achieves the best result at the latent subspace dimensionality of 80, and the results at other dimensionality are pretty close to the best one. This observation basically verifies that *CCL* has a good

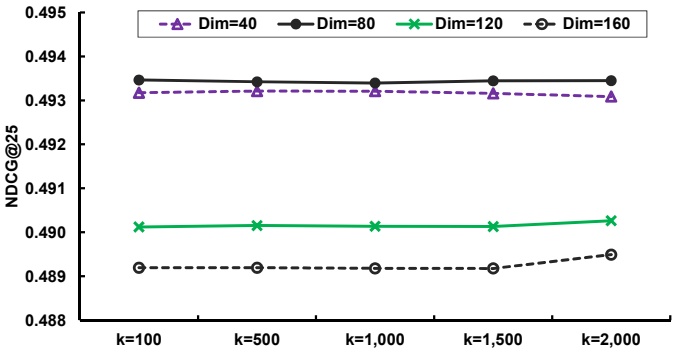

Figure 7: The NDCG@25 performance curve at different dimensionalities of the latent subspace with different numbers of nearest neighbors.

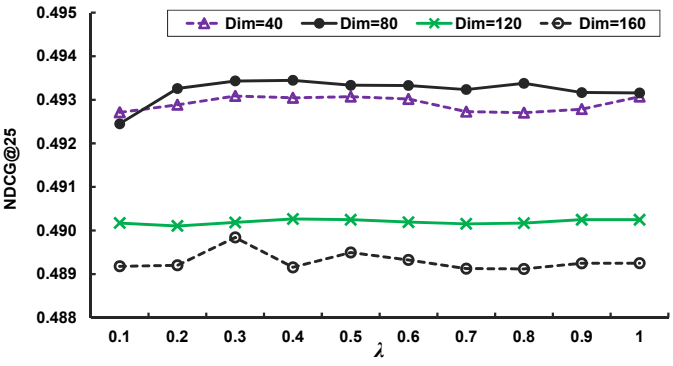

Figure 8: The NDCG@25 performance curve at different dimensionalities of the latent subspace with different λ.

property of being affected very slightly with the change of the dimensionality of the latent subspace.

Another important observation is that when the dimensionality of the latent subspace increases, the performances of all the methods are not always improved accordingly. For example, the best performance of CCL happens at the dimensionality of 80 and for the method CCA, the highest NDCG@25 is observed at the dimensionality of 40. This somewhat indicates a general conclusion that the selection of the latent subspace dimensionality is related to the optimized objective considered in learning the subspace.

4.6 Effect of the Number of Nearest Neighbors

The number of nearest neighbors considered in the structure preservation is another parameter in CCL. In the previous experiments, the number was fixed to 2,000. Next, we conducted experiments to evaluate the performance of our CCL method with the number of nearest neighbors in range of $\{100, 500, 1000, 1500, 2000\}$ at different dimensionality of the latent subspace.

The NDCG@25 with the different number of nearest neighbors are shown in Figure 7. As illustrated in the figure, the optimal k differs at different dimensionality of the latent subspace. However, at each dimensionality of the latent subspace, the performance difference by using different number of nearest neighbors is within 0.0002, which softens the difficulty on choosing the optimal number of nearest neighbors in practice.

4.7 Effect of the Parameter λ

A common problem with multiple regularization terms in a joint optimization objective is the need to set the parameters to tradeoff each component. In the previous experiments, the tradeoff λ is optimally set in order to examine the performance of CCL on image search irrespective of the parameter influence. We further conducted experiments to test the sensitivity of λ towards search performance.

Figure 8 shows the NDCG@25 performance with respect to different values of λ at different dimensionality of the latent subspace. Similar to the effect of the number of nearest neighbors, we can see that the performance curve is very smooth when λ varies in a range from $\{0.1, 0.2, ..., 1.0\}$ at each dimension of the latent subspace. Specifically, when the dimension of the latent subspace is 80, the performance fluctuates within the range of 0.001. Thus, the performance is not sensitive to the change of the tradeoff parameter.

5. DISCUSSION AND CONCLUSION

In this paper, we have investigated the issue of directly learning the multi-view distance between a textual query and an image by leveraging both click data and subspace learning techniques. The click data represent the click relations between queries and images, while the subspace learning aims to learn a latent common subspace between multiple views. We have proposed a novel click-through-based cross-view learning to solve the problem in a principle way. Specifically, we use two different linear mappings to project textual queries and visual images into a latent subspace. The mappings are learned by jointly minimizing the distance of the observed query-image pairs on the click-through bipartite graph and preserving the inherent structure in original single view. Moreover, we make orthogonal assumptions on the mapping matrices. Then the mappings can be obtained efficiently through curvilinear search. We take l_2 norm between the projections of query and image in the latent subspace as the distance function to measure the relevance of a pair of (query, image).

Our future works are as follows. First, the two learned mapping matrices can be extended to the learning of query-query and image-image distances. Next, the learned distances will be further explored for applications such as query expansion, query suggestion, and image clustering, in the learned low-dimensional space. Furthermore, we will investigate the kernel version of our method, making it applicable when kernel matrices instead of features are available.

6. ACKNOWLEDGMENTS

This work was partially supported by the National Natural Science Foundation of China (No. 61390514, No. 61272290), the Fundamental Research Funds for the Central Universities (No. WK2100060011), and the Shenzhen Research Institute, City University of Hong Kong.

7. REFERENCES

[1] R. A. Baeza-Yates and A. Tiberi. Extracting semantic relations from query logs. In *Proceedings of ACM Conference on Knowledge Discovery and Data Mining*, 2007.

