Author(s): Cui, Chaoran; Shen, Jialie; Chen, Zhumin; Wang, Shuaiqiang; Ma, Jun

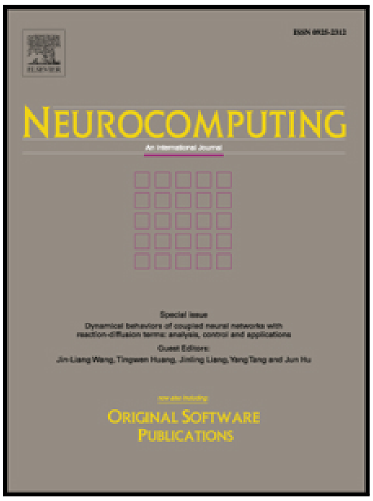
Title: Learning to Rank Images for Complex Queries in Concept-based Search

Year: 2018

Version:

Please cite the original version:

Cui, C., Shen, J., Chen, Z., Wang, S., & Ma, J. (2018). Learning to Rank Images for Complex Queries in Concept-based Search. Neurocomputing, 274, 19-28. https://doi.org/10.1016/j.neucom.2016.05.118


ISSN 0925-2312
ELSEVIER
NEUROCOMPUTING
An International Journal
Special issue
Dynamical behaviors of coupled neural networks with reaction-diffusion terms: analysis, control and applications
Guest Editors:
Jin-Liang Wang, Tingwen Huang, Jinling Liang, Yang Tang and Jun Hu
now also including
ORIGINAL SOFTWARE PUBLICATIONS

Learning to Rank Images for Complex Queries in Concept-based Search

Chaoran Cui, Jialie Shen, Zhumin Chen, Shuaiqiang Wang, Jun Ma

PII: S0925-2312(17)30665-3
DOI: 10.1016/j.neucom.2016.05.118
Reference: NEUCOM 18328

To appear in: *Neurocomputing*

Received date: 17 February 2016
Revised date: 6 April 2016
Accepted date: 9 May 2016

Please cite this article as: Chaoran Cui, Jialie Shen, Zhumin Chen, Shuaiqiang Wang, Jun Ma, Learning to Rank Images for Complex Queries in Concept-based Search, *Neurocomputing* (2017), doi: 10.1016/j.neucom.2016.05.118

Learning to Rank Images for Complex Queries in Concept-based Search

Chaoran Cui[a,b], Jialie Shen[b], Zhumin Chen[c], Shuaiqiang Wang[d], Jun Ma[c]

[a]*School of Computer Science and Technology, Shandong University of Finance and Economics, Jinan 250014, China*
[b]*School of Information Systems, Singapore Management University, Singapore 178902, Singapore*
[c]*School of Computer Science and Technology, Shandong University, Jinan 250101, China*
[d]*Department of Computer Science and Information Systems, Jyväskylä University, Agora 40014, Finland*

Abstract

Concept-based image search is an emerging search paradigm that utilizes a set of concepts as intermediate semantic descriptors of images to bridge the semantic gap. Typically, a user query is rather complex and cannot be well described using a single concept. However, it is less effective to tackle such complex queries by simply aggregating the individual search results for the constituent concepts. In this paper, we propose to introduce the learning to rank techniques to concept-based image search for complex queries. With freely available social tagged images, we first build concept detectors by jointly leveraging the heterogeneous visual features. Then, to formulate the image relevance, we explicitly model the individual weight of each constituent concept in a complex query. The dependence among constituent concepts, as well as the relatedness between query and non-query concepts, are also considered through modeling the pairwise concept correlations in a factorization way. Finally, we train our model to directly optimize the image ranking performance for complex queries under a pairwise learning to rank framework. Extensive experiments on two benchmark datasets well verified the promise of our approach.

Keywords: Concept-based Image Search, Complex Query, Learning to Rank, Factorization Machine

1. Introduction

With rapid advances in Internet and multimedia technologies, the past few years have witnessed an explosive growth of digital images on the Web. The proliferation of images raises an urgent demand for effective image search technologies. Due to the well-known semantic gap between low-level features and high-level semantics [1, 2], current commercial search engines retrieve images mainly based on their associated contextual information such as titles and surrounding text on Web pages. However, since the associated text is usually unreliable to describe the semantic content of images, the performance of text-based image search methods is still far from satisfactory.

As an alternative to text-based image search, concept-based image search has recently attracted increasing attention and proven to be a promising solution for large-scale search tasks [3, 4, 5]. In concept-based image search, a set of concept detectors are pre-built to predict the presence of specific concepts, which provide direct access to the semantic content of images. Given a textual query, it is mapped to a group of primitive concepts, and the search results are made up of the images in which these concepts are likely to appear. Thanks to the continuous progress in visual concept detection [6, 7], current concept-based search techniques can effectively deal with queries involving only one concept. In reality, however, a user query is rather complex and cannot be well represented by a single concept. For example, consider a query like “a person with a camera on the street”, which apparently involves multiple semantic concepts, i.e., “person”, “camera”, and “street”.

Confronted with a complex query comprising several semantic concepts, a natural idea is to combine the individual search results for the constituent concepts in the query. However, such a straightforward strategy may be ineffective due to the following reasons. First of all, many existing methods assume all constituent concepts are of equal importance [8] or determine their combination weights based on some heuristic rules [9]. From the perspective of information theory, the importance of a constituent concept can be interpreted as the information it bears when the complex query is observed [10]. Different constituent concepts typically exhibit different degrees of informativeness, which are data-dependent and difficult to determine in advance. Secondly, the constituent concepts in a complex query do not appear in isolation; instead, they interact with each other in the semantic level and mutually reinforce their roles during the search process. It is inappropriate to consider the constitute concepts independently and ignore their inter-dependence [3]. Lastly, the concepts not in a complex query may also serve as the contextual information to enhance the search accuracy [11]. Recall the aforementioned query example, i.e., “a person with a camera on the street”. If an image has a high response for the detector of a non-query concept “sofa”, we may have high confidence that the image is irrelevant to the query, since “sofa” rarely appears together with the query concept “street”. Nevertheless, the information cues conveyed by the non-query

Email addresses: bruincui@gmail.com (Chaoran Cui), jlshen@smu.edu.sg (Jialie Shen), chenzhumin@sdu.edu.cn (Zhumin Chen), shuaiqiang.wang@jyu.fi (Shuaiqiang Wang), majun@sdu.edu.cn (Jun Ma)

concepts have not been fully exploited in prior concept-based image search methods.

Recently, learning to rank techniques [12] have been extensively studied owing to its potential for improving information retrieval systems. In general, learning to rank refers to applying supervised machine learning algorithms to construct the optimal ranking model in a search task. Intuitively, through the supervision step, the possibility is offered that utilizing the information from the data collection to steer the search process and reduce the need for making heuristic assumptions [13]. Although great success has been achieved [14, 15], few research efforts have been devoted to exploring the potential of learning to rank in concept-based image search.

