Query Difficulty Prediction for Web Image Search

Xinmei Tian, Yijuan Lu, *Member, IEEE*, and Linjun Yang, *Member, IEEE*

***Abstract*—Image search plays an important role in our daily life. Given a query, the image search engine is to retrieve images related to it. However, different queries have different search difficulty levels. For some queries, they are easy to be retrieved (the search engine can return very good search results). While for others, they are difficult (the search results are very unsatisfactory). Thus, it is desirable to identify those "difficult" queries in order to handle them properly. Query difficulty prediction (QDP) is an attempt to predict the quality of the search result for a query over a given collection. QDP problem has been investigated for many years in text document retrieval, and its importance has been recognized in the information retrieval (IR) community. However, little effort has been conducted on the image query difficulty prediction problem for image search. Compared with QDP in document retrieval, QDP in image search is more challenging due to the noise of textual features and the well-known semantic gap of visual features. This paper aims to investigate the QDP problem in Web image search. A novel method is proposed to automatically predict the quality of image search results for an arbitrary query. This model is built based on a set of valuable features that are designed by exploring the visual characteristic of images in the search results. The experiments on two real image search datasets demonstrate the effectiveness of the proposed query difficulty prediction method. Two applications, including optimal image search engine selection and search results merging, are presented to show the promising applicability of QDP.**

***Index Terms*—Image retrieval, image search quality, query difficulty prediction (QDP).**

I. Introduction

With the explosive growth of online image collection, image retrieval plays an important role in our daily life. Much research work has been conducted to search relevant images for a given query term, with emphases on various aspects, e.g., effective low-level visual feature and high-level feature extraction [1]–[5] and ranking and reranking algorithms design [6]–[11].

Manuscript received June 25, 2011; revised November 02, 2011; accepted November 09, 2011. Date of publication November 29, 2011; date of current version July 13, 2012. This work is supported in part by start-up funding from the University of Science and Technology of China to X. Tian, in part by the Research Enhancement Program (REP), and start-up funding from the Texas State University and NSF CRI 1058724 to Y. Lu. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Jinhui Tang.

X. Tian is with the Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei 230027, China (e-mail: xinmei@ustc.edu.cn).

Y. Lu is with the Department of Computer Science, Texas State University, San Marcos, TX 78666 USA (e-mail: yl12@txstate.edu).

L. Yang is with Microsoft Research Asia, Beijing 100080, China (e-mail: linjuny@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/TMM.2011.2177647

Despite the extensive efforts that have been made to improve overall search quality, image search engines/systems still suffer from a radical variance in performance over different queries. Even for search systems that perform very well on average, the quality of their search results may be poor for some queries. For example, Fig. 1 shows the top-10-ranked images of three queries returned by an image search engine. It shows that this search engine performs very well on query "*opera garnier*" with 10 out of 10 relevant images returned, but only 1 out of 10 relevant image returned for query "*demonstration*". This phenomenon of search quality variance over queries is caused by many factors, e.g., the query itself, index approaches, ranking/reranking algorithms, and image collections. To improve the search quality, it is desirable to identify those "difficult" queries in order to handle them properly. For example, for users, they can conduct query expansion/reformulation for these difficult queries and redo the search process to improve their search quality. For the search engines, they can use alternative ranking strategies for these difficult queries or expand the image collection by adding more images related to them.

Query difficulty prediction (QDP) is an attempt to predict the quality of search results returned by a given system for the query over a given collection, in the absence of relevance judgments and without user feedback [12], [13]. Usually, the search quality is measured via average precision, etc. Query difficulty prediction in text document retrieval has been well explored for many years, and a lot of valuable methods have been proposed [12], [14]–[17]. However, in image search, little research has been conducted on image query difficulty prediction. The query difficulty prediction in image search and text document search is essentially different. Compared with QDP in text document retrieval, QDP in image search is more challenging. In text document search, both the query and documents are in the textual domain. Many QDP methods are designed to explore the text distribution relationship between query and returned documents. However, in image search, queries are textual, but returned images are visual. This domain difference essentially makes it more challenging for query difficulty prediction in image search. Besides, in image retrieval, the associated textual information (surrounding text, image URL, etc.) is noisy and insufficient to describe the rich content of images comprehensively and substantially. Visual features are the essential description of the images, but it suffers from the well-known semantic gap [18].

In this paper, we target at query difficulty prediction for the Web image search task. We propose a novel model to automatically predict the query difficulty for any given query through the machine learning approach. First, by analyzing the visual distribution characteristics of good and bad search results of a set of training queries, we derive several valuable features that are related to image search quality. Then, through a learning process, the latent relationship between our derived features and the inherent query difficulty is mined and an query difficulty predic-

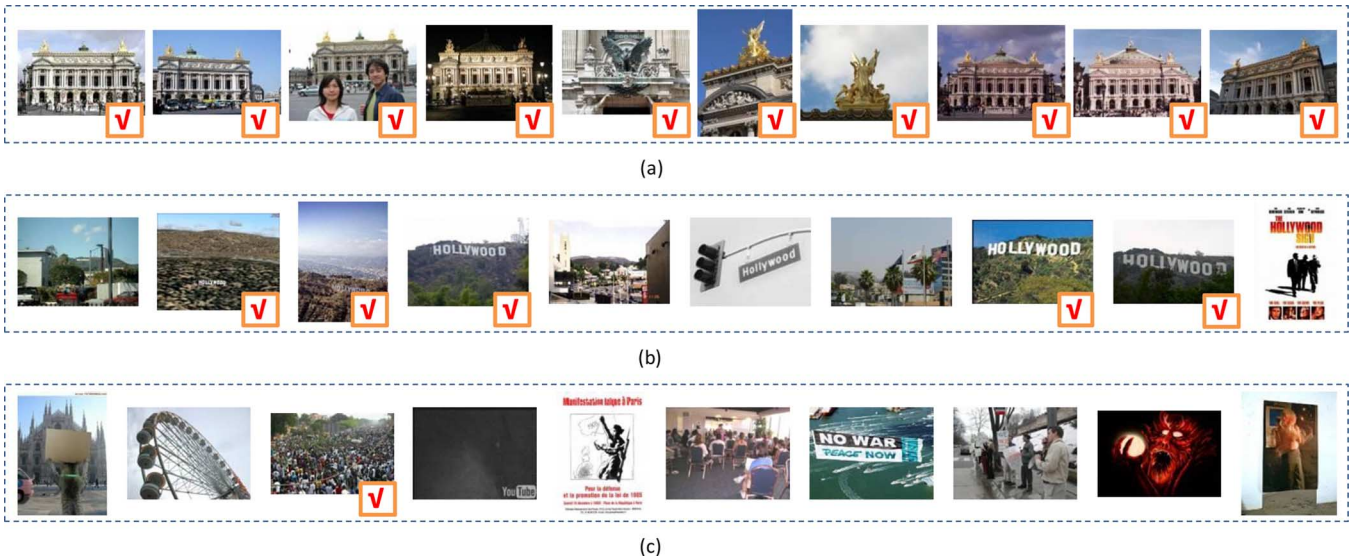


Fig. 1. Top-10-ranked images for three queries ("*opera garnier*", "*Hollywood sign*", and "*demonstration*") using a text-based image search engine, ordered left to right. Query-relevant images are marked by red "√". It shows that this image search engine suffers from a radical variance in search performance over different queries.

tion model is built. Finally, this model will be applied to any new coming query to quantitatively measure its query difficulty.

To the best of our knowledge, it is the first attempt that automatically predicts the query difficulty for Web image search results. The main contributions introduced in this paper are summarized as follows.

- We quantitatively study and formulate the query difficulty prediction problem for the Web image search. We propose a set of valuable features and a machine learning based method to automatically predict the quality of image search results.
- Our proposed approach shows its effectiveness in query difficulty prediction and promising application for optimal search engine selection and search result merging.
- Most of conventional QDP methods usually only predict a value for each query, to indicate the degree of its difficulty. This indicator cannot reflect their exact search performance well. Our proposed model can successfully solve this problem, whose output is the estimation of real search performance, instead of only an indicator. The mean absolute error [19], [20] criteria is introduced to evaluate the capacity of precise performance prediction.

The remainder of this paper is organized as follows. In Section II, we will give an overview of related work. Then in Section III, we will define the query difficulty prediction problem and detail the proposed QDP related features. Experimental results, including applications in image search engine selection and search result merging, are presented in Sections IV and V, respectively, followed by the conclusion in Section VI.

II. Related Work

Query difficulty prediction in text document retrieval has been investigated for many years, and its importance has been recognized in the information retrieval (IR) community [12], [14], [15], [17], [21]–[24]. However, there is little research work for it in image retrieval. In this section, we focus on introducing the related work of query difficulty prediction in text document retrieval, which can be categorized into two groups: pre-retrieval prediction and post-retrieval prediction.

