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 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
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 $W_q$ and $W_v$, 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 Befferman 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 technique 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 \( G = (V, E) \) denote a click-through bipartite. \( V = Q \cup V \) is the set of vertices, which consists of a query set \( Q \) and
an image set \( V \). \( E \) is the set of edges between the query
and image vertices. The number associated with the edge
represents the click times in the image search results of the
query. Suppose there are \( n \) trips \( \{ q_i, v_i, c_i \}^n_{i=1} \) generated
from the click-through bipartite in total, where \( c_i \) is the
the click counts of image \( v_i \) in response to query \( q_i \). Let \( Q =
\{ q_1, q_2, \ldots, q_n \} \in \mathbb{R}^{n \times d_q} \) and \( V = \{ v_1, v_2, \ldots, v_n \} \in
\mathbb{R}^{n \times d_v} \) denote the query and image feature matrix, where
\( q_i \) and \( v_i \) 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 \( C \) 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 ex-
ists for the representation of query and image. The linear
mapping function can be derived from this subspace by
\[
    f(q_i) = q_i W_q, \quad f(v_i) = v_i W_v,
\]
where \( d \) is the dimensionality of the common subspace, and
\( W_q \in \mathbb{R}^{d_q \times d} \) and \( W_v \in \mathbb{R}^{d_v \times d} \) are the transfor-
mation 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
\[
\begin{align*}
    \min_{W_q, W_v} & \mathcal{L}(W_q, W_v) \\
    \text{s.t.} & \ W_q^\top W_q = I, \ W_v^\top W_v = I
\end{align*}
\]
(2)

where \( \mathcal{L}(\cdot) \) denotes the trace function. The matrices \( W_q \)
and \( W_v \) have orthogonal columns, i.e., \( W_q^\top W_q = I, W_v^\top W_v = I \),
where \( I \) is an identity matrix. The constraints restrict \( W_q \)
and \( W_v \) to converge to reasonable solutions rather than go
to \( 0 \) 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 be-
tween 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
\( W_q \) and \( W_v \) 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 po-
tions closely in the shared latent subspace. The estima-
tion of the underlying structure can be measured by the ap-
propriate pairwise similarity between the training samples.
Specifically, it can be given by
\[
\sum_{i,j=1}^n S_{ij}^q ||q_i W_q - q_j W_q||^2 + \sum_{i,j=1}^n S_{ij}^v ||v_i W_v - v_j W_v||^2,
\]
(3)
where \( S^q \in \mathbb{R}^{n \times n} \) and \( S^v \in \mathbb{R}^{n \times n} \) denote the affinity matrices
defined on the queries and images, respectively. Under
the structure preservation criterion, it is reasonable to mini-
mize 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 \( S^q \)
and \( S^v \). Inspired by [7], the elements are computed by Gaussian
functions in this work, i.e.,
\[
S_{ij}^q = \begin{cases} 
    e^{-\frac{||v_i - v_j||^2}{\sigma_i^2}} & \text{if } t_i \in N_k(t_j) \text{ or } t_j \in N_k(t_i) \\
    0 & \text{otherwise}
\end{cases}
\]
(4)

By minimizing this term, the similarity between ex-
amples 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 be-
tween views in Eq.(2) and structure preservation in Eq.(5).
Hence we get the following optimization problem
\[
\begin{align*}
    \min_{W_q, W_v} & \mathcal{L}(W_q, W_v) \\
    \text{s.t.} & \ W_q^\top W_q = I, \ W_v^\top W_v = I
\end{align*}
\]
where \( \mathcal{L}(\cdot) \) is the objective function in Eq.(6). Thus, the optimization problem can be
rewritten as
\[
\begin{align*}
    \min_{W_q, W_v} & \mathcal{L}(W_q, W_v) \\
    \text{s.t.} & \ W_q^\top W_q = I, \ W_v^\top W_v = I
\end{align*}
\]
(6)

\[
\begin{align*}
    \min_{W_q, W_v} & \mathcal{L}(W_q, W_v) \\
    \text{s.t.} & \ W_q^\top W_q = I, \ W_v^\top W_v = I
\end{align*}
\]
(7)