Motivated by the above discussions, in this paper, we propose to introduce the learning to ranking techniques to concept-based image search for complex queries. A collection of concept detectors are first built from social tagged images by jointly leveraging the heterogeneous visual features. To mitigate the limitations of existing methods mentioned above, in the formulation of the image relevance function, we explicitly model the individual weight of each constituent concept in a complex query. The dependence among constituent concepts, as well as the relatedness between query and non-query concepts, are also considered by modeling the pairwise concept correlations. Faced with the underlying overfitting problem arising from too many model parameters, we adopt the Factorization Machine [16] to factorize concept correlations with a low-rank approximation. The learning of different model parameters is effectively integrated into a pairwise learning to rank framework, and we build upon the Ranking SVM algorithm [17] to train our model by directly optimizing the image ranking performance for complex queries. It is worth noting that the scalability of our approach is not degraded, even though the supervision step is introduced. This is because the ground-truth information used in training is only for a limited number of complex queries, but from which a query-independent model can be learned and employed to rank images for all queries.

The main contributions can be summarized as follows:

- Our approach resolves the problem of concept-based image search from the perspective of learning to rank, and directly optimizes the image ranking performance for complex queries.
- Our approach explicitly models the individual weight of each constituent concept. To capture the dependence among constituent concepts, as well as the relatedness between query and non-query concepts, the pairwise concept correlations are also modeled in a factorization way.
- Our approach has been evaluated on two publicly accessible benchmark datasets. The experimental results demonstrate the promise of our approach in comparison with the state-of-the-art methods.

The remainder of this paper is structured as follows. Section 2 reviews the related work. Section 3 details our proposed approach to concept-based image search for complex queries. Experimental results and analysis are reported in Section 4, followed by the conclusion and future work in Section 5.

2. Related Work

2.1. Visual Concept Detection

Serving as the foundation for concept-based image search, visual concept detection has attracted considerable research interests in the multimedia computing community. Typically, it is transformed to a classification problem, in which each concept is treated as a class label and its presence likelihood is estimated by the classifier prediction score. For example, Lu et al. [18] proposed an multi-modality classifier combination framework to improve the accuracy of semantic concept detection. Multiple classifiers trained on different visual features were combined with a probability-based fusion method. Some studies provided insights on how to construct feature representations in building classifiers for concept detection. In [19], an efficient bag-of-visual-word construction method was developed based on sparse non-negative matrix factorization and GPU enabled SIFT feature extraction. Li et al. [20] employed latent Dirichlet allocation approach to cluster the image data into semantic topics, and the distributions of image low-level features on such topics were used as the middle-level features of images. Yan et al. [21] proposed to automatically select semantic meaningful concepts for the event detection task based on both the events-kit text descriptions and the concept high-level feature descriptions. A novel event oriented dictionary representation was then introduced based on the selected semantic concepts. Besides, the zero-shot learning has also been applied to handle event detection in videos [22, 23]. The key idea is to pre-train a number of concept classifiers using data from other sources, such that an event of interest can be detected based on its semantic correlation with respect to each concept, even when no labeled example of this event is supplied.

2.2. Concept-based Image Search

Given a collection of concept detectors, concept-based image search for complex queries can be performed by fusing the individual search results for the constituent concepts in a query. A critical issue in the fusion strategy is to determine the combination weights. Nastsev et al. [24] proposed to assign equal weight to the search result for each constituent concept. Chang et al. [25] weighted the individual concept detectors according to their training performance. Li et al. [26] set the weight to be proportional to the informativeness of a constituent concept. Despite encouraging results reported, these heuristic fusion methods are data-independent and may not be effective to the same degree in different application scenarios. On the contrary, in our approach, the individual weight of each constituent concept is explicitly modeled and automatically determined with the information harvested from the data collection.

Another potential limitation of the above fusion-based methods lies in that they consider the constituent concepts independently and ignore their mutual relationships. To address this issue, Yuan et al. [4] leveraged the plentiful but partially related samples, as well as the users' feedbacks, to handle complex queries in the interactive concept-based video search. By extending this idea, they further proposed a higher-level semantic descriptor named "concept bundle", which integrates multiple primitive concepts, to describe the visual representation of complex semantics and enhance the video search for complex queries [27]. Li et al. [10] learned bi-concept detectors from social tagged images, and applied them in a search engine for retrieving images relevant to bi-concept queries. In [3], the authors developed an image reranking scheme for complex queries by jointly considering multiple relationships between concepts and complex queries from high-level to low-level. Similarly, Guo et al. [5] proposed a multi-layer probabilistic model to incorporate inter-concept relatedness into image reranking for complex queries. Compared to the previous work, our approach models the pairwise concept correlations in a factorization manner. Through this way, we consider not only the dependence among constituent concepts, but also the relatedness between query and non-query concepts.

2.3. Learning to Rank

There is an emerging research interest in learning to rank due to its importance in a wide variety of applications, such as information retrieval [15] and personalized recommendation [28]. Roughly speaking, the existing learning to rank techniques can be divided into three categories: pointwise methods, pairwise methods, and listwise methods. In pointwise methods [29], ranking is treated as a regression or classification problem on individual items to predict their relevance scores. In pairwise methods [30], ranking is transformed to a classification problem on item pairs to predict the preference relation between two items. In listwise methods [31], ranking is performed to minimize a direct loss between the true ranking list and the estimated ranking list. A comprehensive survey of learning to rank can be found in [12]. In this paper, our approach follows the direction of pairwise methods, because of their superior performance and relatively low complexity.

3. Framework

To formulate our problem, we declare some notations in advance. In particular, we use capital letters (e.g., X) and bold lowercase letters (e.g., $\mathbf{x}$) to denote sets and vectors, respectively. We employ non-bold lowercase letters (e.g., x) to represent scalars, and Greek letters (e.g., λ) as hyper-parameters. If not clarified, all vectors are in column form. Table 1 summarizes the key notations and definitions used throughout the paper.

Table 1: Summary of key notations and definitions.

Notation	Definition
C	The set of all concepts
c, q, p	A certain concept
T	The set of all possible complex queries
Q	A certain complex query
x	A certain image
L	The set of all labeled images
x_i, Y_i	A certain labeled image and the set of concepts associated with it
m	The number of concepts
n	The number of labeled images
d	The dimensionality of the latent space for concepts
w_c	The weight of c
$\mathbf{v}_c$	The vector representation of c in the latent space
s_{qc}	The correlation between q and c
S	The set of social tagged images
S_c	The subset of images tagged with c
Z	The set of visual features
z	A certain visual feature
k	The number of visual neighbors
$S_{x,z}$	The neighbor set of x based on z
D	The set of preference pairs
l	The sample size in each iteration during training
$\alpha, \beta, \lambda, \gamma$	The hyper-parameters

Our framework consists of three main components: 1) visual concept detection, 2) image relevance formulation, and 3) ranking-oriented learning. By harnessing freely available social tagged images, visual concept detectors are first built without the need of manually selecting training samples for each concept. With the pre-built concept detectors, an image relevance function for complex queries is then formulated, which explicitly takes into account concept weights and concept correlations. Based on the relevance formulation, the ranking-oriented learning is ultimately developed to determine the model parameters through optimizing the image ranking performance for complex queries. The architecture of our framework is illustrated in Figure 1. In the following, we elaborate on each of the components and give a full description of the associated algorithms.