In pre-retrieval prediction, it attempts to evaluate the search difficulty before the retrieval step [22]–[24]. It mainly relies on statistics of query terms over document collection. He and Ounis [22] proposed several pre-retrieval predictors by considering the intrinsic statistical features of queries, including query length, standard deviation of the inverse document frequency (idf) of terms in query, query scope, and simplified clarity score (SCS). Kwok *et al.* [23] employed the support vector regression to train a query difficulty prediction model with simple features, such as log document frequency and query term frequency. Imran and Sharan [24] proposed two pre-retrieval query difficulty predictors based on the co-occurrence information among query terms. They assumed that higher co-occurrence of query terms means more information conveyed, which leads to easier query or lower query difficulty level. Pre-retrieval prediction has the advantage of efficiency saving relevance scores computation in retrieval process. However, due to the absence of the important retrieval list, pre-retrieval prediction usually does not perform as well as post-retrieval ones. As reported in [22], the post-retrieval clarity score [14] achieves much higher correlations than pre-retrieval SCS [22].

In post-retrieval prediction, the retrieval step is conducted first to return a retrieval list. Then, by analyzing the statistical information within three sources—query, documents in the retrieval list, and the whole document collection—various post-retrieval query difficulty predictors are proposed [12]–[15], [17], [21], [25]–[27]. According to their basic assumptions, we can further group them into three categories, i.e., clarity-based, stability-based, and coherence-based.

Clarity-based methods usually predict query difficulty by investigating the distribution difference between the retrieved documents and the whole document collection. Cronen–Townsend *et al.* [14] proposed the CS, which measures the ambiguity of a query through the Kullback–Leibler (KL)

divergence [28] between the language models created from top-retrieved documents and all documents in the collection. Encouraged by the success of the clarity score, more similar research work has been proposed. For example, Amati *et al.* [25] proposed to use the KL-divergence between the query term frequency in the top retrieved documents and their frequency in the whole collection. Hauff *et al.* [17] proposed improved clarity score to solve the parameter sensitivity problem in CS. Carmel *et al.* [27] proposed the use of distances between queries, relevant documents, and collections as query difficulty predictors. The weighted information gain approach proposed by Zhou and Croft [16] indicates the query difficulty by measuring the information change from an average returned document to the actual retrieval results. It is based on the hypothesis that high-quality retrieval should be much more effective than just returning the average document.

Stability-based methods predict query difficulty mainly based on the investigation of the stability of several retrieval results obtained from different ways. For example, several works [12], [13], [26] predict query difficulty by measuring the stability of retrieval results in the presence of perturbations of the query, collection, and scoring function, respectively. Specifically, Yom-Tov *et al.* [12] estimated the search results quality by measuring the agreement between the top returned results of full query and the top returned result of each of the query terms. Zhou and Croft [26] proposed the ranking robustness, which is defined as the similarity between ranked lists generated from the original collection and the corrupted collection. Aslam and Pavlu [13] first obtained numbers of retrieval lists by using different scoring functions and then mapped each ranked list of documents to a probability distribution. Then, the Jensen–Shannon divergence [29] among these distributions is adopted as a query difficulty predictor. Zhou and Croft [16] first constructed a new query from the top returned documents of the original query. The original query and the new query generated two different ranked lists of their corresponding returned documents. The overlap of documents in these two ranked lists is used for query difficulty prediction.

Coherence-based methods assume that the tightness of the top returned documents can indicate the search quality. He *et al.* [30] proposed the coherence score indicator, which measures the portion of coherent document pairs in the top returned document set. A pair of documents is defined as "coherent" if their similarity exceeds a given threshold. Rudinac *et al.* [21] also exploited the coherence of the top-ranked documents returned by the unexpanded query and several query expansion alternatives, to select the best query expansion for spoken content retrieval.

All the above related work is designed for query difficulty prediction for text document retrieval. Little research has been conducted on image query difficulty prediction for image retrieval. Xing *et al.* [31] used the textual features associated with images (URL, surrounding text, etc.) to predict whether a query is difficult to be represented by images or not. However, this work does not investigate image visual features and does not measure real image search performance. It only classifies queries into two categories "easy" or "hard," relying on simple textual features. Li *et al.* [32] estimated the retrieval difficulty of a given query image by analyzing the CS [14], spatial verification, and appearance consistency between the query image and the retrieved top-ranked ones.

III. The Proposed Approach

For a query q, the search engine can generate a search result $l = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$, where $\mathbf{x}_i$ is the ith-ranked image. The query difficulty is inversely related to the search performance. The query with good retrieval performance means it is not difficult to retrieve (has low query difficulty). Therefore, this paper proposes to predict the query difficulty by approximating the search performance. In other words, given a query q, the query difficulty prediction is to estimate the quality (or performance) of its search result l. We denote the estimated performance prediction as $y'(q)$ and its ground-truth performance as $y(q)$. Here, search performance/quality can be measured via various criteria, e.g., the commonly used measurements in information retrieval, such as precision, recall, average precision (AP) [33], and normalized discounted cumulated gain (NDCG) [34].

In this paper, we formulate the query difficulty prediction as a regression problem. We first explore a set of valuable features to reflect the characteristics of the search results and then use a regression model to capture the dependency between those features and their ground-truth search performance. Specifically, for query q, we extract a QDP related feature vector $\psi(q)$. Our aim is to learn a regression function $f(\psi(q)) = \mathbf{w}^{\mathrm{T}}\psi(q)$ from a set of training samples

$$\{(\psi(q^{(1)}), y(q^{(1)})), (\psi(q^{(2)}), y(q^{(2)})), \ldots, (\psi(q^{(m)}), y(q^{(m)}))\}.$$

The $\mathbf{w}$ is the weighting coefficient vector. In this paper, we adopt the powerful ϵ-support vector regression [35] for the model learning. It is formulated as

$$\begin{aligned} \min_{\mathbf{w},b,\xi,\xi^*} \quad & \frac{1}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w} + C\sum_{i=1}^{m}\xi_i + C\sum_{i=1}^{m}\xi_i^* \\ \text{subject to} \quad & \mathbf{w}^{\mathrm{T}}\psi(q^{(i)}) + b - y(q^{(i)}) \le \epsilon + \xi_i \\ & y(q^{(i)}) - \mathbf{w}^{\mathrm{T}}\psi(q^{(i)}) - b \le \epsilon + \xi_i^* \\ & \xi_i, \xi_i^* \ge 0, \quad i = 1, \ldots, m. \end{aligned} \tag{1}$$

where ξ and ξ^* are the slack variables, and $C > 0$ controls the tradeoff between model complexity and training error.

The regression model $f(\cdot)$ can be derived by solving problem (1). Then this model can be applied to any testing query q' to get its predicted performance $y(q') = f(\psi(q')) = \mathbf{w}^{\mathrm{T}}\psi(q')$.

In the query difficulty prediction model $f(\cdot)$, the QDP related feature vector $\psi(q)$ plays a crucial role. It is nontrivial to design such a feature to capture the query difficulty characteristics. By analyzing the search results, we propose a set of lightweight features from several aspects, including visual clarity score, coherence score, representativeness score, and visual similarity distribution feature.

A. Visual Clarity Score

Since Cronen–Townsend *et al.* [14] first proposed the CS, more research work has been proposed in a similar way with a clarity score technique encouraged by its success [16], [17], [25], [27]. It assumes that a better retrieval result is more distinctive from the whole dataset and therefore has a larger clarity score. The clarity score measures the distribution difference

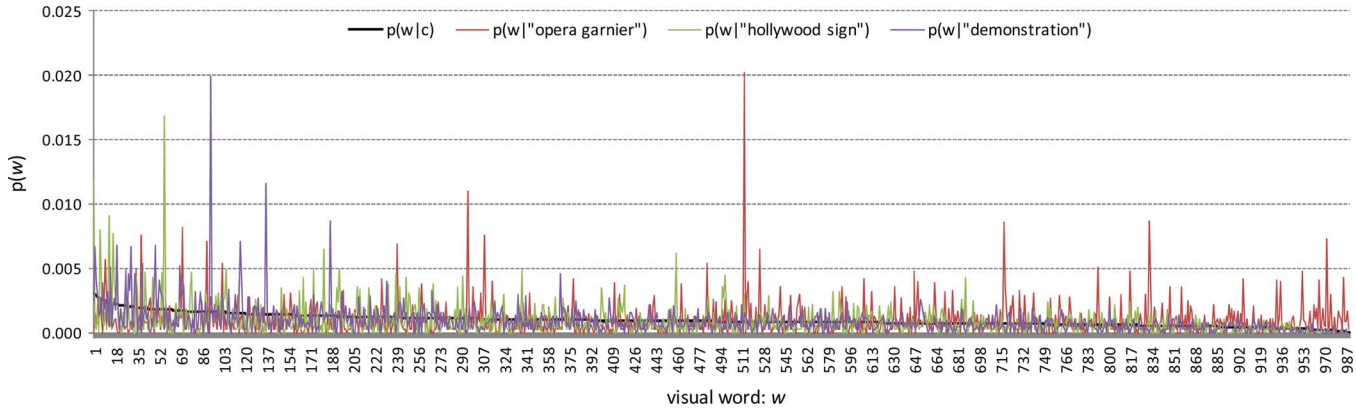


Fig. 2. The collection language model $P(w|c)$ and the query language models for the three example queries in Fig. 1. The visual clarity scores for them are VCS("*opera garnier*") $= 0.77$, VCS("*Hollywood sign*") $= 0.60$, and VCS("*demonstration*") $= 0.47$, respectively. It shows that the visual clarity score reflects the search quality well. (Better view in color version.)

between the top-retrieved documents and the whole document collection. Specifically, it calculates the query language model from top returned documents and the collection language model from all documents in the dataset. The CS is defined as the KL divergence [28] between the query language model and collection language model.