The optimization above is a non-convex problem. Never-
theless, the gradient of the objective function with respect to
\( W_q \) and \( W_v \) can be easily obtained as follows:
\[
\begin{align*}
\nabla_{W_q} L(W_q, W_v) &= 2Q^\top C(W_q - \nabla_{W_q} L) + 2Q^\top L^\top(QW_q) \\
\nabla_{W_v} L(W_q, W_v) &= 2Q^\top C(W_v - \nabla_{W_v} L) + 2Q^\top L^\top(W_q)
\end{align*}
\]
(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 \( \{W_q, W_v\} \) and their corresponding gradients \( \{G_q = \nabla_{W_q} L(W_q, W_v), G_v = \nabla_{W_v} L(W_q, W_v)\} \), we define the skew-symmetric matrices \( P_q \) and \( P_v \) as

\[
P_q = G_q W_q^\top - W_q G_q^\top, \quad P_v = G_v W_v^\top - W_v G_v^\top.
\]

The new point can be searched as a curvilinear function of a step size \( \tau \), such that

\[
F_q(\tau) = (I + \frac{\tau}{2} P_q)^{-1}(I - \frac{\tau}{2} P_q) W_q,
\]

\[
F_v(\tau) = (I + \frac{\tau}{2} P_v)^{-1}(I - \frac{\tau}{2} P_v) W_v.
\]

Then, it is easy to verify that \( F_q(\tau) \) and \( F_v(\tau) \) lead to several characteristics. The matrices \( F_q(\tau) \) and \( F_v(\tau) \) satisfy

\[
(F_q(\tau))^\top F_q(\tau) = (F_v(\tau))^\top F_v(\tau) = I
\]

for all \( \tau \in R \). The derivatives with respect to \( \tau \) are given as

\[
F_q'(\tau) = -(I + \frac{\tau}{2} P_q)^{-1} P_q \left( W_q + \frac{\tau}{2} F_q(\tau) \right),
\]

\[
F_v'(\tau) = -(I + \frac{\tau}{2} P_v)^{-1} P_v \left( W_v + \frac{\tau}{2} F_v(\tau) \right).
\]

In particular, we can obtain \( F_q'(0) = -P_q W_q \) and \( F_v'(0) = -P_v W_v \). Then, \( (F_q(\tau), F_v(\tau))_{\tau \geq 0} \) is a descent curve. We use the classical Armijo-Wolfe based monotone curvilinear search algorithm [26] to determine a suitable step \( \tau \) as one satisfying the following conditions:

\[
L(F_q(\tau), F_v(\tau)) \leq L(F_q(0), F_v(0)) + \rho_1 \tau L'(F_q(\tau), F_v(\tau)),
\]

\[
L'(F_q(\tau), F_v(\tau)) \geq \rho_2 L'(F_q(0), F_v(0)),
\]

where \( \rho_1 \) and \( \rho_2 \) are two parameters satisfying \( 0 < \rho_1 < \rho_2 < 1 \). \( L'(F_q(\tau), F_v(\tau)) \) is the derivative of \( L \) with respect to \( \tau \) and is calculated by

\[
L'(F_q(0), F_v(0)) = -\sum_{t \in \{q, v\}} \text{tr} \left( R_t(\tau)^\top (I + \frac{\tau}{2} P_t)^{-1} R_t \left( \frac{W_t + F_t(\tau)}{2} \right) \right),
\]

where \( R_t(\tau) = \nabla_{W_t} L(F_q(\tau), F_v(\tau)) \) for \( t \in \{q, v\} \). In particular, we have

\[
L'(F_q(0), F_v(0)) = -\sum_{t \in \{q, v\}} \text{tr} \left( G_t^\top (G_t W_t^\top - W_t G_t^\top) W_t \right) = -\frac{1}{2} \| P_q \|^2_F + \frac{1}{2} \| P_v \|^2_F.
\]

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

3.6 CCL Algorithm

After the optimization of \( W_q \) and \( W_v \), 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 \( (\hat{q}, \hat{v}) \in \mathbb{R}^{d_q} \times \mathbb{R}^{d_v} \), we compute the distance value between the pair as

\[
\tau(\hat{q}, \hat{v}) = \| \hat{q} W_q - \hat{v} W_v \|_2.
\]

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)

1: Input: \( 0 < \mu < 1, 0 < \rho_1 < \rho_2 < 1, \epsilon \geq 0 \), and initial \( W_q \) and \( W_v \).

