ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 10, March 2016

A Pre-Eminence Race of Images Based on Magnanimous Views

L.Gopalakrishnan, *Member*, J.Sabari, *Member*, *G.Sureshkannan*, *Member*, and S.N.Vignesh, *Member* Guided by S.Senthi Arumugam AP/CSE,Sasurie College of Engineering,Vijayamangalam ,Email : vgnsh814@gmail.com

***Abstract*—** The inconsistency between textual features and visual contents can cause poor image search results. To solve this problem, click features, which are more reliable than textual information in justifying the relevance between a query and clicked images, are adopted in image ranking model. However, the existing ranking model cannot integrate visual features, which are efficient in refining the click-based search results. In this paper, we propose a novel ranking model based on the learning to rank framework. Visual features and click features are simultaneously utilized to obtain the ranking model. Specifically, the proposed approach is based on large margin structured output learning and the visual consistency is integrated with the click features through a hypergraph regularizer term. In accordance with the fast alternating linearization method, we design a novel algorithm to optimize the objective function. This algorithm alternately minimizes two different approximations of the original objective function by keeping one function unchanged and linearizing the other. We conduct experiments on a large-scale dataset collected from the Microsoft Bing image search engine, and the results demonstrate that the proposed learning to rank models based on visual features and user clicks outperforms state-of-the-art algorithms

.***Index Terms*—Click, Magnanimous , learning to rank.**

I. Introduction

LEARNING to rank has been widely adopted in the fields of information retrieval, data mining, and natural language processing. In general, given a query, the learning to rank system retrieves data from the collection and returns the top-ranked data. A model $f = (q, d)$ can be used to describe the ranking assignment, where q represents a query The learning to rank approach has also been widely used in image retrieval. The query dependent features for each image are extracted from textual information to describe the relation- ship between a query and an image. The textual information

Manuscript received March 28, 2014; revised July 1, 2014; accepted July 2, 2014. Date of publication July 29, 2014; date of current version March 13, 2015. This work was supported in part by the National 973 Program of China under Grant 2014CB347600, in part by the National Natural Science Foundation of China under Grant 61100104, Grant 61272393, and Grant 61322201, in part by the ARC under Grant DP-140102164, Grant FT-130101457, and Grant LP-140100569, in part by the Program for New Century Excellent Talents in University under Grant NCET-12-0323, and in part by the Hong Kong Scholar Programme under Grant XJ2013038. This paper was recommended by Associate Editor L. Shao.

J. Yu is with the School of Computer Science, Hangzhou Dianzi University, Hangzhou 310018, China (e-mail: zju.yujun@gmail.com).

D. Tao is with the Centre for Quantum Computation and Intelligent Systems and the Faculty of Engineering and Information Technology, University of Technology, Sydney, NSW 2007, Australia (e-mail: dacheng.tao@uts.edu.au).

M. Wang is with the School of Computer Science and Information Engineering, Hefei University of Technology, Hefei, Anhui 230009, China (e-mail: eric.mengwang@gmail.com).

Y. Rui is with Microsoft Research Asia, Peking 100080, China (e-mail: yongrui@microsoft.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2014.2336697

and d denotes a data sample. Learning to rank has extensive uses in retrieving documents, searching definitions, answering questions, and summarizing documents [1]. Traditionally, the ranking model $f = (q, d)$ is manually created without training, as in the case of the BM25 ranking function [2]. This model describes $f = (q, d)$ by a conditional probability distribution $p(r|q, d)$. Here, r is assigned with a binary value (1 or 0), and denotes relevance and irrelevance, respectively. To take document retrieval as an example, the probability model can be constructed using the frequency of words shown in the query and the document. The training stage is therefore unnecessary.

Ranking has recently come to be regarded as a learning problem and some machine learning algorithms [3]–[6] have been applied to it. In these methods, the training data is formally described through pairs: $\mathbf{x}^q_i, l^q_i$, where q denotes the query number and i denotes the document number for query q. $\mathbf{x}^q_i$ represents the feature vector, and l_{iq} is its relevance label. To measure the performance of a search engine, the discounted cumulative gain (DCG) has been widely adopted to evaluate relevance in the context of search engines [7]. Treating learning to rank as either a regression or classification problem, Cossock and Zhang [4] formulated this problem using the objective function of a regression model, obtained through pointwise operations. However, these methods neglect the preference relationship that exists among the documents. To solve this problem, a method called the pairwise approach [8], [9] has been proposed and successfully used in document retrieval. This approach collects document pairs from the ranking lists, and assigns a label to each pair that describes the relative relevance of the two documents. It then trains a classification model with the labeled data and adopts it for ranking. Though the training samples of document pairs can be collected easily, the objective of learning is formalized as optimizing errors in the classification of document pairs, rather than optimizing errors in the ranking of documents [10]. To deal with this problem, listwise approaches [11], [12] have been proposed to learn a ranking function by adopting separate lists as samples. The loss function is formulated on the predicted list and the ground-truth list.