Inspired by this, we propose a visual clarity score (VCS) for the image search query difficulty prediction task. This VCS measures the distribution difference between the top returned images and the whole image collection. The difficulty in calculating CS for the image search is that the query and images are in different domains: textual and visual. To solve this domain gap problem, we propose to use the popular visual bag-of-word image representation. We first extract the scale-invariant feature transform (SIFT) [1] features for each image with dense sampling, and use k-means [36] to build a vocabulary/dictionary V with $|V|$ visual words. Then, each SIFT descriptor is quantized into its corresponding visual word. With each image represented as a sequence of visual words, we can estimate the query language model $P(w|q)$ and collection language model $P(w|c)$ for visual words as follows.

For query language model, it is defined as

$$P(w|q) = \sum_{\mathbf{x}\in\mathcal{R}} P(w|\mathbf{x})P(\mathbf{x}|q) \tag{2}$$

where $w \in V$ is a visual word, and $\mathcal{R}$ is a set of images returned for query q. $P(w|\mathbf{x})$ is defined as the term frequency of the word w in image $\mathbf{x}$.

For $P(\mathbf{x}|q)$,

$$P(\mathbf{x}|q) \propto P(q|\mathbf{x})P(\mathbf{x}). \tag{3}$$

Since each image $\mathbf{x}$ has an equal prior $P(\mathbf{x})$, we only need to estimate the likelihood $P(q|\mathbf{x})$. In text document retrieval, $P(q|\mathbf{x})$ is defined as the product of the term frequency of each query term in image $\mathbf{x}$,

$$P(q|\mathbf{x}) = \prod_{q_i\in q} P(q_i|\mathbf{x}). \tag{4}$$

However, in our problem, the query q is in textual domain and cannot be represented by visual words.

Since the $P(q|\mathbf{x})$ denotes the possibility of image $\mathbf{x}$ to be relevant to q, we can estimate it by leveraging the text-based search result. We define $P(q|\mathbf{x})$ as 1 if image $\mathbf{x}$ appears in the top-T returned images for query q, else 0 if image $\mathbf{x}$ does not appear in the top-T returned images for query q,

$$P(q|\mathbf{x}) = \begin{cases} 1, & \text{if Rank } (\mathbf{x}) \le T \\ 0, & \text{else} \end{cases} \tag{5}$$

where $\text{Rank}(\mathbf{x})$ denotes the rank of $\mathbf{x}$ in l. In other words, the query language model is estimated over the top-T-ranked images, which are assumed to be pseudorelevant to the query q according to the widely used pseudorelevance feedback assumption [37], [38].

For the collection language model, $P(w|c)$ is defined as the term frequency of word w over all images in collection c. Then, the visual clarity score is defined as the KL divergence [28] between the language model and collection model

$$\begin{aligned} \text{VCS} &= D_{\text{KL}}(P(w|q)|P(q|c)) \\ &= \sum_{w\in V} P(w|q)\log\frac{P(w|q)}{P(w|c)}. \end{aligned} \tag{6}$$

Fig. 2 shows the collection language model and the query language models for the three example queries in Fig. 1. The clarity scores for them are VCS("*opera garnier*") $= 0.77$, VCS("*Hollywood sign*") $= 0.60$, and VCS("*demonstration*") $= 0.47$, respectively. It demonstrates that the visual clarity score reflects the search quality well.

B. Coherence Score

For a good image search result, the top-ranked images in this result must contain many query relevant images. Since relevant images share common visual patterns, they are more visually similar than query irrelevant images. According to this observation, we can measure the coherence character of the top-ranked images to indicate the search quality, termed *coherence score* (CoS). The effectiveness of coherency score in text document search query difficulty prediction has been demonstrated in [21] and [30]. In this paper, we will investigate its capacity for image search query difficulty prediction.

We examine the visual similarity between any image pair within top-T-ranked ones and count the numbers of coherent image pairs. The coherent image pairs are those whose visual

similarities are larger than certain threshold $\mathrm{Tr}_{\mathrm{sim}}$. The CoS is defined as the ratio of coherent pairs to all image pairs,

$$\mathrm{CoS} = \frac{1}{|T(T-1)|} \sum_{i,j=1,\ldots,T;i\neq j} \delta(\mathbf{x}_i, \mathbf{x}_j). \quad (7)$$

The $\delta(\mathbf{x}_i, \mathbf{x}_j)$ is a binary function defined as

$$\delta(\mathbf{x}_i, \mathbf{x}_j) = \begin{cases} 1, & \text{if sim } (\mathbf{x}_i, \mathbf{x}_j) > \mathrm{Tr}_{\mathrm{sim}} \\ 0, & \text{else.} \end{cases} \quad (8)$$

$\mathrm{sim}(\mathbf{x}_i, \mathbf{x}_j)$ is the visual similarity between images $\mathbf{x}_i$ and $\mathbf{x}_j$. In this paper, each image is represented as a histogram of visual words as described in Section III-A. The intersection kernel [39] is adopted for calculating the similarity between two histograms. The threshold $\mathrm{Tr}_{\mathrm{sim}}$ is defined as that 80% of image pairs in the dataset have smaller visual similarity than this value.

The coherence scores of the three example queries in Fig. 1 are CoS("*opera garnier*") $= 0.53$, CoS("*Hollywood sign*") $= 0.47$, and CoS("*demonstration*") $= 0.09$, respectively. It shows that the CoS also matches the search quality well.

C. Representativeness Score

As widely used as a basic underlying assumption in visual reranking [6], [40], it is assumed that the representative images in the search results are more likely to be query relevant. In other words, a better search result consists of more representative images as top ones. According to this assumption, the representativeness of the top-ranked images can serve as the role for quantifying the search quality. For representative score (RS), we first calculate the density for each image via kernel density estimation (KDE) [41],

$$p_{\mathbf{x}_i} = \frac{1}{|\mathcal{N}(\mathbf{x}_i)|} \sum_{\mathbf{x}_j \in \mathcal{N}(\mathbf{x}_i)} k(\mathbf{x}_i - \mathbf{x}_j) \quad (9)$$

where $\mathcal{N}(\mathbf{x}_i)$ is the set of neighbors of image $\mathbf{x}_i$ among the N images $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ and $k(\mathbf{x})$ is a kernel function that satisfies both $k(\mathbf{x}) > 0$ and $\int k(\mathbf{x}) d\mathbf{x} = 1$.

The representativeness score is defined as the mean of the density of top-T-ranked images in l,

$$\mathrm{RS} = \frac{1}{T} \sum_{i=1}^{T} p_{\mathbf{x}_i}. \quad (10)$$

The representativeness scores for the three example queries in Fig. 1 are RS("*opera garnier*") $= 0.25$, RS("*Hollywood sign*") $= 0.20$, and RS("*demonstration*") $= 0.12$, respectively. We can see that the representativeness score is also consistent with the search quality.

D. Visual Similarity Distribution of Top-Ranked Images

The above three features measure the overall characteristics of the top-T-ranked images. Besides those overall measurements, the following features are designed to exploit them in fine granularity as a complementation.

For query q, given a ranking result l, a visual similarity matrix $\mathbf{M} \in \mathbb{R}^{N\times N}$ can be obtained by calculating pairwise image similarity. The (i, j) element m_{ij} in $\mathbf{M}$ denotes the visual similarity between the ith and jth-ranked images. The visual similarity is in range [0, 1], and we equally divide it into H-bins.

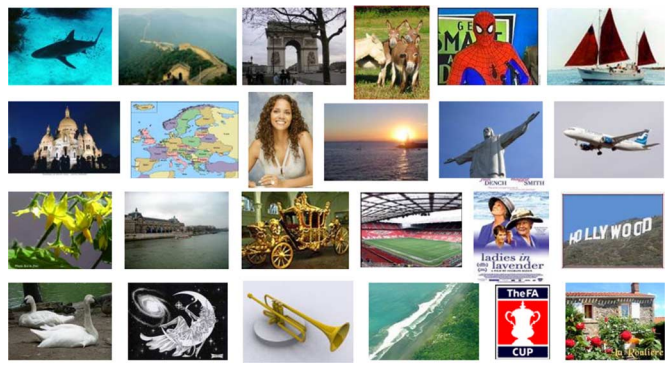


Fig. 3. Example images in Web353 dataset.

Then, the $T \times T$ similarity matrix of top-T-ranked images can be quantified into a H-bin histogram by mapping them into their corresponding bins. We denote this visual similarity distribution histogram feature as VSDH,

$$\mathrm{VSDH}(h) = \frac{1}{T^2} |\{(i,j) | i \leq T, j \leq T, m_{ij} \in h\text{th bin}\}|, \quad h = 1, \ldots, H. \quad (11)$$

With the VCS, CoS, RS, and VSDH, the final QDP related feature vector $\psi(q)$ can be derived by concatenating these four individual features to a long feature vector. Then, the QDP model $f(\cdot)$ will be trained on a set of queries according to (1) to mine the dependency between $\psi(q)$ and the ground-truth search quality $y(q)$.

