Improving Relevance Judgment of Web Search Results with Image Excerpts

Zhiwei Li
Microsoft Research Asia
Sigma Center, Beijing, China
zli@microsoft.com

Shuming Shi
Microsoft Research Asia
Sigma Center, Beijing, China
shumings@microsoft.com

Lei Zhang
Microsoft Research Asia
Sigma Center, Beijing, China
leizhang@microsoft.com

ABSTRACT
Current web search engines return result pages containing mostly text summary even though the matched web pages may contain informative pictures. A text excerpt (i.e. snippet) is generated by selecting keywords around the matched query terms for each returned page to provide context for user's relevance judgment. However, in many scenarios, we found that the pictures in web pages, if selected properly, could be added into search result pages and provide richer contextual description because a picture is worth a thousand words. Such new summary is named as image excerpts. By well designed user study, we demonstrate image excerpts can help users make much quicker relevance judgment of search results for a wide range of query types. To implement this idea, we propose a practicable approach to automatically generate image excerpts in the result pages by considering the dominance of each picture in each web page and the relevance of the picture to the query. We also outline an efficient way to incorporate image excerpts in web search engines. Web search engines can adopt our approach by slightly modifying their index and inserting a few low cost operations in their workflow. Our experiments on a large web dataset indicate the performance of the proposed approach is very promising.

Categories and Subject Descriptors
H.3.3[**Information Systems**]: Information Search and Retrieval, I.2.6 [**Computing Methodologies**]: Artificial Intelligence

General Terms
Design, Algorithms

Keywords
Image Excerpts, Dominant Image, Web Search, Usability, User Interface

1. INTRODUCTION
The web search engines have been indispensable tools to find information from the Internet. They answer the user's query by a ranked list. Each item of the list is a web page, but only text summary of the page is displayed in result pages, which contains only page title and some keywords around the query terms. The purpose of providing a text summary for each result page is to enable the user to quickly judge whether it is what he or she needs. Providing such a simple interface has been philosophy of many search engines because it is quick but informative.

However, such a user interface misses very valuable information in web pages, say images. Usually, a web page may contain some informative images, and these images are indispensable components to present the ideas of the page. For example, we cannot imagine a news site will be if all news images are removed. Why we place some images in web pages when we make them? The reason is very straightforward: we must think images are useful to present our ideas. Thus, intuitively, showing some informative images in search results may be helpful for users to quickly understand what the page is taking about, as well as make better relevance judgment. Figure 1 illustrates the idea of showing some important images in search results of web search engines. Those images displayed in search results are extracted from corresponding web pages. It is obvious that the search results with image are more vivid and informative than traditional search results, in which only text summaries are provided. We define such search results are *image excerpts*, and these informative images are *dominant images*.

Figure 1: Search results of query "the White House". The upper figure is text summary, the lower figure is image excerpts.

From the aspect of designers, we often think a web page consists of two indispensable components, say text contents and images (or other multimedia contents). The two components should be regarded as elements of an "atom". However, current search companies build web search engine to search pages and build image search engine to search images. The two components are not utilized together in search engines to exert their combinational values. Actually, some web search engines have realized this problem, and began to use images to improve their usability. Search engines, like Live.com and Google, will insert a few images got from their image search engines on the top of the search result page for some queries (e.g. the query "David Beckham"). Obviously, such interface is far from enough to embody the value of web images. Such a user interface only can improve the overall usability of web search engines, but cannot help users to make quicker relevance judgment.

WWW 2008, April 21-25, 2008, Beijing, China.
ACM 978-1-60558-085-2/08/04.

In this paper we do not deal with problems on how to generate better text snippets [20], while we only focus on extracting dominant images from web pages to generate image excerpts along with existing text snippets. However, extracting dominant images is non-trivial, there are two difficulties:

1. For most web pages, there are lots of images embedding in them, but not all of these images are dominant images (e.g. advertisement images and decoration pictures).

2. A web page may have many dominant images, but not all these images are relevant with the user's query. For example, the web page illustrated in Figure 2 has three dominant images, but the three images represent different digital cameras, respectively.

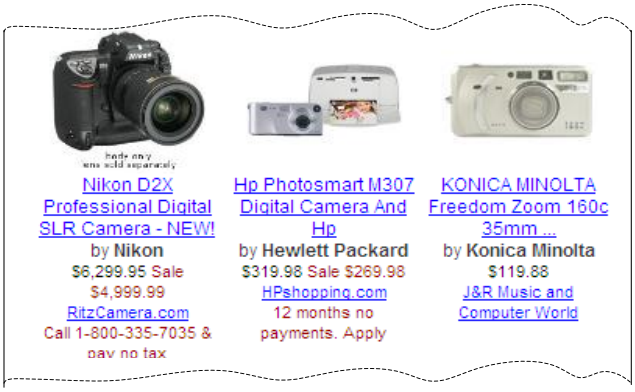

Figure 2: A web page may contain lots of images, and each image may have different meanings.