The learning to rank approach has also been widely used in image retrieval. The query dependent features for each image are extracted from textual information to describe the relationship between a query and an image. The textual information sources include the title, the surrounding text, the HTML alternative texts, or the titles of the host webs. A ranking model is then obtained from the textual features and the manually labeled training set. In practice, the famous image search engines of Google, Yahoo!, and Bing adopt textual information to index web images. Although the performance is acceptable for many queries, the accuracy of retrieved images is still not high in most cases. The probable mismatch between the content of an image and the text from a web page is a major problem. The extracted text does not always precisely describe the characteristics of the image content, as required by the query. One feasible solution to address this problem is to integrate visual information of images [42]–[44], [46], [48]–[50] into the rank learning framework [45], [47]. The query related features can be extracted to represent the relationship between the query and the visual contents, and the textual features can then be integrated with them. Hua and Qi [13] proposed a novel online multilabel learning approach to enable efficient semantic concept annotation in image search. The results of concept detection can be conveniently utilized by existing ranking models; however, it is infeasible to obtain a satisfactory concept detector for each query term due to the complexity of both the semantic concepts and the visual content.

Another feasible solution for combining visual information is visual reranking [14], [15], [39], [40], which combines both the textual and visual information and returns visually satisfying retrieved results. Hence, the ranking list of images obtained from the text-based search can be regarded as a reasonable baseline with certain noises. The visual information of the images is then adopted to shift the related images to the top of the ranking list; thus, visual reranking only adopts the visual information to refine the text-based results rather than assisting the learning process of the text-based ranking model. Many existing reranking methods are based on implicitly adopting pseudo-relevance feedback (PRF); for instance, Yan *et al.* [16] proposed a classification-based method which utilizes uppermost images as pseudo-positive and undermost images as pseudo-negative examples to train a classifier and conduct reranking. Hsu *et al.* [17] also adopted the pseudo-positive and pseudo-negative images to develop a clustering-based reranking method. However, visual reranking methods cannot successfully relegate irrelevant images which have originally been allocated a high rank, and suffer from an unreliable original ranking list because the textual information cannot accurately describe the semantics of the queries.

Instead of textual information, user click has recently been used to measure the relationship between queries and retrieved objects [18], [19], because a number of research works have found that click is more reliable [20] than textual information in justifying the relevance between a query and clicked objects. In this paper, we present a novel ranking model which successfully applies visual features and click features to image retrieval. We call this model visualand click features based learning to rank (VCLTR). Our approach can handle the problems of textual features, i.e., the semantic gaps in describing the relevance between images and query, and can also overcome the drawbacks of visual reranking, i.e., the noise spread and the inability to relegate irrelevant images that have initially been ranked in a high position. Using the click features creates a robust and accurate ranking model, and adopting the visual features will further enhance the model's performance.

There are two important issues in proposing a novel image ranking model. First, the ranking of images is determined according to the interactions between those images. The ranking result is a structured list, but traditional learning algorithms cannot handle the structured result. Second, unlike click features, which are extracted according to specific query, visual features are obtained from images regardless of queries. Therefore, the traditional learning to rank approaches cannot be used directly. Accordingly, we propose a new objective function for our learning to rank model under the framework of large margin structured output learning [21]. Specifically, there are two terms in the objective function: the click features are integrated in terms of a linear model, and the visual features are considered in terms of a hypergraph regularizer [22], [33], [34], [36]–[38], [51] which captures high-order relationships in building the graph. It is nontrivial to directly solve this problem. In accordance with the fast alternating linearization method (FALM) [23], we design a novel algorithm to optimize the object function. This algorithm can alternately minimize two different approximations of the original objective function by keeping one function unchanged and linearizing the other. The experiments are conducted on a large-scale dataset collected from a commercial web image search engine, and the results demonstrate excellent performance by the proposed method.

In summary, the contributions of this paper are threefold.

1) First, we propose a novel learning to rank model called VCLTR which jointly considers visual features and click features in image retrieval. A robust and accurate ranking model can be built by using the click features, and the visual features are effective in further enhancing the model's performance.
2) Second, by integrating the visual features and click features, we design a novel objective function, in which the terms hypergraph regularizer and linear model are, respectively, adopted to take these two features into consideration. We then design a novel optimization algorithm based on FALM to efficiently solve the objective function.
3) Finally, the proposed VCLTR is evaluated over a large-scale and practical image search dataset, in which the click features are collected from real web users. The experimental results suggest the effectiveness of our method.

II. RELATED WORK

In this section, we first provide basic notations for the algorithm in this paper and then introduce definitions about structure learning.

A. Basic Notations

In this paper, we assume that an image is represented by both click information and visual information. The query set for the ranking model can be defined as Q. For a specified query j in the set $q^j \in Q$, it is defined as a triplet: $\{\mathbf{c}^j, \mathbf{x}^j, \mathbf{r}^j\}$.