3.1. Visual Concept Detection

As a prerequisite to realize concept-based image search, various concept detectors need to be built in advance to predict the presence likelihood of the corresponding semantic concepts given a specific image. An appealing source of labeled images for concept detection are social tagged images on the Web [10], in which user-contributed tags encode valuable information about the semantic content of images. As mentioned previously, a typical solution is to train a separate classifier for each concept over social tagged images, and estimate the presence of that concept by the classifier prediction score. However, this concept-specific modeling paradigm suffers from two main disadvantages. First, it is not scalable to cover the potentially

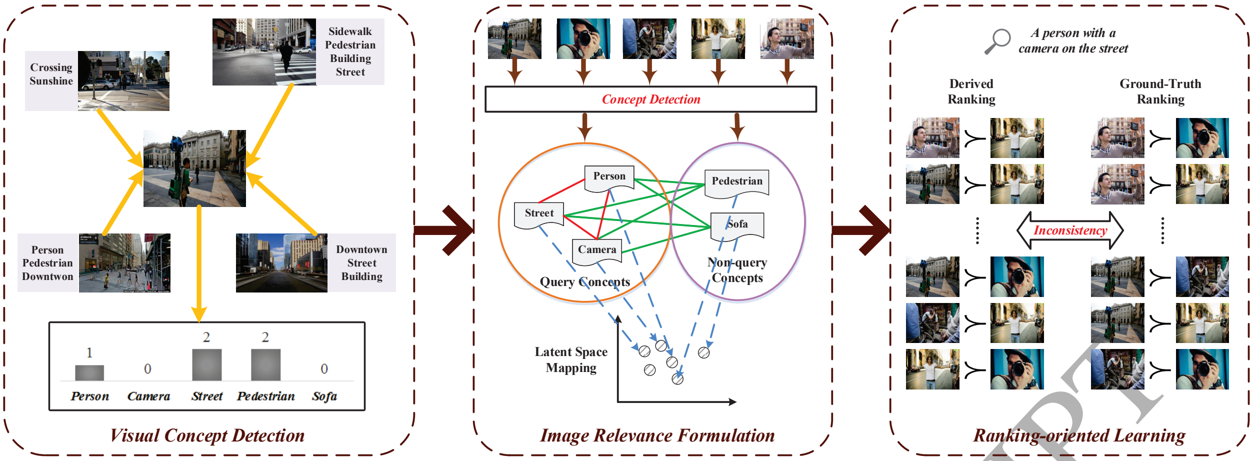


Figure 1: Schematic illustration of the proposed concept-based image search approach for complex queries.

unlimited array of concepts in existence. Second, how to select high quality training examples from social tagged images at large scale is still an open research problem.

To avoid the above problems, we adopt a data-driven approach, called the neighbor voting algorithm [32], for concept detection in this paper. The philosophy behind the neighbor voting algorithm is that if visually similar images are tagged with the same concepts, these concepts are likely to reflect the actual visual content. Despite its simplicity, recent studies [33] have reported that the neighbor voting algorithm remains the state-of-the-art for visual concept detection. In addition, a semantic concept generally has significant diversity in terms of the visual appearance. It is hence insufficient to rely on a single visual feature to characterize such large visual variations. In light of this, we seek to jointly leverage the heterogeneous visual features for building more robust concept detectors.

Let S be a collection of social tagged images, and C a vocabulary consisting of m concepts. For each concept $c \in C$, S_c denotes the subset of images tagged with c, i.e., $S_c \subset S$. We use x to denote an image, and Z is a set of visual features. Given a visual feature $z \in Z$, we represent x using z and find the k nearest neighbors of x from S according to the visual similarity measured over z. $S_{x,z}$ denotes the resulting neighbor set of x, based on which the neighbor voting algorithm constructs a base concept detector as follows:

$$g_z(c,x) = \frac{\left|S_{x,z} \cap S_c\right|}{k} - \frac{\left|S_c\right|}{|S|} , \tag{1}$$

where $|\cdot|$ is the cardinality of a set. Intuitively, the more frequent a concept occurs in the neighbor set, the more relevant it might be to the given image; however, common concepts with high frequency in the entire collection are usually less descriptive, and thus their estimated relevance should be suppressed. Towards this end, the base concept detector $g_z(c,x)$ counts the difference between the distribution of c in x's neighbor set and that in the entire image collection.

To overcome the limitation of single features in describing the visual content, we further combine the base concept detectors obtained with different visual features. The work in [34] compared the unsupervised and supervised combination strategies in the context of neighbor voting model, and the empirical results showed that there is no significant difference in performance between them. However, a major disadvantage of the supervised combination strategy is its expensiveness in terms of the training efforts, which inevitably leads to much more computational cost. In light of this, we adopt the unsupervised uniform combination rule in our approach. Specifically, the concept detector is defined as follows:

$$r(c,x) = \frac{1}{|Z|} \sum_{z \in Z} g_z(c,x) , \tag{2}$$

where $r(c,x)$ indicates the confidence that the concept c is present in the image x.

3.2. Image Relevance Formulation

In this paper, we target at the problem of concept-based image search for complex queries. Let Q be a complex query comprising two or more concepts, i.e., $Q \subset C$ and $|Q| \geq 2$. The key challenge is to establish a function $f(Q,x)$ that measures the relevance score of an image x with respect to the complex query Q. Intuitively, each constituent concept $q \in Q$ partially describes the user's search intentions carried by Q, and $f(Q,x)$ can thus be estimated by aggregating the presence likelihood of each q in x. Inspired by this, we first formulate $f(Q,x)$ as follows:

$$f(Q,x) = \sum_{q \in Q} w_q r(q,x) , \tag{3}$$

where w_q is a weight parameter indicating the importance of q. Distinguished from previous methods combining different constituent concepts heuristically, we explicitly model the individual weight of each constituent concept.

Unlike a single-concept query, a complex query also depicts the intrinsic semantic dependence among its constituent concepts [27]. Different constituent concepts do not appear in isolation; instead, they interact with each other and mutually reinforce their roles in the search process for the complex query. In addition, as aforementioned, the concepts not in the query often provide additional information cues. Hence, it is highly beneficial to retrieve images by simultaneously using both query concepts and non-query ones. In view of this, we explore the possibility of introducing concept correlations to the image relevance function.

WordNet similarity is widely adopted to capture the semantic correlations among concepts. Nonetheless, as it does not directly reflect how people describe the visual content, some highly correlated concepts are usually weakly related in the WordNet ontology [35]. Concept co-occurrence is another commonly used correlation measurement. However, in most annotated corpus, images are frequently associated with only a few concept labels, which may lead to unreliable co-occurrence statistics. More importantly, apart from the positive correlations among concepts, there also exist many important negative correlations. Unfortunately, limited by their non-negative property, both WordNet similarity and co-occurrence statistics cannot reflect the potential negative correlations.

Given the drawbacks of existing correlation measurements, we propose to model the pairwise correlations between concepts, and extend our initial relevance function in Eq. (3) as follows:

$$f(Q,x) = \sum_{q\in Q} w_q r(q,x) + \frac{\alpha}{2}\sum_{q\in Q}\sum_{p\in Q\backslash q} s_{qp} r(q,x) r(p,x) \qquad (4)$$
$$+\beta \sum_{q\in Q}\sum_{c\in C\backslash Q} s_{qc} r(q,x) r(c,x) \ ,$$

where s_{qp} is a model parameter capturing the correlation between two concepts q and p. We assume that the concept correlations are symmetric, i.e., $s_{qp} = s_{pq}$, and both positive and negative values are allowed. In Eq. (4), the first term represents the relevance estimated by separately considering each constituent concept in the complex query, the second term encodes the interactions among constituent concepts, and the last term ensures that the information from non-query concepts can also be utilized. The three parts cooperate with each other, leading to a more accurate estimation for the image relevance. Here, α and β are two hyper-parameters used to control the relative contribution of each term.