IV. Experiments

In this section, we investigate the effectiveness of the proposed query difficulty prediction method by applying it on a real Web image search dataset. For each query q, a text-based image search ranking list is generated by the image search engine. Better search quality means less query difficulty. We estimate the performance of text-based image search result for each query, and then compare the predicted $y'(q)$ with its ground-truth performance $y(q)$.

A. Dataset

In order to demonstrate the capacity of the proposed query difficult prediction method, we conduct experiments on a large public Web image search dataset "Web353". This dataset is collected by Krapac *et al.* [7]. They selected 353 queries from the most frequent terms searched by the users on a popular image search engine.[1] For each query, the top-ranked images found by the search engine are collected. There are about 200 images returned on average for each query, and this dataset contains 71 478 images in total. Fig. 3 shows some example images in this dataset. The 353 queries are diverse in topics, including landmarks ("*Eiffel Tower*"), people (movie, sports, and singer stars), design (painting ("*Guernica*"), logo ("*logo nba*"), flag ("*France flag*"), object (vehicle ("*bicycle*"), building ("*skyscraper*"), instrument ("*violin*")), animal ("*dolphin*"), plant ("*leeks*"), event ("*demonstration*"), etc. Fig. 1 shows the top-10–ranked images for three example queries ("*opera garnier*", "*Hollywood sign*," and "*demonstration*"). The ground-truth relevance label for every image is evaluated

[1]Available [online]: http://www.exalead.com/search/image

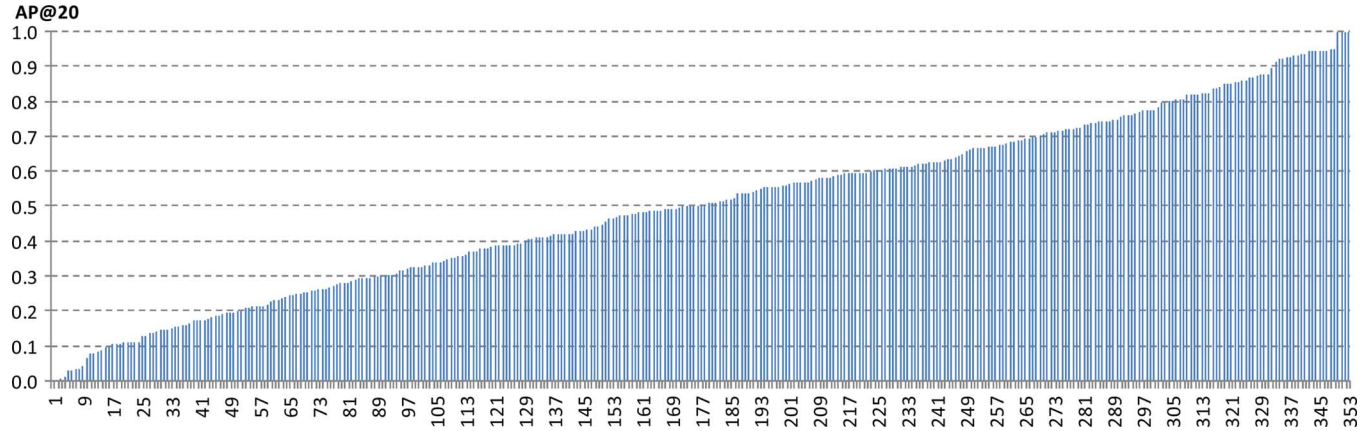


Fig. 4. The AP@20 on each of the 353 queries. Here, queries are sorted according to their AP@20 in ascending order for better view. This figure shows that the performance of the image search engine varies largely over different queries.

on two levels, "relevant" or "irrelevant". In this dataset, there are 43.86% images labeled as relevant.

B. Experimental Setup

For query q, given the text-based image search result returned by the image search engine, its ground-truth performance $y(q)$ is measured via the commonly used noninterpolated average precision (AP) [33] in information retrieval. The AP averages the precision values obtained when each relevant image occurs. The AP of top-T-ranked images AP@T is calculated as

$$\text{AP@}T = \frac{1}{Z_T}\sum_{i=1}^{T}[\text{precision}(i) \times \text{rel}(i)] \tag{12}$$

where $\text{rel}(i)$ is the binary function on the relevance of the ith-ranked image with "1" for relevant and "0" for irrelevant. The $\text{precision}(i)$ is the precision of top-i-ranked images,

$$\text{precision}(i) = \frac{1}{i}\sum_{j=1}^{i}\text{rel}(j). \tag{13}$$

The Z_T is a normalization constant that is chosen to guarantee that AP@$T = 1$ for the perfect ranking result.

The ground-truth search performance in terms of AP@20 for each of the 353 queries in Web353 is illustrated in Fig. 4. In this figure, the queries are sorted in ascending order of AP for better view. It shows that the image search engine suffers a radical variance in performance over different queries.

The density and visual similarity are calculated based on images' visual representation. In this paper, the bag-of-visual word histogram is adopted, as described in Section III-A. The SIFT [1] local descriptors are extracted for each image on a dense grid. Then, a codebook is generated by clustering all the local descriptors into 1000 groups [42]. By quantizing local descriptors into visual words, each image is represented as a histogram. The intersection kernel [39] is adopted for calculating the similarity between two histograms. The H, number of bins in VSDH, is empirically set as 50.

For the ϵ-support vector regression [35] model, we employ an implementation of SVR (LIBSVM) developed by Chang and Lin [43] with radial basis function kernel. The γ is set according to the reciprocal of averaged pairwised distance over training samples. The C and ϵ are empirically set as 10 and 0.1, respectively. We adopt the commonly used leave-one-out [27], [44] for model training. Each time, we train the SVR model on 352 queries and then test this model on the left one query. Repeat it 353 times to ensure that each query has been used exactly once as the test query.

For the baseline, although the work in [31] discusses the image search query difficulty prediction problem, their method is restricted to noun word queries and only determines the noun word is "easy" or "hard" to be represented by images, instead of predicting the real image search quality with a given retrieval list. Since there is no other work on image search query difficulty prediction and it is also a learning-based and post-retrieval query difficulty predictor, we implement the text document search query difficulty method proposed in [15] as a baseline for comparison, by treating the associated textual descriptions of each image as a document. According to [15], we extract textual features for each image from its associated textual descriptions, e.g., URL, surrounding texts, etc. We denote this method as "TextQD" here since this method relies on the associated textual features of the images only.

C. Experimental Results

For each query q, we estimate its performance $y'(q)$ to approximate its ground-truth performance $y(q)$. To verity the effectiveness of our model, we evaluate it from the following three aspects.

1. *Correlation Coefficient:* For the 353 queries, we have their ground-truth performance vector $\mathbf{y} = [y(q^{(1)}), y(q^{(2)}), \ldots, y(q^{(353)})]^{\mathrm{T}}$ and the one predicted by the proposed model $\mathbf{y}' = [y'(q^{(1)}), y'(q^{(2)}), \ldots, y'(q^{(353)})]^{\mathrm{T}}$. All query difficulty prediction work evaluates their methods by measuring the correlation between $\mathbf{y}$ and $\mathbf{y}'$. The commonly used correlation measurements include the Pearson's r liner correlation [45] adopted in [13], [16], [17], [22], [24], [26], and [27], nonparametric rank correlation Kendall's τ [46] adopted in [12], [13], [17], [23], [24], [26], and [27], and Spearman's ρ [47] adopted in [14], [15], [22], [24], and [30]. The correlation coefficients of the above three all range between $[-1.0, 1.0]$, where -1.0 means perfect negative correlation and 1.0 means perfect positive correlation. For a better understanding, Fig. 5 gives an illustration of Pearson's r at different levels.

In this paper, all three correlation tests are adopted. We evaluate the proposed model at several truncation levels for T, i.e., $y(q) = \{\text{AP@}T, T = 10, 20, 40, 60\}$. The choice of T depends

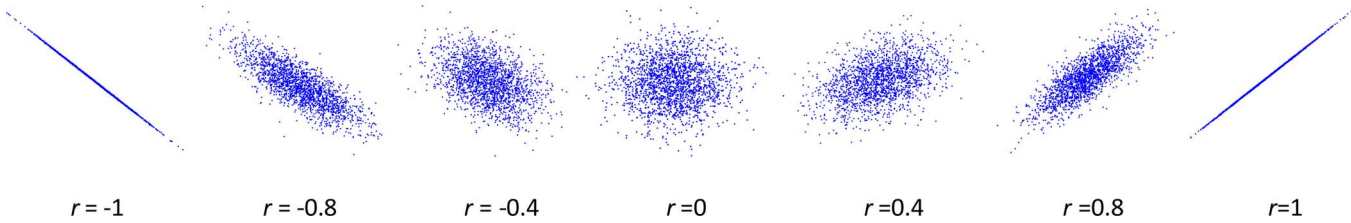


Fig. 5. Illustration of Pearson's linear correlation coefficients r with 2-dimensional toy data.