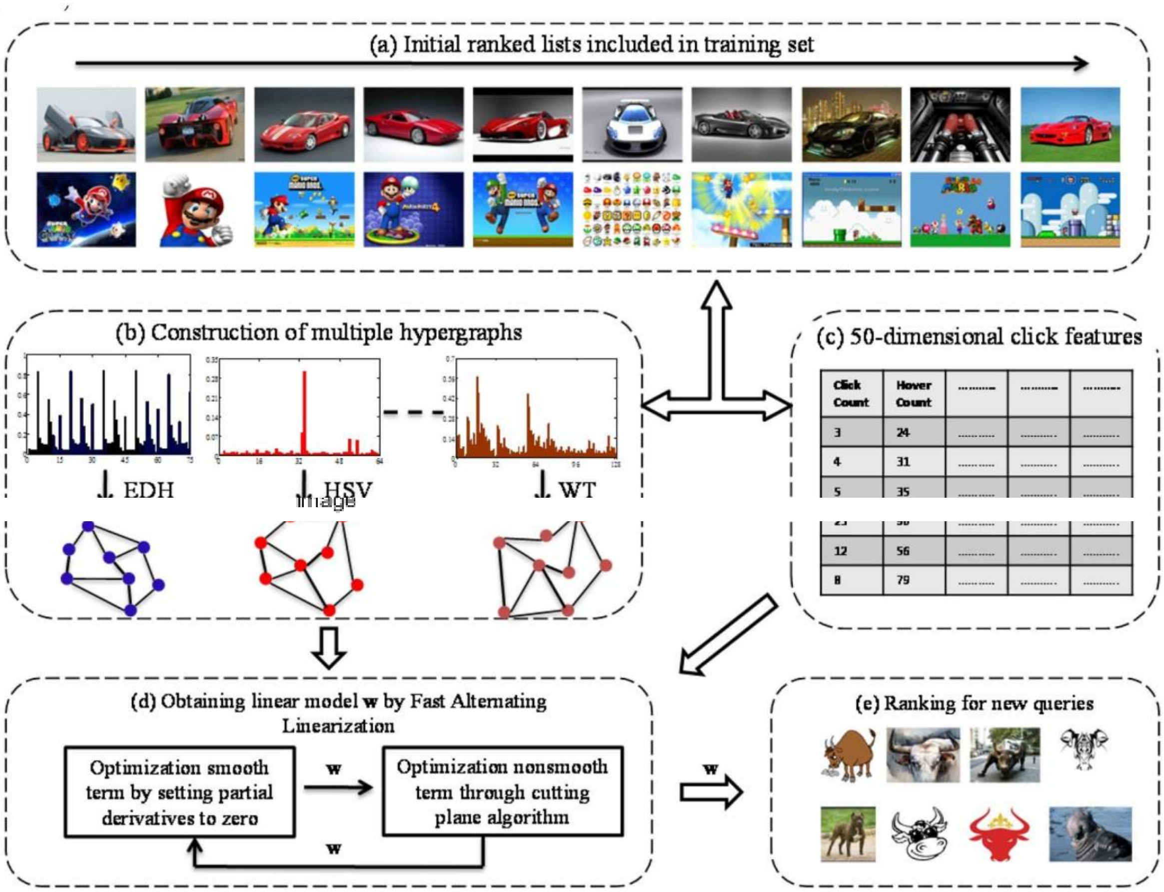


Fig. 1. Framework of click-based image ranking with hypergraph learning. (a) Training set construction using initial ranked image lists. (b) Construction of multiple hypergraphs from different visual features. (c) Building 50-dimensional click features. (d) Optimization of linear model **w** through fast alternating linearization. There are two iterative stages: optimizing smooth term and optimizing nonsmooth term. (e) Ranking the images from new queries.

Here, $\mathbf{c}^j = \mathbf{c}_{1j}, \ldots, \mathbf{c}_{N^j j} \in C$ are the query dependent click features; $\mathbf{x}^j = \mathbf{x}_{1j}, \ldots, \mathbf{x}_{N^j j} \in X$ represents the visual features; $\mathbf{r}^j = \mathbf{r}_{1j}, \ldots, \mathbf{r}_{N^j j} \in R$ describes the results of ranking according to the labels by human experts. N^j is the number of images corresponding to the query j. In this case, $\mathbf{c}_{ij} \in R^{D_c}$, $\mathbf{x}_{ij} \in R^{D_x}$, $\mathbf{r}_{ij} \in R$ are the D_c dimensional click features, D_x dimensional visual features and the ranking result of image i according to query q^j. In general, we can denote the ranking function as $f : C \times X \rightarrow R$, which projects the click and visual feature spaces $C \times X$ to the ranking space R. Therefore, the aim of rank learning is to obtain the optimal ranking function f. The expected ranking loss in the training set Q can be defined as

$$L_Q(f) = \frac{1}{N} \sum_{j=1}^{N} L\left(\mathbf{r}^j, f\left(\mathbf{c}^j, \mathbf{x}^j\right)\right) \tag{1}$$

where the function $L\left(\mathbf{r}^j, \underline{\mathbf{r}}\right)$ estimates the loss of the ranking result $\underline{\mathbf{r}} = f\left(\mathbf{c}^j, \mathbf{x}^j\right)$ with the ground-truth $\mathbf{r}^j$. The scale of query Q is N.