A potential problem in the above formulation is that it requires a huge amount of parameters to model the correlation between each pair of concepts in the vocabulary. From the viewpoint of statistical learning theory, too many model parameters may degrade the model stability and result in the overfitting problem. The existing work [36] on text information processing has demonstrated that the semantic space spanned by textual keywords can be approximated by a smaller set of *latent factors*. As one kind of text information, semantic concepts are also subject to such a low-rank property [37]. Inspired by this, we apply the Factorization Machine [16] to model the pairwise concept correlations in a factorization way. Specifically, each concept $c \in C$ is mapped to a vector $\mathbf{v}_c \in \mathbb{R}^d$ in a d-dimensional latent space, and the correlation s_{qp} is subsequently approximated by $s_{qp} = \mathbf{v}_q^T \mathbf{v}_p$. Intuitively, s_{qp} corresponds to the dot product of $\mathbf{v}_q$ and $\mathbf{v}_p$ in the latent space, which is a commonly used measure for matching textual vectors. As a result, the image relevance function can be reformulated as follows:

$$f(Q,x) = \sum_{q\in Q} w_q r(q,x) + \frac{\alpha}{2}\sum_{q\in Q}\sum_{p\in Q\backslash q} \mathbf{v}_q^T \mathbf{v}_p r(q,x) r(p,x) \qquad (5)$$
$$+\beta \sum_{q\in Q}\sum_{c\in C\backslash Q} \mathbf{v}_q^T \mathbf{v}_c r(q,x) r(c,x) \ .$$

Because the intrinsic dimensionality of the latent space is typically much smaller than the total number of concepts (i.e., $d \ll m$), the number of model parameters in Eq. (5) is significantly reduced. Besides, it has been shown that the problems of concept synonymy and polysemy can be more easily handled in a low-dimensional semantic space.

3.3. Ranking-oriented Learning

We aim to enhance the accuracy of concept-based image search for complex queries by learning the relevance function f in a supervised way. In the supervised scenario, we are given a set of labeled images $L = \{x_1, x_2, \ldots, x_n\}$, where each image x_i is associated with Y_i that denotes the set of concepts having been assigned to x_i. Let T be a set of complex queries. Given a complex query $Q \in T$, the ground-truth relevance of x_i with respect to Q is defined as:

$$rel(Q, x_i) = |Q \cap Y_i| \ . \qquad (6)$$

Eq. (6) ensures that the images associated with more query concepts will be assigned higher relevance. Based on the ground-truth relevance, a set of pairwise preference relations $D \subseteq T \times L \times L$ can be further derived:

$$D = \left\{(Q, x_i, x_j) \mid rel(Q, x_i) > rel(Q, x_j)\right\} \ , \qquad (7)$$

where each triple (Q, x_i, x_j) reflects the partial order information of the ground-truth image ranking for Q. To optimize the image ranking performance for complex queries, we require the relevance function f to satisfy the preference pairs in D as much as possible. In other words, the goal of learning is to minimize the following empirical risk:

$$R(f) = \frac{1}{|D|}\sum_{(Q,x_i,x_j)\in D} \mathbb{1}\left(f(Q,x_i) \le f(Q,x_j)\right) \ , \qquad (8)$$

where $\mathbb{1}(\cdot)$ is an indicator function that outputs 1 if the input Boolean expression is true and zero otherwise. Actually, $R(f)$ measures the proportion of the preference pairs misordered by the relevance function f.

Since the indicator function $\mathbb{1}(\cdot)$ is nonsmooth, directly optimizing the empirical risk in Eq. (8) is computationally infeasible [14]. To address the problem, we adopt the Ranking SVM framework [17] as the backbone of our learning method. The basic idea of Ranking SVM is to replace $\mathbb{1}\left(f(Q,x_i) \le f(Q,x_j)\right)$ with the hinge loss function $\left[1 - \left(f(Q,x_i) - f(Q,x_j)\right)\right]_+$. As a result, the relevance function f can be learned through the following optimization problem:

Algorithm 1 The Pegasos Algorithm

```
Input: Set of preference pairs $D$, sample size $l$, and learning rate $\gamma$
Output: Model parameters $\theta = (w, \mathbf{v})$
for $c \in C$ do
    Initialize $w_c$ and $\mathbf{v}_c$ randomly
end for
repeat
    Sample a subset $D_s$ of $l$ training triples from $D$
    Compute $\nabla_\theta \Omega$ based on Eq. (10)
    Update $\theta = \theta - \gamma \nabla_\theta \Omega$
until convergence
```

Optimization Problem 1.

$$\begin{aligned} &\underset{w,\mathbf{v},\xi}{min.} \quad \frac{\lambda_1}{2}\sum_{c\in C} w_c^2 + \frac{\lambda_2}{2}\sum_{c\in C}\|\mathbf{v}_c\|_2^2 + \frac{1}{|D|}\sum_{(Q,x_i,x_j)\in D}\xi_{Q,x_i,x_j} \qquad (9)\\ &s.t. \qquad \forall (Q,x_i,x_j)\in D:\\ &f(Q,x_i) \geq f(Q,x_j) + 1 - \xi_{Q,x_i,x_j}\ , \quad \xi_{Q,x_i,x_j} \geq 0\ . \end{aligned}$$

Here, ξ_{Q,x_i,x_j} is a slack variable associated with the triple (Q, x_i, x_j). It can be demonstrated that the average over all slack variables is an upper bound on the empirical risk in Eq. (8). λ_1 and λ_2 are the hyper-parameters representing the weights of the regularization terms.

The main difficulty of Optimization Problem 1 lies in that there are too many (i.e., $|D|$) constraints to be considered. To solve it efficiently, we resort to the Pegasos algorithm [38] to optimize the primal form of the problem. At each iteration of the Pegasos algorithm, a subset D_s of l training triples is first sampled from D uniformly at random. Then, the subgradients with respect to the model parameters involved with the triples in D_s are computed. Specifically, we use θ to denote an arbitrary model parameter, and the subgradient of the objective function Ω regarding θ can be computed by:

$$\nabla_\theta \Omega = \lambda\theta - \frac{1}{l}\sum_{(Q,x_i,x_j)\in D_s}\nabla_\theta \xi_{Q,x_i,x_j}\ , \qquad (10)$$

$$\begin{aligned} \nabla_\theta \xi_{Q,x_i,x_j} = {} & \mathbb{1}\Big(f(Q,x_i) - f(Q,x_j) < 1\Big) \qquad (11)\\ & \times \Big(\nabla_\theta f(Q,x_i) - \nabla_\theta f(Q,x_j)\Big)\ , \end{aligned}$$

where $\nabla_\theta f(Q,x)$ is the subgradient of the relevance function f with respect to θ, which is calculated by:

$$\begin{cases} 1)\ r(q,x) & \text{if } \theta = w_q,\ q \in Q;\\ 2)\ \alpha\sum_{p\in Q\backslash q}\mathbf{v}_p r(q,x)r(p,x) & \text{if } \theta = \mathbf{v}_q,\ q\in Q;\\ +\,\beta\sum_{c\in C\backslash Q}\mathbf{v}_c r(q,x)r(c,x) & \\ 3)\ 0 & \text{if } \theta = w_c,\ c\in C\backslash Q;\\ 4)\ \beta\sum_{q\in Q}\mathbf{v}_q r(q,x)r(c,x) & \text{if } \theta = \mathbf{v}_c,\ c\in C\backslash Q. \end{cases} \qquad (12)$$

Lastly, θ is updated in the opposite direction of $\nabla_\theta\Omega$ with a learning rate γ. The pseudo-code of the Pegasos algorithm is presented in Algorithm 1.