TABLE I
CORRELATION COEFFICIENTS AND P-VALUES OF QUERY DIFFICULTY PREDICTION

		Kendall's τ (P-value)	Pearson's r (P-value)	Spearman's ρ (P-value)
T=10	TextQD	0.157 (*1.19e-05*)	0.223 (*2.45e-05*)	0.237 (*6.87e-06*)
	Ours	**0.364** (***3.61e-24***)	**0.517** (<***1e-50***)	**0.525** (<***1e-50***)
T=20	TextQD	0.197 (*3.48e-08*)	0.293 (*2.09e-08*)	0.286 (*4.46e-08*)
	Ours	**0.363** (***2.53e-24***)	**0.501** (<***1e-50***)	**0.526** (<***1e-50***)
T=40	TextQD	0.186 (*1.96e-07*)	0.269 (*3.05e-07*)	0.286 (*4.63e-08*)
	Ours	**0.317** (***6.96e-19***)	**0.431** (<***1e-50***)	**0.465** (<***1e-50***)
T=60	TextQD	0.209 (*4.41e-09*)	0.304 (*5.58e-09*)	0.310 (*2.69e-09*)
	Ours	**0.280** (***4.00e-15***)	**0.368** (***8.66e-13***)	**0.403** (***3.22e-15***)

TABLE II
CORRELATION COEFFICIENTS AND P-VALUES OF EACH INDIVIDUAL FEATURE ($T = 20$)

	Kendall's τ (P-value)	Pearson's r (P-value)	Spearman's ρ (P-value)
VCS	0.158 (*8.91e-06*)	0.172 (*1.15e-03*)	0.238 (*6.02e-06*)
CoS	0.306 (*1.59e-17*)	0.438 (<*1e-50*)	0.447 (<*1e-50*)
RS	0.271 (*3.22e-14*)	0.367 (*1.02e-12*)	0.395 (*1.25e-14*)
VSDH	0.314 (*1.49e-18*)	0.445 (<*1e-50*)	0.462 (<*1e-50*)

on the need in real applications. For example, it is very common that most users may only view the images ranked in several top pages. The correlation coefficients and corresponding P-values are given in Table I. It reveals that our method (Ours) outperforms TextQD consistently over all Ts. Our method achieves a strong correlation between our predicted search performance and the ground-truth performance. The P-value is far less than 0.05, which indicates that the correlation between them is statistically significant. The reason why TextQD does not work well is that, in image search, the textual features are not the essential descriptions for the images' content; therefore, a lot of noise (e.g., mismatching between images and surrounding texts) may be introduced.

We also investigate the effectiveness of each of the four features proposed in Section III. The experimental results are presented in Table II. It shows that each of the four features has positive correlation with the query difficulty. Compared with the results in Table I ($T = 20$), the combination of the four features achieves better results than each individual feature. Fig. 6 shows scatter plots of the ground-truth AP (x-axis) with the three scores (a), (b), and (c), the AP predicted by our method using only VSDH feature (d), the baseline method TextQD (e), and our proposed method using all four kinds of features (f). It shows that our method achieves the best correlation with the ground-truth AP.

2. *Classification Accuracy—For Hard and Easy Queries:* Since one of our aims is to detect the "hard" queries for handling them properly later, we further evaluate our method by constructing a hard and easy queries classification problem. We split the 353 queries into two categories, "easy" and "hard", according to their ground-truth search performance. We first get the average performance over all queries avgAP as a threshold. We define queries whose ground-truth performances are larger/smaller than this threshold avgAP as "easy"/"hard" ones. For example, when $T = 20$, the threshold avgAP $= 0.5028$, and there are 175 easy queries and 178 hard queries. With this two-category query splitting, it becomes a two-class classification problem. For each query q, we check our predicted performance $y'(q)$, if $y'(q)$ is larger than avgAP, then q is predicted as "easy", else this query is predicted as "hard". The classification accuracy is defined as

$$\mathrm{AC} = \frac{\sharp\text{Correctly predicted queries}}{\sharp\text{Total queries}} \tag{14}$$

Besides this overall accuracy, we also examine the prediction accuracy on each of the two categories, i.e., $\mathrm{P}_{\mathrm{Easy}}$ and $\mathrm{P}_{\mathrm{Hard}}$. They are defined as

$$\mathrm{P}_{\mathrm{Easy}} = \frac{\sharp\text{Correctly predicted Easy queries}}{\sharp\text{Total Easy queries}} \tag{15}$$

$$\mathrm{P}_{\mathrm{Hard}} = \frac{\sharp\text{Correctly predicted Hard queries}}{\sharp\text{Total Hard queries}}. \tag{16}$$

The experimental results with different Ts are shown in Table III. It shows that our method can classify most queries

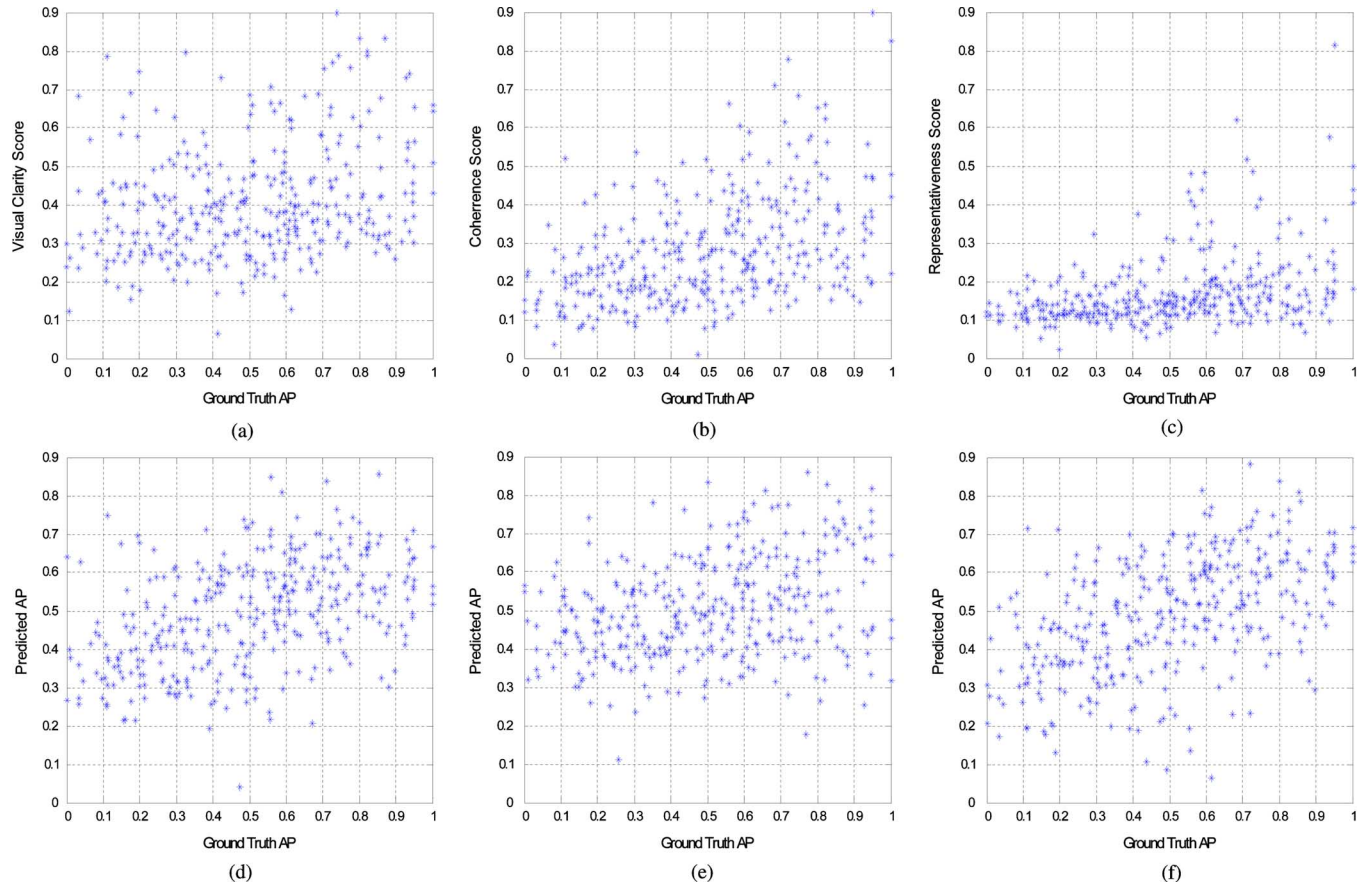


Fig. 6. Scatter plots of the correlation between ground-truth AP@20 (x-axis) with each of the four individual features: (a) visual clarity score; (b) coherence score; (c) representative score; (d) visual similarity distribution histogram; and (e) the baseline method TextQD; as well as (f) the AP predicted by our proposed method using all four kinds of features. Each star (*) in these scatter plots corresponds to a query. It shows that our method achieves the best correlation with the ground-truth AP. (a) VCS: $\tau = 0.158, r = 0.172, \rho = 0.238$. (b) CoS: $\tau = 0.306, r = 0.438, \rho = 0.447$. (c) RS: $\tau = 0.271, r = 0.367, \rho = 0.395$. (d) VSDH: $\tau = 0.314, r = 0.445, \rho = 0.462$. (e) TextQD: $\tau = 0.197, r = 0.293, \rho = 0.286$. (f) Ours: $\tau = 0.363, r = 0.501, \rho = 0.526$