B. Structure Learning for Ranking

According to the elaborations in [21], the ranking result is structured because the ranks of different images for a query are interdependent. Therefore, we can adopt the large margin structured output learning framework [21] to formulate the rank learning problem. As shown in [21], given the query with click features $\mathbf{c}^j$, the ranking function can be defined as

$$\underline{\mathbf{r}} = \psi\left(\mathbf{c}^j\right) = \arg\max_{\mathbf{r}\in R} \left(\mathbf{c}^j, \mathbf{r}; \mathbf{w}\right) \tag{2}$$

where $\mathbf{w}$ is the model parameter vector. $\left(\mathbf{c}^j, \mathbf{r}; \mathbf{w}\right)$ can be defined as a linear function of $\mathbf{w}$ in

$$\left(\mathbf{c}^j, \mathbf{r}; \mathbf{w}\right) = \mathbf{w}^T \theta\left(\mathbf{c}^j, \mathbf{r}\right) \tag{3}$$

where $\theta\left(\mathbf{c}^j, \mathbf{r}\right)$ projects the click feature $\mathbf{c}^j$ and the ranking prediction $\mathbf{r}$ into real values. We use $\theta\left(\mathbf{c}^j, \mathbf{r}\right) = \sum_{i=1}^{N} \mathbf{c}_{ij} \mathbf{r}_i$, and then (3) can be rewritten as

$$\underline{\mathbf{r}} = \arg\max_{\mathbf{r}\in R} \mathbf{u}^{jT} \mathbf{r} \tag{4}$$

where $\mathbf{u}^j = [\mathbf{w}^T\mathbf{c}_{1j}, \ldots, \mathbf{w}^T\mathbf{c}_{N^j j}]$ denote the score list of the images for the query q^j. Intuitively, the cosine angle between the score list $\mathbf{u}^j$ and $\mathbf{r}$ are maximized to make the direction of $\mathbf{r}$ comply with $\mathbf{u}^j / \|\mathbf{u}^j\|$.

III. Our Proposed Learning to Rank Model

Because click features are noisy, the click-based ranking model of (2) does not perform well in visual search. In this

TABLE I
IMPORTANT NOTATIONS USED IN THIS PAPER AND THEIR DESCRIPTIONS

Notations	Descriptions	Notations	Descriptions
Q	Query set Q for ranking model	$\mathcal{G}=(\mathcal{V},\mathcal{E},\omega)$	The hypergraph $\mathcal{G}$ formed by vertex set $\mathcal{V}$, hyperedge set $\mathcal{E}$, and hyperedge weight vector ω.
q^j	The specified query j is defined as a triplet $\{\mathbf{c}^j,\mathbf{x}^j,\mathbf{r}^j\}$	$\mathbf{H}$	The incidence matrix for hypergraph
$\mathbf{c}^j=\left[\mathbf{c}_1^j,\ldots,\mathbf{c}_{N^j}^j\right]$	Query dependent click features	$\mathbf{D}_v$	The diagonal matrices of vertex degrees
$\mathbf{x}^j=\left[\mathbf{x}_1^j,\ldots,\mathbf{x}_{N^j}^j\right]$	Visual features	$\mathbf{D}_e$	The diagonal matrices of hyperedge degrees
$\mathbf{r}^j=\left[\mathbf{r}_1^j,\ldots,\mathbf{r}_{N^j}^j\right]$	The ranking results according to the labels applied by human experts	$\Psi(\mathbf{w},\mathbf{c}^j,\mathbf{x}^j,\mathbf{r})$	The ranking function
N^j	The number of images corresponding to the query j	$\theta(\mathbf{c}^j,\mathbf{r})$	This function projects the click feature $\mathbf{c}^j$ and the ranking prediction $\mathbf{r}$ into real values
$L(\mathbf{r}^j,\bar{\mathbf{r}})$	The function estimates the loss of the ranking result $\bar{\mathbf{r}}$ with the ground-truth $\mathbf{r}^j$	η	A Lagrange multiplier for eliminating the equality constraint.
$\mathbf{w}$	The model parameter vector	S^j	The working set for the jth query
ξ^j	The slack variable for query j		

paper, we propose a novel ranking model which utilizes visual features and click features to support the ranking model learning. The basic assumption is that relevant images for a query should obtain the characteristic of visual consistency, and visually similar images should obtain a similar ranking output. Fig. 1 provides the details of the proposed algorithm. First, groups of images with their ranks are collected to form the training set. Then, we collect 50-dimensional click features (including click count and hover count) corresponding to these images, and build multiple hypergraphs from the visual features of the images. Fast alternating optimization is conducted to obtain the model **w**, which is used for the ranking of new queries. The smooth and nonsmooth terms in the objective function are separately solved through two iterative stages: optimizing the smooth term (the closed form solution is obtained by setting the partial derivative to zero) and optimizing the nonsmooth term (using the cutting plane algorithm).