Once the model parameters are learned, given a new complex query, the relevance score of a specific image with respect to the query can be estimated by Eq. (5). Based on this score, we obtain the image ranking results for the complex query.

Table 2: Statistics of the experimental datasets.

	Dataset I	Dataset II
# of images	25,000	55,615
# of concepts	18	81
Avg. # of concepts per image	2.03	4.21
# of complex queries	121	488

4. Experiments

In this section, we report a series of experiments conducted to evaluate our approach in the scenario of concept-based image search for complex queries.

4.1. Datasets

To ensure the accuracy and fairness of our empirical results, we adopted two benchmark image datasets collected from Flickr[1] in our evaluation. Dataset I is MIRFlickr [39], which consists of 25,000 images. In this dataset, the ground-truth labeling for 18 concepts has been provided, and the average number of concepts per image is 2.03. Note that these concepts all correspond to frequent tags in Flickr and cover different genres including scenes, objects, and events. Dataset II is NUS-WIDE-LITE [40], which contains 55,615 images with their associated tags. Likewise, the ground-truth annotations of 81 concepts for all images are available in the dataset. Each image is annotated with an average of 4.21 concepts.

Since there are no pre-defined complex queries available, we need to first construct the query set. Following the procedures in [41], we created a complex query by randomly combining the given concepts in the dataset. As reported in [42], Web queries are generally short, and the average number of terms per query is 2.4. According to the recent statistics[2] from the US, only less than 6.2% of the queries have more than 5 terms. Therefore, the length of a complex query was set to be between 2 and 5 concepts in our experiments. Besides, we only kept the complex queries for which more than 1% of all the images are annotated with their constituent concepts. The preceding steps finally led to 121 complex queries for Dataset I and 488 for Dataset II, respectively. On both datasets, we took half of the complex queries for training, and used the rest for testing. The main statistics of the datasets are summarized in Table 2.

[1] http://www.flickr.com/

[2] http://www.keyworddiscovery.com/keyword-stats.html/

4.2. Experimental Settings

To implement the concept detectors described in Section 3.1, we used five types of low-level visual features to represent each image, namely, 1) 64-dimensional color histogram, 2) 144-dimensional color correlogram, 3) 73-dimensional edge direction histogram, 4) 128-dimensional wavelet texture, and 5) 225-dimensional block-wise color moment. These features characterize images from different perspectives of color, shape and texture. On the basis of each feature, we used the L_1 metric to measure the visual distance between images. Given an image, all the other images were ranked by their distance from it and the k nearest neighbors were subsequently discovered.

Given a complex query, we generated the ranking list by sorting images in descending order of their relevance with respect to the query. We adopted the Normalized Discounted Cumulative Gain (NDCG) [43] to evaluate the quality of an ranking list. NDCG at the n-th position is computed as:

$$NDCG@n = \frac{1}{N}\sum_{i=1}^{n}\frac{2^{rel(i)}-1}{\log(1+i)}, \tag{13}$$

where $rel(i)$ is the relevance of the i-th image in the ranking list, which is defined in Eq. (6). N is a normalization constant used to ensure that the NDCG score of the ground-truth image ranking is 1. The average value of NDCG@n ($n = 10, 50, 100$) over all test complex queries was reported to evaluate the overall performance.

There are several hyper-parameters in our model. For the trade-off parameters α and β in Eq. (4), we carried out a grid search over the range of $[0, 1]$ with the granularity of 0.1. The best performance was achieved when $\alpha = 0.6$ and $\beta = 0.1$. For the dimension of the latent space d, we considered the values in the range of $[5, 50]$ with a step size of 5. The results demonstrated that there is no significant performance improvement when d is beyond 10. To reduce the computational complexity, we chose $d = 10$ on both datasets. For the number of visual neighbors k in Eq. (1), we set $k = 300$, and the effect of the value of k on the performance will be discussed later. For the regularization parameters λ_1 and λ_2 in Optimization Problem 1, we performed a logarithmic grid search from 10^{-5} to 10^5 with the scaling factor of 10, and observed the best performance when $\lambda_1 = \lambda_2 = 0.1$. In Algorithm 1, for the sample size l and the learning rate γ, we empirically used $l = 3000$ and $\gamma = 0.01$, respectively.

4.3. Competitors

We compared our approach against several state-of-the-art methods for concept-based image search. For these baseline methods, the parameters were tuned via 5-fold cross validation. Specifically, the competitors in our experiments are:

- **TagMatch**: This method simply estimates the relevance of an image based on the overlap between the tags associated with the image and the concepts in the given query.

- **TagProp** [8]: This method exploits a weighted nearest-neighbor model together with the distance metric learning to predict the presence probability of a concept. Given a complex query, it takes the product of the presence probabilities of constituent concepts as the relevance score of an image.

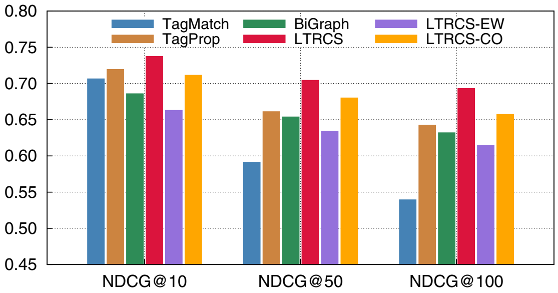


Figure 2: Performance comparison on Dataset I.

- **BiGraph** [44]: This method proposes a bi-relational graph model that comprises both the image graph and the concept graph, and connects them by an additional bipartite graph induced from concept assignments. The random walk with restart algorithm is performed over the graph by setting the constituent concepts of a complex query as the starting nodes. The relevance scores can be calculated according to the stationary distribution for all image nodes.

- **LTRCS**: This is our proposed approach that introduces the learning to rank techniques to concept-based image search for complex queries.

In our approach, we model the individual weight of each constituent concept in a complex query. The pairwise concept correlations are also modeled to capture the dependence among constituent concepts, as well as the relatedness between query and non-query concepts. To investigate the efficacy of each component, two variants of our original model were also introduced to the comparison:

- **LTRCS-EW**: Instead of explicitly modeling the weight of each constituent concept, this method assigns equal weights to all constituent concepts in a complex query.

- **LTRCS-CO**: Rather than learning the concept correlations in a supervised manner, this method uses the co-occurrence statistics as the correlation measurement.

All the methods listed above were fully implemented in Python or Matlab, and tested on a server equipped with 24-core 2.00GHz Intel Xeon processor and 32GB RAM.

4.4. Overall Performance

Figure 2 displays the empirical results of different methods on Dataset I. It is clearly shown that LTRCS consistently outperforms the other competitors in all evaluation metrics. For example, compared with TagMatch, TagProp, and BiGraph, LTRCS gains 4.4%, 2.5%, and 7.5% relative improvement in terms of NDCG@10, respectively. To further analyze the results, we performed paired t-test [45] to compare the difference

Table 3: Performance comparison among methods for complex queries with various lengths, in terms of NDCG@10.