TABLE III
PERFORMANCE OF QUERY DIFFICULTY PREDICTION IN TERMS OF CLASSIFICATION ACCURACY

		Accuracy(%)	P_{Easy}(%)	P_{Hard}(%)
T=10	TextQD	60.06	55.32	65.45
	Ours	**72.80**	**70.74**	**75.15**
T=20	TextQD	61.47	57.87	65.14
	Ours	**72.24**	**71.91**	**72.57**
T=40	TextQD	63.17	60.69	65.56
	Ours	**68.27**	**64.16**	**72.22**
T=60	TextQD	61.76	60.23	63.19
	Ours	**64.87**	**61.99**	**67.58**

correctly and outperforms TextQD consistently. For example, in $T = 20$, 72.24% queries are correctly classified to their categories by our method while only 61.47% queries are correctly classified by TextQD. By further investigating $\mathrm{P_{Easy}}$ and $\mathrm{P_{Hard}}$, we find that our method performs well on both categories. We notice that the performance decreases with T growing. The reason is that, when T grows, more irrelevant images are involved, which bring more noise in QDP related feature extraction.

3. *MAE—Mean Absolute Error:* Besides the correlation coefficient and classification accuracy, we also introduce to use the mean absolute error (MAE) to evaluate our proposed model. MAE is widely used in age estimation [19], [20] and collaborative filtering system [48], [49] problems. Here, we introduce it as a new measurement for query difficulty prediction problem. The MAE is defined as the mean of absolute prediction error $e(q)$ over all queries, i.e., $\mathrm{MAE} = 1/(n_q)\sum_q e(q)$, where $e(q) = |y(q) - y'(q)|$ and n_q is the number of queries. The experimental results in terms of MAE are given in Table IV. It shows that our method achieves less MAE than TextQD method. However, this moderate MAE still has a large space for improvement. Precise prediction of the search performance is challenging and very important in real applications. In the future, we will further investigate this problem and make an effort to reduce the estimation error.

Conclusion: The above three criteria, i.e., *correlation coefficient*, *classification accuracy*, and *MAE*, measure the discriminative power of our query difficulty prediction model at different granularities. The experimental results discussed above demonstrate the capacity of our proposed query difficulty prediction method.

D. Complexity Analysis

The complexity consists of two parts. One is the time complexity for calculating the similarity matrix $\mathbf{M}$, which is $O(VN^2)$, where V is the dimension of the bag-of-visual word histogram and N is the number of images in text-based

Fig. 7. Example images in the 29 queries dataset. Each image represents one query.

TABLE IV
MEAN ABSOLUTE ERROR OF QUERY DIFFICULTY PREDICTION

	T=10	T=20	T=40	T=60
TextQD	0.229	0.198	0.178	0.162
Ours	**0.193**	**0.173**	**0.163**	**0.151**

search result list. The other is the time complexity for calculating the proposed four features in Section III, which is $O(VT + T^2 + \log(N)T^2 + HT^2)$, where T is the truncation level and H is the number of bins in VSDH. In addition to theoretical analysis, we also test the time cost experimentally. The averaged time cost on Web353 dataset is about 0.2 s per query, where $T = 40$, $H = 50$, $V = 1000$, and N is about 200. The algorithm is implemented using MATLAB and runs on a PC with 3.40-GHz Intel Core CPU and 4-GB memory in a single thread. From the theoretical analysis and experimental data discussed above, we can see that the efficiency of our method is acceptable for online applications.

V. APPLICATIONS ON IMAGE SEARCH ENGINE SELECTION AND SEARCH RESULT MERGING

In this section, we investigate the effectiveness of the proposed method by applying it to optimal image search engine selection and search result merging. Specifically, each query q has two ranking lists generated by two search engines—Bing and Google. The search performance of the two search engines varies largely on different queries, as shown in Fig. 8. By estimating the query difficulty for each query on two search engines, we can determine which search engine returns better search results for query q and then present the better one to users.

A. Dataset

We collected a dataset from two popular image search engines, Microsoft's Bing and Google. A total of 29 popular queries were selected from a commercial image search engine query log and popular tags from Flickr. These queries cover a vast range of topics, including scene ("*sky*", "*winter*"), objects ("*funny dog*", "*grape*"), named person ("*George W. Bush*"), etc. We submitted each query to Bing and Google, respectively, and collected at most the top-1000 images returned by each search engine. There are 50 566 images contained in this dataset in total. Some example images in this dataset are shown in Fig. 7. For each query, the relevance labels of returned images are evaluated on two levels: "relevant" or "irrelevant". In this dataset, there are 42.23% images labeled as relevant.

Fig. 8 gives the AP@40 on each query for the two search engines as well as the overall performance MAP (mean AP over all queries). We find that although Bing and Google give comparable MAPs (0.5224 and 0.5236, respectively), their performances on individual queries are quite different. Fig. 8 shows that Bing achieves better performance on about half of the queries (15/29) while Google performs better on the other half (14/29). If we can automatically determine the query difficulty for each query on two search engines, a better performance can be obtained by selecting optimal search engine for each query.

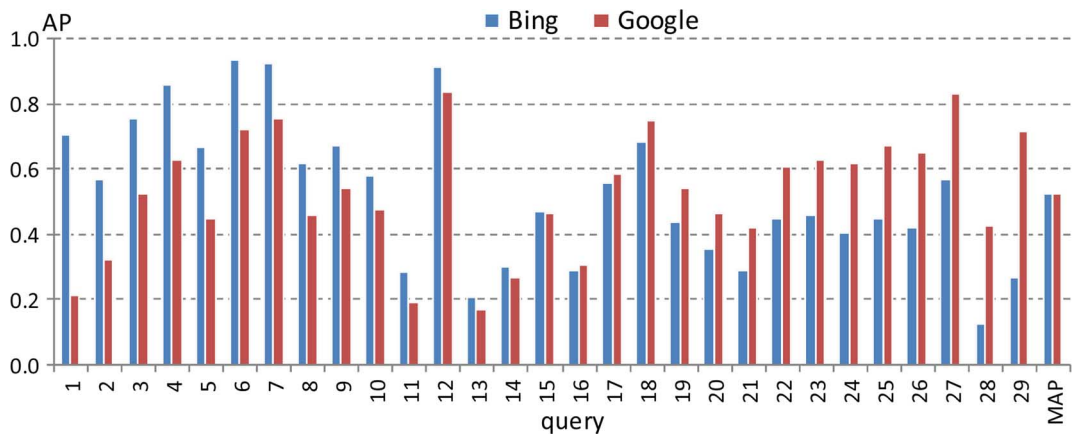


Fig. 8. The AP@40 on each of the 29 queries as well as the mean AP (MAP) over all queries for two search engines—Bing and Google (here, queries are sorted according to their AP difference for better view).

TABLE V
CORRELATION COEFFICIENTS AND ACCURACY IN SEARCH ENGINE SELECTION FROM {BING, GOOGLE}

	Kendall's τ (P-value)	Pearson's r (P-value)	Spearman's ρ (P-value)	Accuracy(%)
T=10	0.3625 (*0.006*)	0.4402 (*0.017*)	0.4804 (*0.008*)	72.41
T=20	0.3153 (*0.016*)	0.4807 (*0.008*)	0.4655 (*0.012*)	75.86
T=40	0.2167 (*0.103*)	0.3498 (*0.063*)	0.3414 (*0.070*)	75.86
T=60	0.3251 (*0.013*)	0.4682 (*0.010*)	0.4931 (*0.007*)	72.41

The experimental setting is the same as that in Section IV. The bag-of-visual words histogram is adopted for image representation, and the leave-one-out method is applied for model training.

B. Evaluation

For each query q, there are two ranking lists: l_{Bing} and l_{Google}. We predict the search quality $y'(l_{\text{Bing}})$ and $y'(l_{\text{Google}})$ respectively. The predicted performance difference is $\delta'_q = y'(l_{\text{Bing}}) - y'(l_{\text{Google}})$. We compare δ'_q with the ground-truth performance difference $\delta_q = y(l_{\text{Bing}}) - y(l_{\text{Google}})$. We also evaluate it from the following two aspects.

1) *Correlation Coefficient*: The Kendall's τ, Pearson's r and Spearman's ρ correlation coefficients between the ground-truth performance difference vector $\Delta = [\delta_1, \delta_2, \cdots, \delta_{29}]^{\mathrm{T}}$ and the one predicted by our model $\Delta = [\delta'_1, \delta'_2, \ldots, \delta'_{29}]^{\mathrm{T}}$.
2) *Prediction Accuracy [defined as in (14)]:* Here, correctly predicted queries are those queries that satisfy $\delta_q \delta'_q > 0$, i.e., the preference relationship between the two ranking lists for query q is correctly predicted.