2: for \( \text{iter} = 1 \) to \( T_{\text{max}} \) do

3: compute gradients \( G_q \) and \( G_v \) via Eq.(8).

4: if \( \|G_q\|_F^2 + \|G_v\|_F^2 \leq \epsilon \) then

5: exit.

6: end if

7: compute \( P_q \) and \( P_v \) by using Eq.(9).

8: compute \( L'(F_q(0), F_v(0)) \) according to Eq.(14).

9: set \( \tau = 1 \).

10: repeat

11: \( \tau = \mu \tau \)

12: compute \( F_q(\tau) \) and \( F_v(\tau) \) via Eq.(10).

13: compute \( L'(F_q(\tau), F_v(\tau)) \) via Eq.(13).

14: until Armijo-Wolfe conditions in Eq.(12) are satisfied

15: update the transformation matrices:

\[
W_q = F_q(\tau),
\]

\[
W_v = F_v(\tau).
\]

16: end for

17: Output:

distance function: \( \forall \hat{q}, \hat{v}, \tau(\hat{q}, \hat{v}) = \| \hat{q} W_q - \hat{v} W_v \|_2 \).

The time complexity of CCL mainly depends on the computation of \( G_q, G_v, P_q, P_v, F_q(\tau), F_v(\tau), \) and \( L'(F_q(\tau), F_v(\tau)) \). Obviously, the computation complexity of \( G_q \) and \( G_v \) is \( O(n^2 \times d_q) \) and \( O(n^2 \times d_v) \), respectively. \( P_q \) and \( P_v \) take \( O(d_q^2 \times d) \) and \( O(d_v^2 \times d) \).

The matrix inverse \( (I + \frac{\tau}{2} P_q)^{-1} \) and \( (I + \frac{\tau}{2} P_v)^{-1} \) dominate the computation of \( F_q(\tau) \) and \( F_v(\tau) \) in Eq.(10). By forming \( P_q \) and \( P_v \) as the outer product of two low-rank matrices, the inverse computation cost decreases a lot. As defined in Eq.(9), \( P_q = G_q W_q^\top - W_q G_q^\top \) and \( P_v = G_v W_v^\top - W_v G_v^\top \). \( P_q \) and \( P_v \) can be equivalently rewritten as \( P_q = X_q Y_q^\top \) and \( P_v = X_v Y_v^\top \), where \( X_q = [G_q, W_q] \), \( Y_q = [W_q, -G_q] \) and \( X_v = [G_v, W_v] \), \( Y_v = [W_v, -G_v] \). According to Sherman-Morrison-Woodbury formula, i.e.,

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

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

\[
(I + \frac{\tau}{2} P_q)^{-1} = I - \frac{\tau}{2} X_q (I + \frac{\tau}{2} Y_q X_q)^{-1} Y_q.
\]

Furthermore, \( F_q(\tau) \) can be rewritten as

\[
F_q(\tau) = W_q - \tau X_q (I + \frac{\tau}{2} Y_q X_q)^{-1} Y_q W_q.
\]

For \( F_v(\tau) \), we can get the corresponding conclusion. Since we typically have \( d \ll d_q \), the cost of inverting \( (I + \frac{\tau}{2} Y_q X_q) \in \mathbb{R}^{d_q \times d_q} \) is much lower than inverting \( (I + \frac{\tau}{2} P_q) \in \mathbb{R}^{d_q \times d_q} \). The inverse of \( (I + \frac{\tau}{2} Y_q X_q)^{-1} \) takes \( O(d^3) \), thus
the computation complexity of $F_q(\tau)$ is $O(d_q d_q^2) + O(d_q^3)$. Similarly, $F_q(\tau)$ is $O(d_q d_q^2) + O(d_q^3)$. The computation of $L^\tau(F_q(\tau), F_q(\tau))$ has a cost of $O(n^2 \times d_q) + O(n^2 \times d_q) + O(d_q d_q^2) + O(d_q^3) + O(d_q^3)$. 