To address the two problems, we propose an approach consisting of two consecutive steps. In the first step, we train a classifier to classify images to dominant images vs. non-dominant images. But different from a common classifier, we optimize our classifier to assign a dominant score to each dominant image. This score will be used in the next step to select the best images. The first step can be performed off-line. In the second step, we combine the user's query and the dominant score got in the first step to select the most important and relevant image to generate image excerpts. This step has to be performed on-line, but the cost of this step is very low if we have indexed images according to their annotation text (i.e. file name and surrounding text).

This paper is organized as follows. In section 2, we review previous work. The framework and details of the proposed approach are given in section 3, 4 and 5. Experiments to evaluate this approach are reported in section 6. The user study is given in section 7. At last, we conclude this paper and point our future work in section 8.

2. PREVIOUS WORK

Previous studies have used different methods to summarize web documents. Some works are focused on extracting most representative sentences or phases [15, 16, 20, 28]. Ocelot [15] is a system for summarizing web pages using probabilistic models to generate the gist of a web page. Buyukkokten et al. [3] introduce five methods for summarizing parts of Web pages on handheld devices. Delort el al. [20] exploit the effect of context in web page summarization. Shen et al. [28] propose a new web summarization algorithm, which extracts the main topic of a web page through a page-layout analysis to enhance the accuracy of classification. In the web search tasks, the summarization needs consideration of search queries. Current web search engines like Google or Live most set the summaries as the texts in which search terms appear in the documents. However, presenting text summaries to users has proven to be less effective than graphical summaries in some search tasks [21, 13].

A number of studies have involved the design of graphical interfaces for presenting documents. Ayers and Stasko's thumbnails [14] consist of a reduced view of the left upper corner of a document, which is assumed to be most representative part in the document. Dziadosz and Chandraseka [21] claimed that graphical thumbnails can greatly improve the efficiency by which users to find out relevant documents from list of documents in search results. Kopetzky and Mühlhuser [24] describe a system in which links from a web page are represented by corresponding thumbnail of the document that appears temporarily when users move a mouse over the hyperlink. If the user has previously seen the page, the visual representation may aid in recognizing or classifying it [19, 23], which is usually not true in web search tasks where users are unlikely to have seen many of the documents before.

As demonstrated in previous studies [21, 13], although thumbnails are perceived as images, people usually need to read textual information presented in thumbnail previews, which causes additional time cost and reading difficulty due to poor accessibility of textual information on thumbnails. Thus, Woodruff et al [13] designed a new kind of textually-enhanced thumbnail that enforces readability of certain parts of the document within thumbnail and displays highlighted terms transparently overlaid on the reduced document. However, experiments in this study also showed that most of users were highly relying on the highlighted keywords for identifying document relevance, which again, to some extension, falls into the inefficiency suffered from text summaries.

Using a thumbnail of the "whole" page as an indication of layout of the page and all other methods in previous work leads us to ask: whether there are other more informative methods for summarizing web documents. Previous study [29] is most similar to our work, which produces web page "caricatures", containing selected features of a page often rendered in an abstract form: title, representative image, number of images, abstract, etc. In this work, the representative images in a document are selected as that can best convey the content of that document. Thus, a web document may contain multiple representative images with different contextual indication. However, in the web search tasks, the extraction of representative images needs to comprehensively consider consistence of an image with users' search queries.

We believe such indicative images are more suitable for indicating document content than thumbnails. However, as page thumbnail can give hints about the style as well as the layout of the page, one may argue that it can also present the included images to the users. However, this is untenable due to the poor accessibility of images on the thumbnails usually rendered as limited size at search results. Moreover, the desired image may not be contained on the reduced version of thumbnails [14].

Google news search [22] makes good use of images in its search results. The presence of images on the news search results is helpful to let users identify whether the news are relevant to the information need. However, it can only provide the heading or logo images on the site or newspapers of a news result, consequently resulting in an inconsistence of the displayed images with the news content. Moreover, we found that images are also available and useful in general web search tasks.

3. FRAMEWORK OF DOMINANT IMAGE EXTRACTION ALGORITHM

As a value-added component to web search engines, the proposed approach adopts the same workflow as web search engines. The workflow of a web search engine can be divided into two phases: off-line indexing and on-line searching. In the first phase, it crawl web pages and build index for them. In the second phase, it matches query terms in its index, and ranks pages as some criterion [7, 27]. Accordingly, our dominant image extraction algorithm also consists of two consecutive steps: an off-line dominant image detection step and an on-line dominant image selection step. Figure 3 illustrates the workflow of our approach.