Search Engine Selection: We also evaluate four different Ts, i.e., $\{10, 20, 40, 60\}$, as in Section IV. The experimental results in terms of correlation coefficients and accuracy are reported in Table V. It shows that good correlation coefficients are achieved, and the P-values are smaller than 0.05 in most cases except $T = 40$. The prediction accuracy arrives above 70%. It demonstrates that the model can choose better search engine from Bing and Google for a majority of queries. Therefore, a better performance will be achieved after this suitable search engine selection. The MAP of Bing ($\text{Text}_{\text{Bing}}$), Google ($\text{Text}_{\text{Google}}$), and the one generated by our model selection ($\text{Select}_{\text{Ours}}$) are given in Table VI. Column $\text{Select}_{\text{Ideal}}$ in Table VI is the maximal MAP by selecting optimal search engine for each query according to their ground-truth performance ideally. Table VI shows that $\text{Select}_{\text{Ours}}$ achieves consistent performance improvements over both $\text{Text}_{\text{Bing}}$ and $\text{Text}_{\text{Google}}$ as well as random selection $\text{Select}_{\text{Random}}$ for all Ts. From Tables V and VI, we draw the conclusion that the proposed model chooses better search engine from Bing and Google for most queries, and therefore it can be successfully applied to optimal search engine selection.

Search Results Merging: In search engine selection, for each query, we choose a better one from l_{Bing} and l_{Google}. Instead of this binary selection, we can merge the two results to get a better one. For query q, when we have no idea of the performance of the two search results, the two results may contribute equally for the final merging results. If we have the knowledge of which one is better than the other, then a higher merging weight can be assigned to it while a lower weight is assigned to the other one. Our model can serve this role by using the predicted performance difference Δ' to set appropriate merging weight. Specifically, the Δ' is first normalized into $[-1, 1]$, and then for query q, the weighting coefficients for l_{Bing} and l_{Google} are defined as $w_{\text{Bing}} = (1/2)(1 + \delta'_q)$ and $w_{\text{Google}} = (1/2)(1 - \delta'_q)$, respectively. To form the merging result, we assign a score to each image. The score for a image is determined by three factors: its original rank position in l, its density p, and the aforementioned search engine specific weighting coefficient (l_{Bing} or l_{Google}). Specifically, the score for the ith-ranked images in l_{Bing} is $i \times (1 - p_{\mathbf{x}_i}) \times (1 - w_{\text{Bing}})$. The score for the ith-ranked image in l_{Google} is $i \times (1 - p_{\mathbf{x}_i}) \times (1 - w_{\text{Google}})$. The final

TABLE VI
MAP (× 100) COMPARISON IN SEARCH ENGINE SELECTION FROM {BING, GOOGLE}

	$\text{Text}_{\text{Bing}}$	$\text{Text}_{\text{Google}}$	$\text{Select}_{\text{Random}}$	$\text{Select}_{\text{Ours}}$	$\text{Select}_{\text{Ideal}}$
T=10	60.32	75.31	67.82	**75.53**	*80.70*
T=20	57.91	64.26	61.09	**67.11**	*71.51*
T=40	52.24	52.36	52.30	**56.30**	*60.71*
T=60	49.18	44.35	46.77	**50.13**	*54.63*

TABLE VII
MAP (× 100) COMPARISON BETWEEN WEIGHTED MERGING ACCORDING TO OUR QUERY DIFFICULTY PREDICTION AND EQUAL WEIGHT MERGING

	$\text{Text}_{\text{Bing}}$	$\text{Text}_{\text{Google}}$	$\text{Merge}_{\text{Equal}}$	$\text{Merge}_{\text{Weighted}}$
T=10	60.32	75.31	74.51	**75.49**
T=20	57.91	64.26	65.72	**68.31**
T=40	52.24	52.36	60.31	**60.51**
T=60	49.18	44.35	55.79	**56.23**

merging ranking list is derived by sorting all images in l_{Bing} and l_{Google} in ascending order of their scores. The performance of weighted merging results are given in Table VII, compared with the equal weight merging in which $w_{\text{Bing}} = w_{\text{Google}} = 0.5$. The search engine selection is actually a hard merging of the two search results with weights either 1 or 0. Table VII clearly demonstrates that merging by leveraging our query difficulty prediction outperforms the equal weight merging consistently.

Besides the applications described above, our method also has many other applications in a variety of image retrieval areas. For example, our method can be used to help collect massive clean training data. Automatic training data collection is an important issue in many applications, e.g., annotation [50], concept detection [51], and captioning [52]. By automatically predicting the quality of the data collected from Web, our method can keep the collected data clean by discarding noisy data. Another useful application of our method is that it can provide prior information for image search reranking methods to tune their parameters according to the predicted query difficulty and can automatically determine optimal reranking algorithms and features for each query [53], [54]. Furthermore, our method can be used to perform selective automatic query expansion and suggestion for users.

VI. CONCLUSION AND FUTURE WORK

In this paper, we propose a method to automatically predict the query difficulty for Web image search. We design four valuable features and then formulate the query difficulty prediction as a regression problem. The experimental results on two real Web image search datasets have demonstrated the effectiveness of our query difficulty prediction approach and also its promising applications in automatic search engine selection and search result merging.

In this paper, we focus solely on leveraging the visual features for query difficulty prediction. We do not investigate the combination of both textual and visual features. The joint utilization of features from both cues is assumed to derive better performance. We leave it as our future work and also will explore more sophisticated query difficulty prediction related features and to build more effective methods in the future.

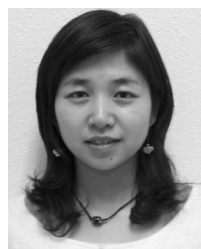
REFERENCES

[1] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.
[2] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2006, pp. 2169–2178.
[3] J. Tang, X.-S. Hua, M. Wang, Z. Gu, G.-J. Qi, and X. Wu, "Correlative linear neighborhood propagation for video annotation," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 39, no. 2, pp. 409–416, 2009.
[4] M. Wang, X.-S. Hua, R. Hong, J. Tang, G.-J. Qi, and Y. Song, "Unified video annotation via multigraph learning," *IEEE Trans. Circuits, Syst. Video Technol.*, vol. 19, no. 5, pp. 733–746, 2009.
[5] J. Tang, H. Li, G.-J. Qi, and T.-S. Chua, "Image annotation by graph-based inference with integrated multiple/single instance representations," *IEEE Trans. Multimedia*, vol. 12, no. 2, pp. 131–141, 2010.
[6] Y. Jing and S. Baluja, "Visualrank: Applying pagerank to large-scale image search," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 11, pp. 1877–1890, 2008.
[7] J. Krapac, M. Allan, J. Verbeek, and F. Juried, "Improving web image search results using query-relative classifiers," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 1094–1101.
[8] X. Tian, D. Tao, X.-S. Hua, and X. Wu, "Active reranking for web image search," *IEEE Trans. Image Process.*, vol. 19, no. 3, pp. 805–820, 2010.
[9] M. Wang, K. Yang, X.-S. Hua, and H.-J. Zhang, "Towards a relevant and diverse search of social images," *IEEE Trans. Multimedia*, vol. 2, no. 8, pp. 829–842, 2010.
[10] B. Geng, L. Yang, C. Xu, and X.-S. Hua, "Content-aware ranking for visual search," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 3400–3407.
[11] L. Yang and A. Hanjalic, "Learning from search engine and human supervision for web image search," in *Proc. ACM Int. Conf. Multimedia*, 2011, pp. 1365–1368.
[12] E. Yom-Tov, S. Fine, D. Carmel, and A. Darlow, "Learning to estimate query difficulty: Including applications to missing content detection and distributed information retrieval," *Proc. ACM SIGIR Special Interest Group on Inf. Retrieval*, pp. 512–519, 2005.
[13] J. A. Aslam and V. Pavlu, "Query hardness estimation using Jensen-Shannon divergence among multiple scoring functions," in *Proc. Eur. Conf. Inf. Retrieval*, 2007, pp. 198–209.
[14] S. Cronen-Townsend, Y. Zhou, and W. B. Croft, "Predicting query performance," in *Proc. ACM SIGIR Special Interest Group on Inf. Retrieval*, 2002, pp. 299–306.
[15] E. C. Jensen, S. M. Beitzel, D. Grossman, O. Frieder, and A. Chowdhury, "Predicting query difficulty on the web by learning visual clues," in *Proc. ACM SIGIR Special Interest Group on Inf. Retrieval*, 2005, pp. 615–616.
[16] Y. Zhou and W. B. Croft, "Query performance prediction in web search environments," in *Proc. ACM SIGIR Special Interest Group on Inf. Retrieval*, 2007, pp. 543–550.
[17] C. Hauff, V. Murdock, and R. Baeza-Yates, "Improved query difficulty prediction for the web," in *Proc. ACM Int. Conf. Inf. and Knowl. Manag.*, 2008, pp. 439–448.
[18] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 12, pp. 1349–1380, 2000.
[19] X. Geng, Z.-H. Zhou, and K. Smith-Miles, "Automatic age estimation based on facial aging patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 12, pp. 2234–2240, 2007.
[20] G. Guo, Y. Fu, C. R. Dyer, and T. S. Huang, "Image-based human age estimation by manifold learning and locally adjusted robust regression," *IEEE Trans. Image Process.*, vol. 17, no. 7, pp. 1178–1188, 2008.
[21] S. Rudinac, M. Larson, and A. Hanjalic, "Exploiting result consistency to select query expansions for spoken content retrieval," in *Proc. Eur. Conf. Inf. Retrieval*, 2010, pp. 645–648.
[22] B. He and I. Ounis, "Inferring query performance using pre-retrieval predictors," in *Proc. Symp. String Process. Inf. Retrieval*, 2004, pp. 43–54.
[23] K.-L. Kwok, L. Grunfeld, H. L. Sun, and P. Deng, "Trec 2004 robust track experiments using PIRCs," in *Proc. TREC*, 2004.