[2] B. Bai, J. Weston, D. Grangier, R. Collobert, K. Sadamasa, Y. Qi, C. Cortes, and M. Mohri. Polynomial semantic indexing. In *Proceedings of Advances in Neural Information Processing Systems*, 2009.

[3] D. Beeferman and A. L. Berger. Agglomerative clustering of a search engine query log. In *Proceedings of ACM Conference on Knowledge Discovery and Data Mining*, 2000.

[4] C. F. Cadieu, H. Hong, D. Yamins, N. Pinto, N. J. Majaj, and J. J. DiCarlo. The neural representation benchmark and its evaluation on brain and machine. In *Proceedings of International Conference on Learning Representations*, 2013.

[5] C. Cortes, M. Mohri, and A. Rostamizadeh. Two-stage learning kernel algorithms. In *Proceedings of International Conference on Machine Learning*, 2010.

[6] N. Craswell and M. Szummer. Random walks on the click graph. In *Proceedings of ACM Conference on Research and Development in Information Retrieval*, 2007.

[7] Z. Fang and Z. Zhang. Discriminative feature selection for multi-view cross-domain learning. In *Proceedings of ACM Conference of Information and Knowledge Management*, 2013.

[8] 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):210–233, 2014.

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

[10] X.-S. Hua, L. Yang, J. Wang, J. Wang, M. Ye, K. Wang, Y. Rui, and J. Li. Clickage: Towards bridging semantic and intent gaps via mining click logs of search engines. *Proceedings of ACM International Conference on Multimedia*, 2013.

[11] V. Jain and M. Varma. Learning to re-rank: Query-dependent image re-ranking using click data. In *Proceedings of International World Wide Web Conference*, 2011.

[12] T. Joachims. Optimizing search engines using clickthrough data. In *Proceedings of ACM Conference on Knowledge Discovery and Data Mining*, 2002.

[13] T. Joachims, L. Granka, B. Pan, H. Hembrooke, F. Radlinski, and G. Gay. Evaluating the accuracy of implicit feedback from clicks and query reformulations in web search. *ACM Trans. on Information Systems*, 25(2), 2007.

[14] M. Kloft, U. Brefeld, S. Sonnenburg, P. Laskov, K.-R. Muller, and A. Zien. Evaluating search engines by modeling the relationship between relevance and clicks. In *Efficient and accurate l_p-norm multiple kernel learning*, 2009.

[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proceedings of Advances in Neural Information Processing Systems*, 2012.

[16] A. Kumar, P. Rai, and H. Daume. Co-regularized multi-view spectral clustering. In *Proceedings of Advances in Neural Information Processing Systems*, 2011.

[17] G. R. G. Lanckriet, N. Cristianini, P. L. Bartlett, L. E. Ghaoui, and M. I. Jordan. Learning the kernel matrix with semidefinite programming. *Journal of Machine Learning Research*, 5:27–72, 2004.

[18] X. Li, Y.-Y. Wang, and A. Acero. Learning query intent from regularized click graphs. In *Proceedings of ACM Conference on Research and Development in Information Retrieval*, 2008.

[19] Q. Mei, D. Zhou, and K. W. Church. Query suggestion using hitting time. In *Proceedings of ACM Conference of Information and Knowledge Management*, 2008.

[20] T. Mei, Y. Rui, S. Li, and Q. Tian. Multimedia Search Reranking: A Literature Survey. *ACM Computing Surveys*, 46(3), Sept. 2014.

[21] S. Melacci and M. Belkin. Laplacian support vector machines trained in the primal. *Journal of Machine Learning Research*, 12:1149–1184, 2011.

[22] I. Muslea, S. Minton, and C. Knoblock. Active learning with multiple views. *Journal of Artificial Intelligence Research*, 27(1):203–233, 2006.

[23] Y. Pan, T. Yao, K. Yang, H. Li, C.-W. Ngo, J. Wang, and T. Mei. Image search by graph-based label propagation with image representation from dnn. *Proceedings of ACM International Conference on Multimedia*, 2013.

[24] B. Poblete and R. A. Baeza-Yates. Query-sets: using implicit feedback and query patterns to organize web documents. In *Proceedings of International World Wide Web Conference*, 2008.

[25] R. Rosipal and N. Krämer. Overview and recent advances in partial least squares. *Subspace, Latent Structure and Feature Selection*, pages 34–51, 2006.

[26] W. Sun and Y.-X. Yuan. *Optimization theory and methods: nonlinear programming*, volume 98. springer, 2006.

[27] M. Trevisiol, L. Chiarandini, L. M. Aiello, and A. Jaimes. Image ranking based on user browsing behavior. In *Proceedings of ACM Conference on Research and Development in Information Retrieval*, 2012.

[28] J.-R. Wen, J.-Y. Nie, and H. Zhang. Clustering user queries of a search engine. In *Proceedings of International World Wide Web Conference*, 2001.

[29] Z. Wen and W. Yin. A feasible method for optimization with orthogonality constrains. *Mathematical Programming*, 142:397–434, 2013.

[30] W. Wu, H. Li, and J. Xu. Learning query and document similarities from click-through bipartite graph with metadata. *Proceedings of ACM Conference on Web Search and Data Mining*, 2013.

[31] C. Xu, D. Tao, and C. Xu. A survey on multi-view learning. *CoRR abs/1304.5634*, 2013.

[32] T. Yao, T. Mei, C.-W. Ngo, and S. Li. Annotation for free: Video tagging by mining user search behavior. *Proceedings of ACM International Conference on Multimedia*, 2013.

[33] S. Yu, B. Krishnapuram, R. Rosales, and R. Rao. Bayesian co-training. *Journal of Machine Learning Research*, pages 2649–2680, 2011.