Important notations used in this paper are presented in Table I.

A. Problem Formulations

Since some noises exist in click features, the click only ranking model as shown in (3) is insufficient for the visual search. It is assumed that relevant images for a query should have the visual consistency property. Based on the assumption, we will prefer a ranking list that is not only the relevant from the click feature perspective, but also possessing a high visual consistency. Inspired by the graph based Laplacian [24], the proposed model takes the following form:

$$\begin{aligned}\underline{\mathbf{r}} &= \arg\max_{\mathbf{r}\in R} \mathbf{w}, \mathbf{c}^j, \mathbf{x}^j, \mathbf{r} \\ &= \arg\max_{\mathbf{r}\in R} \mathbf{w}^T\theta(\mathbf{c}^j, \mathbf{r}) - \lambda R(\mathbf{G}^j, \mathbf{r}) \qquad (5)\end{aligned}$$

where $\lambda > 0$ is a tuning parameter to balance the click relevance $\mathbf{w}^T\theta(\mathbf{c}^j, r)$ and the visual consistency term $R(\mathbf{G}^j, \mathbf{r})$. $\mathbf{G}^j$ represents the adjacency graph which measures the similarities between each pair of images, with the elements defined as

$$\mathbf{G}_{mn}^j = \begin{cases}\mathrm{Sim}(\mathbf{x}_m^j, \mathbf{x}_n^j), & \text{if } \mathbf{x}_n^j \in N(\mathbf{x}_m^j)\\ 0, & \text{otherwise.}\end{cases} \qquad (6)$$

Here, Sim $(\mathbf{x}_m^j,\mathbf{x}_n^j)$ defines the similarity between $\mathbf{x}_m^j$ and $\mathbf{x}_n^j$, and $\mathbf{G}_{mn}^j$ is a simple graph [24] by employing k-nearest-neighbor (KNN) or epsilon-ball strategy to define the neighborhood relationship $N(\bullet)$. The introduced term $R(\mathbf{G}^j, \mathbf{r})$ in (5) aims to constraint that visually similar images are assigned with consistent rankings, and its definition can be achieved from various perspectives.

The graph based Laplacian proposed by [24], is a well-known regularizer that utilizes the manifold geometric of the marginal distribution information underlying the data features to boost the learning task. The geometric information is formally modeled by a graph consistency term, which under our visual consistency setting is defined as

$$\begin{aligned}&\arg\min_{\mathbf{r}\in R} R(\mathbf{G}^j, \mathbf{r})\\ &= \arg\min_{\mathbf{r}\in R} \frac{1}{2}\sum_{m,n=1}^{N^j} \mathbf{G}_{mn}^j(\mathbf{r}_m - \mathbf{r}_n)^2\\ &= \arg\min_{\mathbf{r}\in R} \sum_{m=1}^{N^j} \mathbf{G}_{mn}\mathbf{r}_m^2 - \sum_{m,n=1}^{N^j} \mathbf{G}_{mn}\mathbf{r}_m\mathbf{r}_n\\ &= \arg\min_{\mathbf{r}\in R} \mathbf{r}^T\mathbf{L}_{\mathrm{Lap}}^j\mathbf{r} \qquad (7)\end{aligned}$$

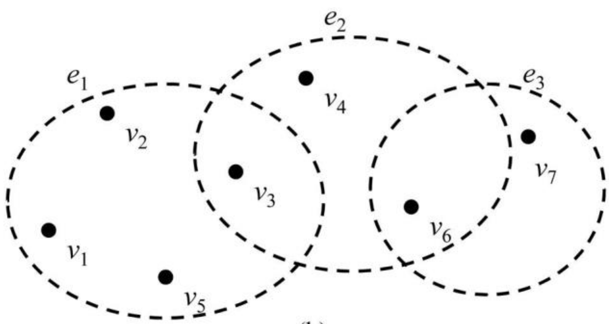
where $\mathbf{L}_{\mathrm{Lap}}^j$ is the Laplacian matrix of the graph. The incorporation of this term into the ranking function will drive visually similar documents to be assigned with similar rank predictions, which implements our visual consistency objective. Since the problem in (7) is a minimization problem, we need to add minus before it to incorporate it with (3), which is a maximization problem. Therefore, we obtain the objective function (5), which can be rewritten as

$$\begin{aligned}\underline{\mathbf{r}} &= \arg\max_{\mathbf{r}\in R} \mathbf{w}, \mathbf{c}^j, \mathbf{x}^j, \mathbf{r}\\ &= \arg\max_{\mathbf{r}\in R} \mathbf{w}^T\theta(\mathbf{c}^j, \mathbf{r}) - \lambda\, \mathbf{r}^T\mathbf{L}_{\mathrm{Lap}}^j\mathbf{r}. \qquad (8)\end{aligned}$$

The graph based Laplacian methods [24] consider only the pairwise relationship between two samples and ignore the relationship in a higher order. For instance, from a graph we

	e_1	e_2	e_3
v_1	1	0	0
v_2	1	0	0
v_3	1	1	0
v_4	0	1	0
v_5	1	0	0
v_6	0	1	1
v_7	0	0	1