[24] H. Imran and A. Sharan, "Co-occurrence based predictors for estimating query difficulty," in *Proc. 2010 IEEE Int. Conf. Data Mining Workshops*, 2010, pp. 867–874.
[25] G. Amati, C. Carpineto, and G. Romano, "Query difficulty, robustness, and selective application of query expansion," in *Proc. Eur. Conf. Inf. Retrieval*, 2004, pp. 127–137.
[26] Y. Zhou and W. B. Croft, "Ranking robustness: A novel framework to predict query performance," in *Proc. ACM Int. Conf. Inf. Knowl. Manag.*, 2006, pp. 567–574.
[27] D. Carmel, E. Yom-Tov, A. Darlow, and D. Pelleg, "What makes a query difficult?," in *Proc. ACM SIGIR Special Interest Group on Inf. Retrieval*, 2006, pp. 390–397.
[28] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: Wiley-Interscience, 1991.
[29] J. Lin, "Divergence measures based on the Shannon entropy," *IEEE Trans. Inf. Theory*, vol. 37, no. 1, pp. 145–151, 1991.
[30] J. He, M. Larson, and M. De Rijke, "Using coherence-based measures to predict query difficulty," in *Proc. Eur. Conf. Inf. Retrieval*, 2008, pp. 689–694.
[31] X. Xing, Y. Zhang, and M. Han, "Query difficulty prediction for contextual image retrieval," in *Proc. Eur. Conf. Inf. Retrieval*, 2010, pp. 581–585.
[32] Y. Li, Y. Luo, D. Tao, and C. Xu, "Query difficulty guided image retrieval system," in *Proc. Int. Conf. Adv. Multimedia Model.*, 2011, pp. 479–482.
[33] Trecvid Video Retrieval Evaluation [Online]. Available: hppt://www-nlpir.nist.gov/projects/trecvid/
[34] K. Järvelin and J. Kekäläinen, "Cumulated gain-based evaluation of IR techniques," *ACM Trans. Inf. Syst.*, vol. 20, no. 4, pp. 422–446, 2002.
[35] V. Vapnik, *Statistical Learning Theory*. New York: Wiley, 1998.
[36] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. 5th Berkeley Symp. Math. Stat. Probability*, 1967, pp. 281–297, Univ. of California Press.
[37] J. G. Carbonell, Y. Yang, R. E. Frederking, R. D. Brown, Y. Geng, and D. Lee, "Translingual information retrieval: A comparative evaluation," in *Proc. Int. Joint Conf. Artif. Intell.*, 1997, pp. 708–714.
[38] R. Yan, A. G. Hauptmann, and R. Jin, "Multimedia search with pseudo-relevance feedback," in *ACM Int. Conf. Image Video Retrieval*, 2003, pp. 238–247.
[39] M. J. Swain and D. H. Ballard, "Color indexing," *Int. J. Comput. Vis.*, vol. 7, no. 1, pp. 11–32, 1991.
[40] W. H. Hsu, L. S. Kennedy, and S.-F. Chang, "Video search reranking through random walk over document-level context graph," in *ACM Int. Conf. Multimedia*, 2007, pp. 971–980.
[41] E. Parzen, "On estimation of a probability density function and mode," *Ann. Math. Stat.*, vol. 33, no. 3, pp. 1065–1076, 1962.
[42] J. Deng, A. C. Berg, K. Li, and F.-F. Li, "What does classifying more than 10 000 image categories tell us?," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 71–84.
[43] C.-C. Chang and C.-J. Lin, Libsvm: A Library for Support Vector Machines [Online]. Available: http://www.csie.ntu.edu.tw/~cjlin/libsvm/2004
[44] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. New York: Wiley, 2001.
[45] E. Kreyszig, *Advanced Engineering Mathematics*. New York: Wiley, 1997.
[46] S. M. Kendall and J. D. Gibbons, *Rank Correlation Methods*. London, U.K.: Edward Arnold, 1990.
[47] J. D. Gibbons and S. Chakraborty, *Nonparametric Statistical Inference*. New York: Marcel Dekker, 1992.
[48] J. L. Herlocker, J. A. Konstan, L. G. Terveen, John, and T. Riedl, "Evaluating collaborative filtering recommender systems," *ACM Trans. Inf. Syst.*, vol. 22, no. 1, pp. 5–53, 2004.
[49] M. R. McLaughlin and J. L. Herlocker, "A collaborative filtering algorithm and evaluation metric that accurately model the user experience," in *Proc. ACM SIGIR Special Interest Group on Inf. Retrieval*, 2004, pp. 329–336.
[50] X. Tian, L. Yang, J. Wang, X. Wu, and X.-S. Hua, "Transductive video annotation via local learnable kernel classifier," in *Proc. IEEE Int. Conf. Multimedia and Expo.*, Hannover, Germany, Jun. 23–26, 2008, pp. 1509–1512.
[51] J. Tang, S. Yan, R. Hong, G.-J. Qi, and T.-S. Chua, "Inferring semantic concepts from community-contributed images and noisy tags," in *Proc. ACM Int. Conf. Multimedia*, 2009, pp. 223–232.
[52] R. Hong, M. Wang, M. Xu, S. Yan, and T.-S. Chua, "Dynamic captioning: Video accessibility enhancement for hearing impairment," in *Proc. ACM Int. Conf. Multimedia*, 2010, pp. 421–430.
[53] X. Tian, L. Yang, J. Wang, Y. Yang, X. Wu, and X.-S. Hua, "Bayesian video search reranking," in *Proc. ACM Int. Conf. Multimedia*, 2008, pp. 131–140.
[54] L. Yang and A. Hanjalic, "Supervised reranking for web image search," in *Proc. ACM Int. Conf. Multimedia*, 2010, pp. 183–192.

Xinmei Tian received the B.S. and Ph.D. degrees from the University of Science and Technology of China, Hefei, Anhui, China, in 2005 and 2010, respectively, both in the Department of Electronic Engineering and Information Science.

From 2007 to 2010, she was a Research Intern with the media computing group at Microsoft Research Asia, Beijing, China. From 2008 to 2010, she was a Research Assistant at Hongkong Polytechnic University, Hongkong and Nanyang Technological University, Singapore. During 2010–2011, she was a Postdoctoral Researcher at Texas State University, San Marcos, TX. She is currently an Associate Professor in the Department of Electronic Engineering and Information Science, University of Science and Technology of China. Her current research interests include computer vision and multimedia information retrieval.



Yijuan Lu (M'05) received the Ph.D. degree from the University of Texas, San Antonio, in 2008.

She is currently an Assistant Professor in the Department of Computer Science, Texas State University, San Marcos, TX. During 2006, 2007, 2008, she was a summer Intern Researcher at FXPAL lab, Web Search & Mining Group, Microsoft Research Asia (MSRA), and National Resource for Biomedical Supercomputing (NRBSC) at the Pittsburgh Supercomputing Center (PSC), Pittsburgh. She was the Intern Researcher at Media Technologies Lab, Hewlett-Packard Laboratories (HP) 2008, and Research Fellow of the Multimodal Information Access and Synthesis (MIAS) Center at the University of Illinois at Urbana-Champaign (UIUC) 2007. Her current research focuses on multimedia information retrieval, computer vision, and machine learning. She has published extensively and serves as reviewer at top conferences and journals.

Dr. Lu received the TSU Junior Faculty Research Enhancement Award in 2008 and 2010, and her research projects are supported by the USA National Science Foundation REU and CRI program. She is the 2007 Best Paper Candidate in the Retrieval Track of the Pacific-Rim Conference on Multimedia (PCM) and the recipient of the 2007 Prestigious HEB Dissertation Fellowship, and the 2007 Star of Tomorrow Internship Program of MSRA. She is a member of the Association for Computing Machinery (ACM).

Linjun Yang (M'08) received the B.S. and M.S. degrees from the East China Normal University and Fudan University, Shanghai, China, in 2001 and 2006, respectively.

Since 2006, he has been with Microsoft Research Asia, Beijing, China, where he is currently an Associate Researcher in the Media Computing Group. Since 2009, he has been working towards the Ph.D. degree from Delft University of Technology, Delft, The Netherlands. His current interests are in the broad areas of multimedia information retrieval, with focus on multimedia search ranking and large-scale Web multimedia mining. He has received the Best Paper Award from ACM Multimedia 2009 and the Best Student Paper Award from the ACM Conference on Information and Knowledge Management 2009. He is a member of the Association for Computing Machinery (ACM).