	TagMatch	TagProp	BiGraph	LTRCS	LTRCS-EW	LTRCS-CO
2 Concepts	0.714	0.668	0.637	**0.722***	0.622	0.708
3 Concepts	0.698	0.660	0.655	**0.723***	0.618	0.706
4 Concepts	0.655	0.648	0.629	**0.716***	0.597	0.684
5 Concepts	0.620	0.640	0.629	**0.701***	0.593	0.655

Bold typeset indicates the best performance, and * indicates it is statistically significant at $p < 0.05$ compared with the runner-up.

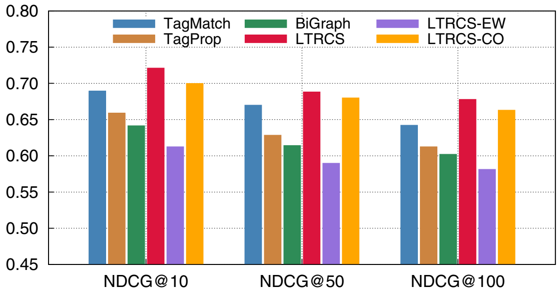


Figure 3: Performance comparison on Dataset II.

between LTRCS and the other methods, and found that the improvement of LTRCS is statistically significant at the significance level of 0.05. These results verify the potential of LTRCS in concept-based image search for complex queries.

As can be seen, in comparison to LTRCS, the two variants, LTRCS-EW and LTRCS-CO, both suffer certain performance degradation in different metrics. Since each of them determines one kind of model parameters based on heuristic rules, such results point clearly to the importance of learning concept weights and concept correlations in a supervised manner. Besides, we notice that LTRCS-EW experiences a more significant decrease in performance than LTRCS-CO, which implies that explicitly modeling concept weights makes a greater contribution to the effectiveness of our approach.

Figure 3 summarizes the comparison results on Dataset II. Again, the proposed approach LTRCS outperforms its counterparts with statistically significant improvement in all metrics. To our surprise, the existing methods, TagProp and BiGraph, substantially fall behind TagMatch, which only calculates the relevance of images by matching their associated tags against the given query. In contrast, LTRCS consistently achieves superior performance to TagMatch, reaching to 4.3% relative improvement on average. These findings further support the conclusion that our approach emerges as the most effective search scheme for complex queries among all the competitors.

4.5. *Performance Across Queries with Different Lengths*

Intuitively, a complex query composed of more concepts carries more sophisticated search intentions, which also increase the difficulty of the search task for the query. Motivated by this, we further studied how different methods behave for complex queries with various lengths. In our experiments, the length of a complex query ranged from 2 to 5 concepts. We adopted Dataset II as the evaluation testbed, since it contains sufficient queries of different lengths. Out of the 245 test queries on Dataset II, the number of queries of lengths 2, 3, 4, and 5 are 82, 95, 53, and 15, respectively.

Table 3 presents the performance of different methods for queries of different lengths in terms of NDCG@10. We can see that with the increase of the length of queries, the search performance of all methods drops gradually. This phenomenon is consistent with the intuition that it is more challenging to generate accurate search results for queries consisting of more concepts. As expected, LTRCS achieves the best performance in all cases. Especially, LTRCS gains a higher relative improvement for the complex queries with 4 or 5 concepts, leading to at least 4.7% and 7.1% for the two types in terms of NDCG@10. These results indicate that our approach is particularly applicable to long queries in concept-based image search.

4.6. *Impact of Number of Visual Neighbors*

In this study, we develop the neighbor voting algorithm to build visual concept detectors. A key parameter in the algorithm is the number of visual neighbors considered, i.e., the parameter k. To investigate the impact of k, we conducted experiments to observe the performance variation of our approach when changing k from 10 to 2000. Figure 4 shows how the performance varies with different values of k on Dataset I, where three curves fluctuate, reflecting the impact of k in terms of different metrics. It can be observed that all performance curves have a similar variation trend. Specifically, as k increases, the performance curves go up at first, but when k is beyond a certain threshold, they turn to decline with further increase of k. We believe this phenomenon is reasonable because a small number of neighbors are unable to completely characterize the semantics of a given image, whereas too many neighbors may introduce information irrelevant to that image. In our case, the best performance is achieved when $k = 300$.

4.7. *Efficiency Analysis*

To further examine the practical utility of our approach, in this subsection, we analyze the efficiency of our learning algorithm. The complexity of estimating the image relevance in Eq. (5) is $O(\overline{q}md)$, where $\overline{q}$ is the average length of complex queries. Given the fact that most complex queries are composed of only a few concepts, we have $\overline{q} \ll m$, and the complexity can be

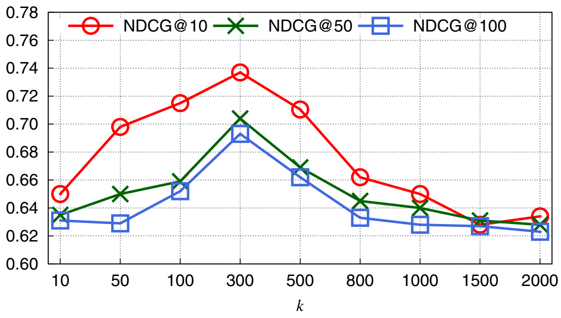


Figure 4: Impact of the number of visual neighbors *k*.

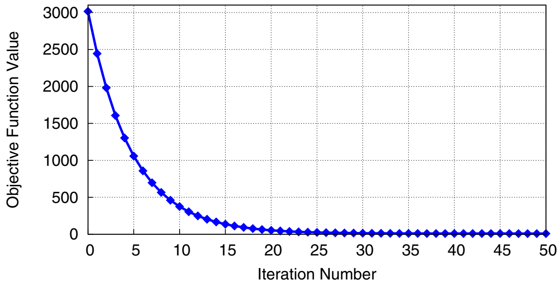


Figure 5: Convergence process of the iterative optimization.

regarded as $O(md)$. In Eq. (12), the subgradient of the relevance function f can be computed in $O(md)$. Consequently, the overall complexity of one iteration in Algorithm 1 is $O(lmd)$.

In actual experiments, our Python implementation of the algorithm took approximately 4.16 seconds per iteration on Dataset I and 7.57 seconds on Dataset II, respectively. Figure 5 displays the convergence process of the iterative optimization, which was measured by the objective function value over a set of randomly selected training triples. It shows that the algorithm generally converges within 30 iterations during training. In Table 4, we report the training time of our approach in comparison with that of the other supervised competitors, i.e., TagProp and BiGraph. Clearly, LTRCS gives a substantial reduction in the training time when compared to BiGraph. Although LTRCS takes over 1.5 times longer than TagProp, it has a significant superiority in accuracy as shown in Figure 2 and Figure 3. We believe the gain outweighs the loss. Once training is completed, during testing, our approach took an average of 0.17 seconds to yield the image ranking result for a complex query. This means that our trained model can be used interactively by users without any perceived delay. To sum up, the above analysis verifies that our approach is computationally efficient and applicable to large-scale use cases.

Table 4: Training time comparison (in seconds).