(a)



(b)

Fig. 2. Hypergraph construction. (a) Vertices set V and the hyperedge set E. (b) Hypergraph which completely illustrates the complex relationships among vertices.

can easily find two close samples according to the pairwise similarities, but it is not easy to predict whether there are three or more close samples. Essentially, modeling the high-order relationship among samples will significantly improve ranking performance. Hypergraph learning [22] can address this problem. Unlike a graph [24] that has an edge between two vertices, a set of vertices is connected by a hyper-edge in a hypergraph. The following notations are used to describe the hypergraph learning. Let V denote a finite set of objects, and let E be a family of subsets e of V such that $_{e\in E} = V$. The details of hypergraph construction is shown in Fig. 2. Hence, a hypergraph $G = V, E, \omega$ is formed by the vertex set V, the hyperedge set E, and the hyper-edge weight vector ω. Here, each hyperedge e_i is assigned a weight $\omega(e_i)$. A $V \times |E|$ incidence matrix $\mathbf{H}$ denotes G with elements

$$H(v, e) = \begin{cases} 1 & \text{if } v \in e \\ 0 & \text{if } v \notin e. \end{cases} \tag{9}$$

Based on $\mathbf{H}$, the vertex degree of each vertex $v \in V$ is

$$d(v) = \sum_{e\in E} \omega(e) H(v, e) \tag{10}$$

and the edge degree of a hyperedge $e \in E$ is

$$\delta(e) = \sum_{v\in V} H(v, e). \tag{11}$$

We use $\mathbf{D}_v$ and $\mathbf{D}_e$ to denote the diagonal matrices of vertex degrees and hyperedge degrees, respectively. Let $\mathbf{W}$ represent the diagonal matrix of the hyperedge weights. According to [22], the regularizer on the hypergraph is defined by

$$(\mathbf{r}) = \frac{1}{2} \sum_{e\in E} \sum_{v_i, v_j \in V} \frac{\omega(e) H(v_i, e) H(v_j, e)}{\delta(e)} \left(\frac{r_{v_i}}{\sqrt{d(v_i)}} - \frac{r_{v_j}}{\sqrt{d(v_j)}} \right)^2 \tag{12}$$

where v_i and v_j are two selected vertices. According to the definitions in [22], the hypergraph Laplacian can be constructed by $\mathbf{L}_{hyperLap} = \mathbf{I} -$, where $= \mathbf{D}_v^{-(1/2)} \mathbf{H} \mathbf{W} \mathbf{D}_e^{-1} \mathbf{H}^T \mathbf{D}_v^{-(1/2)}$. The hypergraph Laplacian $\mathbf{L}_{hyperLap}$ can be used in (8) to replace graph based Laplacian matrix.

B. Learning w

We adopt the large margin structured output learning [21] framework to estimate the model parameters $\mathbf{w}$ in (8). This framework can handle the learning of complex and structured outputs like trees, sets and ranking lists. Based on the labeled training set Q, we want to obtain a weight vector $\mathbf{w}$ so that the ranking model can perfectly predict the ranks of the images for the queries in Q. Slack variables [35] are introduced to accommodate the noises in the training data, and the proposed learning problem is defined as follows:

$$\begin{aligned} &\min_{\mathbf{w},\xi} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{j=1}^{|Q|} \xi_j \\ &\text{s.t. } \forall q^j \in Q, \xi_j \geq 0, \forall \mathbf{r} = \mathbf{r}^j \\ &\quad \mathbf{w}, \mathbf{c}^j, \mathbf{v}^j, \mathbf{r}^j - \mathbf{w}, \mathbf{c}^j, \mathbf{v}^j, \mathbf{r} \geq \mathbf{r}^j, \mathbf{r} - \xi_j \end{aligned} \tag{13}$$

where $C > 0$ is the trade-off parameter to balance the model complexity $\|\mathbf{w}\|^2$ and the upper bound of the prediction loss $\sum_{j=1}^{|Q|} \xi_j$. $\mathbf{r}^j, \mathbf{r}$ is the ranking loss function to measure the loss between the prediction $\mathbf{r}$ and the ground truth $\mathbf{r}^j$.