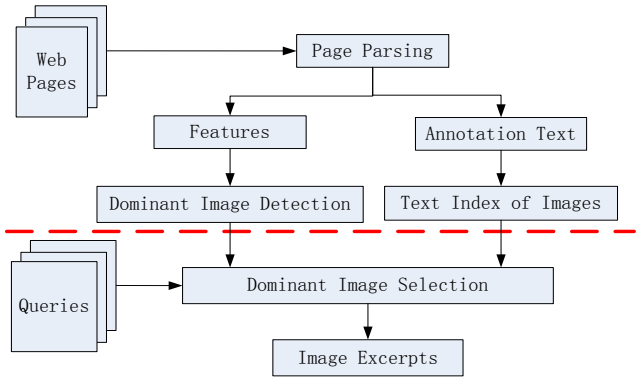

Figure 3: The workflow of the proposed dominant image extraction approach. The upper part is the off-line dominant image detection and index process, and the lower part is the on-line dominant image selection process.

Dominant Image Detection At first, we extract features both from web pages and images, and then use a classifier to determine which images are dominant images for their hosting web pages. At the same time, we compute a dominant score *d(p)* for each dominant image *p*. This real valued score reflects how important the image *p* is for its hosting web page. In feature extraction module, we also extract text annotations of dominant images from their hosting pages, and these annotations are indexed to enable quick relevance measure in the next step.

Dominant Image Selection After receiving the user's query *q*, we first retrieve the most relevant web pages (this is the work of web search engines), and then select the most relevant dominant image for each page in the search results. Since some pages have more than one dominant image, which may have different meanings, we use the annotation text automatically extracted in the first step to compute a relevance score *r(p,q)* for each dominant image *p*. Actually, this is exactly the work what web image search engines are doing. This relevance score can be used to select the most relevant dominant images.

Because in our interface design, for each item (a web page) in search results, only its "best" dominant image will be used to generate its image excerpt, we combine the two evidences together to determine which image is the best one. The final score *s(p,q)* of an image *p* is computed by

$$s(p,q) = \beta \cdot d(p) + (1-\beta) \cdot r(p,q) \qquad (1)$$

where $\beta \in [0,1]$ is a coefficient determined experimentally, and q is the query. This simple combination enables a lot of fast approximate ranking algorithms. Moreover, because our approach can be fit in workflow of web search engines, and can adopt the same distributed computing architecture, its scalability is not a problem.

4. DOMINANT IMAGE DETECTION

Usually, even in one web page, there are lots of images, but most of them are advertisement images, logo and decoration images. We stat the image numbers of pages used in our experiments, there are 13.47 images on average. Thus, we need a classifier to discriminate dominant images from non-dominant images. It is worth noticing that a page may have no dominant images, even if there are lots of images in it.

4.1 Features for Dominant Image Detection

For practical considerations, we carefully select some low-cost features to train the classifier. These features can be categorized into three groups according to their properties.

4.1.1 Image Level Features

This group of features is extracted by analyzing image content. Different from traditional usages of visual features, we do utilize some middle-level features instead of utilizing low-level features directly. Usually, dominant images tend to have better qualities than non-dominant images. Thus this group of features focuses on measuring the qualities of images.

Image Size is computed by $width \times height$, where $width$ and $height$ denote the width and height of an image, respectively. Dominant images lean to be bigger than non-dominant images.

Aspect Ratio of an image is simply computed by

$$\frac{\min(\text{width}, \text{height})}{\max(\text{width}, \text{height})}$$

Dominant images lean to be with bigger aspect ratios than non-dominant images.

Image Quality features consist of three kinds of image quality metrics: sharpness, contrast and colorfulness. We adapt the image quality measurement methods proposed in [9, 11]. Sharpness of an image is assessed by computing the ratio of the number of "clear" edges to the number of all edges. Contrast is defined as the ratio of the brightness of foreground to the brightness of background. Colorfulness of an image is defined as how many colors in this image, but we quantized it into 10 levels.

Image Categorizations consists of two kinds of image taxonomies, named as *photo* vs. *graphics* [5] and *with human faces* vs. *without human face* [12]. These features are Boolean valued. Dominant images lean to be photo and contain human faces in them.

Image Format feature reflects whether an image is an animation or not. Dominant images lean to be static images, while advertisement images, logos and banners are often animation images.

4.1.2 Page Level Features

Dominant images are defined as the most important and informative images of their hosting web pages. They are often placed in the attention attractive areas of web pages. Thus, the layout of web page is an important evidence to determine which images are dominant images. According to these observations, we extract of a group of features which can reflect the importance of an image in a web page.

Position consists of the *x* and *y* coordinates of an image in its hosting page. Dominant images lean to locate at the center or at the top of a web page.

Area Ratio is defined as the ratio of *image size* to *page size* (defined as page width multiple page height).