	TagProp	BiGraph	LTRCS
Dataset I	70.2	467.0	124.8
Dataset II	139.2	1324.4	227.1

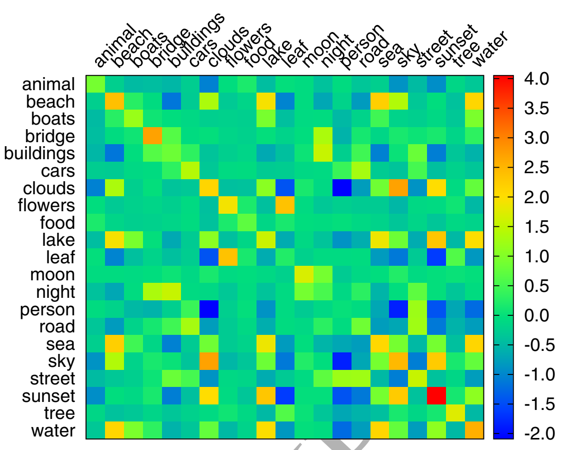


Figure 6: Illustration of the learned pairwise correlations between concepts.

4.8. Correlation Illustration

In our framework, we model the concept correlations to capture the dependence among constituent concepts as well as the relatedness between query and non-query concepts. To gain a more intuitive understanding, we randomly sample a subset of concepts, and illustrate the learned correlations between each pair of the sampled concepts in Figure 6, where a color map is used to indicate the magnitude of the correlations.

From the figure, we can see that many frequently co-occurring concepts, such as (clouds, sky), (beach, sea) and (lake, sunset), are assigned higher correlations. Analogously, the pairs of concepts with the same or similar meanings, like (road, street) and (sea, water), also have higher correlations. In contrast, lower negative correlation values are allocated to those rarely co-occurring concepts, such as (beach, buildings), (person, sky) and (leaf, sunset). Note that the range of the learned correlation values is asymmetric about zero. Moreover, the elements on the main diagonal represent the self-correlation of each concept. It can be clearly observed that, a diagonal element generally has a higher correlation value compared with the other elements in the same row or column, which is in accordance with the intuition that a concept is more correlated with itself than with others. In view of these findings, we believe that various kinds of relationships among concepts can be effectively captured by the learned correlations.

5. Conclusion and Future Work

In this paper, we have investigated the challenge of concept-based image search for complex queries, and addressed the problem from the perspective of learning to rank. With freely available social tagged images, we build concept detectors by jointly leveraging the heterogeneous visual features. To avoid the risk of making heuristic assumptions, the individual weight of each constituent concept in a complex query is explicitly modeled when estimating the image relevance. To capture the dependence among constituent concepts, as well as the relatedness between query and non-query concepts, the pairwise concept correlations are also modeled with a low-rank approximation. The learning of different parameters are performed through directly optimizing the image ranking performance for complex queries. Extensive experiments have been conducted on two benchmark datasets in comparison with the state-of-the-art methods from different aspects. The results have demonstrated the effectiveness of our approach.

Our future work will focus on three directions. Firstly, we intend to apply the distance metric learning techniques to improve the quality of visual neighbors for concept detection. Secondly, we plan to experiment with other learning to rank algorithms to enhance the learning process of our current scheme. Finally, we will further investigate the scalability of our approach by experimenting on larger image datasets.

Acknowledgements

This work is supported by Academy of Finland (268078), the Natural Science Foundation of China (71402083), and the Project of Shandong Province Higher Educational Science and Technology Program (J15LN56). Dr. Jialie Shen was supported by Singapore Ministry of Education under Academic Research Fund Tier-2 (MOE Ref: MOE2013-T2-2-156).

References

[1] E. de Ves, G. Ayala, X. Benavent, J. Domingo, E. Dura, Modeling user preferences in content-based image retrieval: A novel attempt to bridge the semantic gap, Neurocomputing 168 (2015) 829 – 845.

[2] L. Nie, M. Wang, Z.-J. Zha, T.-S. Chua, Oracle in image search: A content-based approach to performance prediction, ACM Transactions on Information Systems 30 (2) (2012) 13.

[3] L. Nie, S. Yan, M. Wang, R. Hong, T.-S. Chua, Harvesting visual concepts for image search with complex queries, in: Proceedings of the 20th ACM International Conference on Multimedia, ACM, 2012, pp. 59–68.

[4] J. Yuan, Z.-J. Zha, Y.-T. Zheng, M. Wang, X. Zhou, T.-S. Chua, Utilizing related samples to enhance interactive concept-based video search, IEEE Transactions on Multimedia 13 (6) (2011) 1343–1355.

[5] D. Guo, P. Gao, Complex-query web image search with concept-based relevance estimation, World Wide Web (2015) 1–18.

[6] Y. Yan, E. Ricci, R. Subramanian, O. Lanz, N. Sebe, No matter where you are: Flexible graph-guided multi-task learning for multi-view head pose classification under target motion, in: Proceedings of the IEEE International Conference on Computer Vision, IEEE, 2013, pp. 1177–1184.

[7] Z.-J. Zha, T. Mei, Y.-T. Zheng, Z. Wang, X.-S. Hua, A comprehensive representation scheme for video semantic ontology and its applications in semantic concept detection, Neurocomputing 95 (2012) 29–39.

[8] M. Guillaumin, T. Mensink, J. Verbeek, C. Schmid, Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation, in: Proceedings of the IEEE International Conference on Computer Vision, IEEE, 2009, pp. 309–316.

[9] C. G. Snoek, B. Huurnink, L. Hollink, M. De Rijke, G. Schreiber, M. Worring, Adding semantics to detectors for video retrieval, IEEE Transactions on Multimedia 9 (5) (2007) 975–986.

[10] X. Li, C. G. Snoek, M. Worring, A. W. Smeulders, Harvesting social images for bi-concept search, IEEE Transactions on Multimedia 14 (4) (2012) 1091–1104.

[11] B. Siddiquie, R. Feris, L. Davis, Image ranking and retrieval based on multi-attribute queries, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2011, pp. 801–808.

[12] T.-Y. Liu, Learning to rank for information retrieval, Foundations and Trends in Information Retrieval 3 (3) (2009) 225–331.

[13] L. Yang, A. Hanjalic, Supervised reranking for web image search, in: Proceedings of the 18th ACM International Conference on Multimedia, ACM, 2010, pp. 183–192.

[14] C. Cui, J. Ma, T. Lian, Z. Chen, S. Wang, Improving image annotation via ranking-oriented neighbor search and learning-based keyword propagation, Journal of the Association for Information Science and Technology 66 (1) (2015) 82–98.

[15] C. Cui, J. Shen, J. Ma, T. Lian, Social tag relevance estimation via ranking-oriented neighbour voting, in: Proceedings of the 23rd ACM International Conference on Multimedia, ACM, 2015, pp. 895–898.

[16] S. Rendle, Factorization machines with libfm, ACM Transactions on Intelligent Systems and Technology 3 (3) (2012) 57.

[17] T. Joachims, Optimizing search engines using clickthrough data, in: Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2002, pp. 133–142.

[18] B. Lu, G. Wang, Y. Yuan, D. Han, Semantic concept detection for video based on extreme learning machine, Neurocomputing 102 (2013) 176–183.

[19] S. Tang, Y.-D. Zhang, Z.-X. Xu, H.-J. Li, Y.-T. Zheng, J.-T. Li, An efficient concept detection system via sparse ensemble learning, Neurocomputing 169 (2015) 124–133.

[20] H. Li, L. Liu, F. Sun, Y. Bao, C. Liu, Multi-level feature representations for video semantic concept detection, Neurocomputing 172 (2016) 64–70.