The direct optimization of (13) is nontrivial. In this part, we adopt the fast alternating linearization method (FALM) [23] to solve this problem. This algorithm can alternatively minimize two different approximations of the original objective function, obtained by keeping one function unchanged and linearizing the other. According to [23], we can separate the original function into two parts as

$$f(\mathbf{w}) = \frac{1}{2} \|\mathbf{w}\|^2 \tag{14}$$

and

$$\begin{aligned} &g(\mathbf{w}) = C \sum_{j=1}^{|Q|} \xi_j \\ &\text{s.t. } \forall q^j \in Q, \xi_j \geq 0, \forall \mathbf{r} = \mathbf{r}^j \\ &\quad \mathbf{w}, \mathbf{c}^j, \mathbf{v}^j, \mathbf{r}^j - \mathbf{w}, \mathbf{c}^j, \mathbf{v}^j, \mathbf{r} \geq \mathbf{r}^j, \mathbf{r} - \xi_j. \end{aligned} \tag{15}$$

Thus, we can transform the initial problem in (13) as

$$\min\{f(\mathbf{w}) + g(\mathbf{z}) : \mathbf{w} - \mathbf{z} = \mathbf{0}\}. \tag{16}$$

V.MODULES

The project contains six modules. They are,

- User Endorsement
- Training Data Set
- Semantic chain transitions
- Clustering Keywords With Weight Vector
- Efficient Search Result
- Report
  - User Wise Mining
  - Image Utilization

A.USER ENDORSEMENT

User Endorsement is the initial module in this application. The new user has to do the registration process to access the application in online. The registration process includes username, password, address, phone etc. Once the registration process is completed successfully the user can login with the username and password and then image search is performed.

B.TRAINING DATA SET

During the training phase of the system the images are considered with no annotation. The images are loaded with certain similarity of keywords. As the users issue queries and the images is picked based on the similarity measure between the user query and the web page information. The system automatically identifies the similarity images based on the Meta information. The user never annotates the images explicitly, this happens by the system transparently from the user. The system uses the annotations available from the training phase but also the keyword relevance probability weights also evaluated during the training phase to return images that better reflect the users preferences and improve user satisfaction.

C.SEMANTIC CHAIN TRANSITIONS

The user implicitly relates the retrieved (downloaded) images to user query. The semantic chain transitions in the order of the keywords the aim of the proposed approach is to quantify logical connections between keywords. If some user relates image to his query, where keyword follows keyword and this occurs m times, then the one step transition probability is being updated this procedure constructs a sequence chain where each keyword corresponds to a state. Each time a keyword appears in a query, its state counter is advanced; if another keyword follows in the same query, their interstate link counter is also advanced. The occurrences of the keywords but also the sequencing of these occurrences is both measured this way. The queries pertaining to an image are batch processed for this image, the counters are advanced, and the probabilities are updated as efficient results.

D.CLUSTERING KEYWORDS WITH WEIGHT VECTOR

In this module, the relation between the image and the keyword mapped in the sequence transactions are aggregation here. By clustering the keyword space into similar keywords fast retrieval can be performed. For this purpose, the Aggregate sequence indexing known as weight of all the queries asked by all users regardless of the selected images is constructed in this step. The kernel of this process is calculated in a similar to the previous step even though a kernel it will be used to cluster the keyword space rather than estimating an explicit probability distribution, hence the purpose of the aggregate sequence is to model keyword relevance. So the optimization is performed. The aggregate sequence will be used to cluster the keyword space and define explicit relevance links between the keywords by means of this clustering.

E.EFFICIENT SEARCH RESULT

The Efficient Search Result is the final module in this project. Here user submits the query to retrieve the respective image they required. The server process the high level image retrieving techniques such as the aggregate sequence transaction and clustering is performed based on the keyword aggregation using semantic indexing and checks the relationship between the image and the keyword and shortlist the unwanted images and efficient search result will be displayed to the user.

F.REPORT

Report is the final module in this application. Here the user wise mining and maximum utilization of image in the search process is taken as the report for future transaction. In the user wise mining the favorite type of image of an individual can be identified. In the image utilization process the images maximum downloaded by the end user is identified.

VI.BLOCK DIAGRAM

A method called the pair wise approach has been proposed and successfully used in document retrieval. This approach collects document pairs from the ranking lists, and assigns a label to each pair that describes the relative relevance of the two documents. It then trains a classification model with the labeled data and adopts it for ranking. Though the training samples of document pairs can be collected easily, the objective of learning is formalized as optimizing errors in the classification of document pairs, rather than optimizing errors in the ranking of documents. To deal with this problem, list wise approaches have been proposed to learn a ranking function by adopting separate lists as samples. The loss function is formulated on the predicted list and the ground-truth list.

In this paper, a novel framework is proposed for image re-ranking. Instead of manually defining a universal concept dictionary, it learns different semantic spaces for different query keywords individually and automatically. The semantic space related to the images to be re-ranked can be significantly narrowed down by the query keyword provided by the user. For example, if the query keyword is “apple,” the concepts of “mountain” and “Paris” are irrelevant and should be excluded. Instead, the concepts of "computer" and "fruit" will be used as dimensions to learn the semantic space related to "apple." The query-specific semantic spaces can more accurately model the images to be re-ranked, since they have excluded other potentially unlimited number of irrelevant concepts, which serve only as noise and deteriorate the re-ranking performance on both accuracy and computational cost. The visual and textual features of images are then projected into their related semantic spaces to get semantic signatures. At the online stage, images are re-ranked by comparing their semantic signatures obtained from the semantic space of the query keyword. The semantic correlation between concepts is explored and incorporated when computing the similarity of semantic signatures.