As $d \ll d_q, d_e \ll n$, the overall complexity of the Algorithm 1 is $T_{max} \times T \times O(n^2 \times \max(d_q, d_e))$, where $T$ is the number of searching for appropriate $\tau$ which satisfies the Armijo-Wolfe conditions and it is usually less than ten in our experiments. Take the training of the Armijo-Wolfe conditions and it is usually less than ten 

Similarly, 

data of one commercial image search engine. The dataset [10]. It was collected from one year click-through 

4. EXPERIMENTS 

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 

$\langle \hat{v}q, q \rangle = \| \hat{q}W_q - \hat{q}W_q \rangle \| \rangle$ and $\langle \hat{q}v, \hat{q}v \rangle = \| \hat{v}W_q - \hat{v}W_q \|$, 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 {query, image} 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 $-\text{C64} - \text{P} - \text{N} - \text{C128} - \text{P} - \text{N} - \text{N} - \text{C192} - \text{C192} - \text{C128} - \text{P} - \text{F4096} - \text{F1024} - \text{F1000}$, 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 $L$) follow the first and second max-pooling layers. The weights of DNN are learned on ILSVRC-20101, which is a subset of ImageNet2 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.

Comparative 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,

\[ \text{Image} - \text{C64} - \text{P} - \text{N} - \text{C128} - \text{P} - \text{N} - \text{N} - \text{C192} - \text{C192} - \text{C128} - \text{P} - \text{F4096} - \text{F1024} - \text{F1000}, \]

\[ \text{N-Gram SVM} \]


Figure 2: Examples in Clickture dataset (upper row: clicked images; lower row: search query with click times on the upper image).
different dimensionality of the latent subspace. The changes of the value are given at different dimensionality of the latent subspace.

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 map-
ping 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 relev-
ance score is predicted on the original visual features. For
the other four subspace learning methods, the dimensionali-
ty 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 \(\lambda\) in
the overall objective function is set within \((0.1, 0.2, ..., 1.0)\).
We set \(\gamma=0.3\), \(\rho_1=0.2\), and \(\rho_2=0.9\) 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 relevan-
cy as the performance metric. Given an image ranked list,

the NDCG score at the depth of \(d\) in the ranked list is
deﬁned by:
\[
NDCG@d = \frac{Z_d \sum_{j=1}^{d} \frac{2^r_j - 1}{\log(1 + j)}}
\]

where \(r_j = \{\text{Excellent} = 3, \text{Good} = 2, \text{Bad} = 0\}\) 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 \(\tau\) satisfying the Armijo-Wolfe condi-
tions to achieve an approximate minimizer of \(L(F_q(\tau), F_v(\tau))\) in
Algorithm 1 instead of ﬁnding the global minimization
due to its computationally expensive, we depict the average
overall objective value of Eq.(6) for one query-image pair
versus iterations to illustrate the convergence of the algo-
rithm. As shown in Figure 3, the value does decrease as
the iterations increase at all the dimensionality of the lat-
ent subspace. Speciﬁcally, after 100 iterations, the average
objective value between query mapping and image projec-
tion 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
two 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 perfor-
mances 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. Further-
more, CCL by additionally incorporating structure preser-
vation 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 preserv-
similarity in the original space simultaneously.
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 C-CA 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.” 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.

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

<table>
<thead>
<tr>
<th>N-Gram</th>
<th>PLS</th>
<th>PSI</th>
<th>CCA</th>
<th>CCL</th>
<th>N-Gram</th>
<th>PLS</th>
<th>PSI</th>
<th>CCA</th>
<th>CCL</th>
<th>N-Gram</th>
<th>PLS</th>
<th>PSI</th>
<th>CCA</th>
<th>CCL</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SVM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SVM</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

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.

![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.](image-url)
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 \(\lambda\)

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 \(\lambda\) 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 \(\lambda\) towards search performance.

Figure 8 shows the NDCG@25 performance with respect to different values of \(\lambda\) 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 \(\lambda\) 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