[21] Y. Yan, Y. Yang, D. Meng, G. Liu, W. Tong, A. G. Hauptmann, N. Sebe, Event oriented dictionary learning for complex event detection, IEEE Transactions on Image Processing 24 (6) (2015) 1867–1878.

[22] X. Chang, Y. Yang, A. G. Hauptmann, E. P. Xing, Y.-L. Yu, Semantic concept discovery for large-scale zero-shot event detection, in: Proceedings of the 24th International Conference on Artificial Intelligence, 2015, pp. 2234–2240.

[23] X. Chang, Y. Yang, G. Long, C. Zhang, A. G. Hauptmann, Dynamic concept composition for zero-example event detection, arXiv preprint arXiv:1601.03679.

[24] A. P. Natsev, M. R. Naphade, J. TešiĆ, Learning the semantics of multimedia queries and concepts from a small number of examples, in: Proceedings of the 13th ACM International Conference on Multimedia, ACM, 2005, pp. 598–607.

[25] S.-F. Chang, W. Hsu, L. Kennedy, L. Xie, A. Yanagawa, E. Zavesky, D.-Q. Zhang, Columbia university trecvid-2005 video search and high-level feature extraction, in: NIST TRECVID Workshop, 2005.

[26] X. Li, D. Wang, J. Li, B. Zhang, Video search in concept subspace: a text-like paradigm, in: Proceedings of the 6th ACM International Conference on Image and Video Retrieval, ACM, 2007, pp. 603–610.

[27] J. Yuan, Z.-J. Zha, Y.-T. Zheng, M. Wang, X. Zhou, T.-S. Chua, Learning concept bundles for video search with complex queries, in: Proceedings of the 19th ACM International Conference on Multimedia, ACM, 2011, pp. 453–462.

[28] W. Zhao, Z. Guan, Z. Liu, Ranking on heterogeneous manifolds for tag recommendation in social tagging services, Neurocomputing 148 (2015) 521–534.

[29] L. Nie, Y.-L. Zhao, X. Wang, J. Shen, T.-S. Chua, Learning to recommend descriptive tags for questions in social forums, ACM Transactions on Information Systems 32 (1) (2014) 5.

[30] C. Kang, D. Yin, R. Zhang, N. Torzec, J. He, Y. Chang, Learning to rank related entities in web search, Neurocomputing 166 (2015) 309–318.

[31] S. Huang, S. Wang, T.-Y. Liu, J. Ma, Z. Chen, J. Veijalainen, Listwise collaborative filtering, in: Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2015, pp. 343–352.

[32] X. Li, C. Snoek, M. Worring, Learning social tag relevance by neighbor voting, IEEE Transactions on Multimedia 11 (7) (2009) 1310–1322.

[33] T. Uricchio, L. Ballan, M. Bertini, A. Del Bimbo, An evaluation of nearest-neighbor methods for tag refinement, in: Proceedings of the IEEE International Conference on Multimedia and Expo, IEEE, 2013, pp. 1–6.

[34] X. Li, Tag relevance fusion for social image retrieval, Multimedia Systems (2014) 1–12.

[35] X. Li, T. Uricchio, L. Ballan, M. Bertini, C. G. Snoek, A. Del Bimbo, Socializing the semantic gap: A comparative survey on image tag assignment, refinement and retrieval, arXiv preprint arXiv:1503.08248.

[36] R. Zhao, W. Grosky, Narrowing the semantic gap-improved text-based web document retrieval using visual features, IEEE Transactions on Multimedia 4 (2) (2002) 189–200.

[37] X. Li, Y.-J. Zhang, B. Shen, B.-D. Liu, Low-rank image tag completion

with dual reconstruction structure preserved, Neurocomputing 173 (2016) 425–433.

[38] S. Shalev-Shwartz, Y. Singer, N. Srebro, A. Cotter, Pegasos: Primal estimated sub-gradient solver for svm, Mathematical Programming 127 (1) (2011) 3–30.

[39] M. Huiskes, M. Lew, The mir flickr retrieval evaluation, in: Proceedings of the 1st ACM International Conference on Multimedia Information Retrieval, ACM, 2008, pp. 39–43.

[40] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, Y. Zheng, Nus-wide: a real-world web image database from national university of singapore, in: Proceedings of the ACM International Conference on Image and Video Retrieval, ACM, 2009, pp. 48:1–48:9.

[41] X. Cao, H. Zhang, X. Guo, S. Liu, X. Chen, Image retrieval and ranking via consistently reconstructing multi-attribute queries, in: Proceedings of the 19th European Conference on Computer Vision, Springer, 2014, pp. 569–583.

[42] A. Spink, D. Wolfram, M. B. Jansen, T. Saracevic, Searching the web: The public and their queries, Journal of the American Society for Information Science and Technology 52 (3) (2001) 226–234.

[43] L. Nie, M. Wang, Z. Zha, G. Li, T.-S. Chua, Multimedia answering: enriching text qa with media information, in: Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2011, pp. 695–704.

[44] H. Wang, H. Huang, C. Ding, Image annotation using bi-relational graph of images and semantic labels, in: Proceedings of the IEEE International Conference on Computer Vision, IEEE, 2011, pp. 793–800.

[45] L. Nie, L. Zhang, Y. Yang, M. Wang, R. Hong, T.-S. Chua, Beyond doctors: future health prediction from multimedia and multimodal observations, in: Proceedings of the 23rd ACM International Conference on Multimedia, ACM, 2015, pp. 591–600.

Chaoran Cui received the B.S. degree in Software Engineering in 2010 and the Ph.D. degree in Computer Science and Technology in 2015, both from Shandong University, Jinan, China. At present, he is a postdoctoral research fellow in School of Information Systems, Singapore Management University (SMU), Singapore. His research interests include information retrieval, social multimedia, and computer vision.

Jialie Shen is an assistant professor in Information Systems and Lee Foundation Fellow, School of Information Systems, Singapore Management University (SMU), Singapore. He received his Ph.D. in Computer Science from the University of New South Wales (UNSW), Australia in the area of large-scale media retrieval and database access methods. He worked as a faculty member at UNSW, Sydney and researcher at information retrieval research group, the University of Glasgow for a few years, before moving to the SMU, Singapore. His main research interests include information retrieval, economic-aware media analysis, and statistical machine learning.

Zhumin Chen received his Ph.D. degree in Computer Science and Technology from Shandong University, Jinan, in 2008. Currently, he is an associate professor and master supervisor in the School of Computer Science and Technology, Shandong University. He is a senior member of CCF and a member of ACM. His research interests include Web information retrieval, data mining and social network analysis.

Shuaiqiang Wang received the B.S. and Ph.D. degrees in Computer Science from Shandong University, China, in 2004 and 2009, respectively. Currently, he is an assistant professor at the University of Jyväskylä, Finland. Before that, he was an associate professor at the Shandong University of Finance and Economics, China, from 2011 to 2014, and a postdoctoral research associate at Texas State University in 2010. His research interests include information retrieval and data mining. He is a member of the IEEE.

Jun Ma received the B.S. degree from Ibaraki University, Japan and the Ph.D. degree from Kyushu University, Japan. He worked as a senior researcher in Ibaraki University in 1994 and in German National Computer Research Center (GMD) from 1999 to 2004. Now he is a professor in School of Computer Science and Technology, Shandong University, Jinan, China. His research interests include information retrieval, data mining, parallel computing and natural language processing.

Author 1

Author 2

Author 3

Author 4

Author 5