VII.CONCLUSION

Growth in content-based retrieval has been unquestionably rapid. In the recent years, more than 200 content-based retrieval systems have been developed, the majority of which are based on low level features. In particular, they can be classified into two main categories: 1) those that perform semantics mining based on the analysis of textual information associated to images, such as annotations, assigned keywords, captions, alternative (alt) text in html pages or surrounding text, and 2) those that are based on the extraction of low-level visual features such as color, texture in order to perform alignment, classification, browsing, searching, summarization, etc. in image collections. Methods of the first category depend on laborious annotation, while the latter methods usually cannot effectively capture semantics. The similarity measures between visual features do not necessarily match human perception and, thus, retrieval results of low-level approaches are generally unsatisfactory and often unpredictable.

WEB-SCALE image search engines mostly use keywords as queries and rely on surrounding text to search images. They suffer from the ambiguity of query keywords, because it is hard for users to accurately describe the visual content of target images only using keywords. For example, using "apple" as a query keyword, the retrieved images belong to different categories (also called concepts in this paper), such as "red apple," "apple logo," and "apple laptop."

This is the most common form of text search on the Web. Most search engines do their text query and retrieval using keywords. The keywords based searches they usually provide results from blogs or other discussion boards. The user cannot have a satisfaction with these results due to lack of trusts on blogs etc. low precision and high recall rate. In early search engine that offered disambiguation to search terms. User intention identification plays an important role in the intelligent semantic search engine.

REFERENCES

[1] Jun Yu,, "Learning to Rank Using User Clicks and Visual Features for Image Retrieval," *Synth. Lect. Cybernetics. Technol.*, vol. 45, no. 4, pp. 1–113, APRIL 2015.

[2] S. Robertson and S. Walker, "Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval," in *Proc. Annu. Int. ACM SIGIR Conf. Res. Dev. Inf. Retrieval*, 1994, pp. 232–241.

[3] J. Xu and H. Li, "AdaRank: A boosting algorithm for information retrieval," in *Proc. Int. ACM SIGIR Conf. Res. Dev. Inf. Retrieval*, 2007, pp. 391–398.

[4] D. Cossock and T. Zhang, "Statistical analysis of Bayes optimal subset ranking," *IEEE Trans. Inf. Theory*, vol. 54, no. 11, pp. 5140–5154, Nov. 2008.

[5] T. Liu, "Learning to rank for information retrieval," *Found. Trends Inf. Retrieval*, vol. 3, no. 3, pp. 225–331, 2009.

[6] C. Burges, "From RankNet to LambdaRank to LambdaMART: An overview," Microsoft Res., Tech. Rep. MSR-TR-2010-82, 2010.

[7] K. Jarvelin and J. Kekalainen, "Cumulated gain-based evaluation of IR techniques," *ACM Trans. Inf. Syst.*, vol. 20, no. 4, pp. 422–446, 2002.

[8] R. Herbrich, T. Graepel, and K. Obermayer, "Large margin rank boundaries for ordinal regression," in *Advances in Large Margin Classifiers*. Cambridge, MA, USA: MIT Press, 2000, pp. 115–132.

[9] J. Ye, J.-H. Chow, J. Chen, and Z. Zheng, "Stochastic gradient boosted distributed decision trees," in *Proc. ACM Conf. Inf. Knowl. Manag.*, Hong Kong, 2009, pp. 2061–2064.

[10] Y. Cao *et al.*, "Adapting ranking SVM to document retrieval," in *Proc. SIGIR*, Seattle, WA, USA, 2006, pp. 186–193.

[11] Z. Cao, T. Qin, T.-Y. Liu, M.-F. Tsai, and H. Li, "Learning to rank: From pairwise approach to listwise approach," in *Proc. Int. Conf. Mach. Learn.*, Corvallis, OR, USA, 2007, pp. 129–136.

[12] F. Xia, T. Y. Liu, J. Wang, W. Zhang, and H. Li, "Listwise approach to learning to rank: Theory and algorithm," in *Proc. Int. Conf. Mach. Learn.*, 2008, pp. 1192–1199.

[13] X. Hua and G. Qi, "Online multi-label active annotation: Towards largescale content-based video search," in *Proc. ACM Int. Conf. Multimedia*, 2008, pp. 141–150.

[14] W. Hsu, L. Kennedy, and S.-F. Chang, "Video search reranking via information bottleneck principle," in *Proc. Annu. ACM Int. Conf. Multimedia*, Santa Barbara, CA, USA, 2006, pp. 35–44.

[15] X. Tian *et al.*, "Bayesian video search reranking," in *Proc. ACM Int. Conf. Multimedia*, Vancouver, BC, Canada, 2008, pp. 131–140.

[16] R. Yan, A. Hauptmann, and R. Jin, "Multimedia search with pseudorelevance feedback," in *Proc. CIVR*, Urbana-Champaign, IL, USA, 2003, pp. 238–247.

[17] W. Hsu, L. Kennedy, and S. Chang, "Video search reranking through random walk over document-level context graph," in *Proc. SIGMM*, Augsburg, Germany, 2007, pp. 971–980.