To get the positions of images and sizes of pages, we use a web browser to render web pages to "images" [2]. Most web browsers provide programming interfaces to retrieve attributes of pages displaying in them. However, the computational cost of this process is very high.

Number of Bigger Images is defined as how many images are bigger than this one in the same page. Dominant images lean to be the biggest images in hosting pages. In experiments, we will show the power of a rule based detector only based on this sample feature.

4.1.3 Website Level Features

Actually, the structure of web pages is very complex, and full of noisy contents. Advertisement images, logos and banners often have good qualities and may locate at important areas of web pages, but they are non-dominant images. In practice, these noise images degrade the precision of our algorithm seriously. Fortunately, we find that these noisy images existing in pages of the same website have some common characteristics, which are very useful to distinguish them from dominant images.

External Image is a Boolean value to denote an image is provided by other website (if an image and its hosting web page have different hosts in their URLs). Usually, advertisements are contents provided by advertisement agencies or commercial companies. Thus, they usually have different hosts with their hosting pages.

Duplicate Image is a Boolean valued feature. It denotes whether an image is duplicate appearing in different pages of a website. Images like logo and advertisement images are often duplicate appearing on many pages of the same website, while dominant images are less duplicated. In one site, duplicate images often have the same URL, so in this situation this feature is easy to extract. To deal with near-duplicate images with different URLs and considering the large scale dataset of web search engines, we designed a hash based algorithm [32]. A signature (a 24 bits integer in our experiments) is computed for each image. Images with the same signature will be regarded as duplicate images.

4.2 Normalization of Features

Because a dominant image is associated with its hosting web page, the absolute feature values of images of different web pages are not comparable, and the decision that an image is a dominant image for a page cannot be made until having examined all images of this web page. For example, we cannot classify a high-quality image to be a dominant image only according to its own features. We must compare it with other images in the same page. Thus, we need to normalize those real valued features of images belonging to the same web page, and the normalized features must reflect the importance of images for their hosting web pages. Consequently, they are comparable. To achieve this goal, we utilize a linear function to map its minimum value of a feature to zero, while map its maximum value to one:

$$f(x) = \begin{cases} 0 & x = min \\ \dfrac{x - min}{max - min} & min < x < max \\ 1 & x = max \end{cases}$$

where x represents a feature, and min and max represent the minimum and maximum value of this feature of images in the same web page, respectively. After normalization, features are more meaningful. For example, the new meaning of the **Image Size** feature is the percentage of images smaller than this image. Not all features need to be normalized (e.g. Boolean features).

4.3 RankBoost with Regularization

Actually, as mentioned in section 2.2, besides a dominant vs. non-dominant image classifier, we also need to compute a real valued score *d(p)* to reflect the salience of a dominant image *p*. Thus, this problem is more similar with a regression problem than a classification problem. Unfortunately, in the process of labeling training data, we find it is difficult for a user to assign a real valued score for each image, while they only can evaluate whether an images is a dominant image or this image is better than that image in a page. Thus, we ask users to label images in a page into three groups: non-dominant (0), dominant (1) and excellent (2). Only images in the last two groups are regarded as dominant images. The objective of our algorithm is to find a function, which can map images represented by features to real valued scores, and the orders of the scores are coherent with the orders of their groups. For example, if two images x_0 and x_1 are from group 1 and group 2, respectively, their dominant scores should meet $h(x_0) < h(x_1)$. Sometimes such ordinal relationships are obtained by user's feedback or from the click records logged by search engines [6]. Such an ordinal regression problem is often termed as a ranking problem [4, 6].

RankBoost [4] is a widely utilized algorithm, which can learn a strong ranking function by combining some weak ranking functions. A weak ranking function, $h(x)$, can be the ordinal of a feature of ranked objects (e.g. the order of *image size* in our problem), and it also can be a complex non-linear function of multiple features. The input of *RankBoost* is some object pairs, and each pair $< x_0, x_1 >$ denotes an ordinal relationship: x_1 should be ranked higher than x_0. For example, in our problem we permute images in the three groups to get ordinal pairs. The two images of a pair come from two different groups. The output is a ranking function $\mathrm{H(x)} = \sum_{t=1}^{T} \alpha_t h_t(x)$, where T is the number of weak ranking functions, and α_t is the weight of each function. Like AdaBoost, RankBoost utilize an iteratively gradient decent algorithm to minimize an exponential loss function

$$J(X) = \sum_{x_0, x_1} exp\big(h(x_0) - h(x_1)\big) = \sum_{x_0, x_1} exp(-\rho(x_0, x_1))$$

where x_0 and x_1 denote all image pairs, and $\rho(x_0, x_1)$ defined as $\sum_{t=1}^{T} \alpha_t (h_t(x_0) - h_t(x_1))$ is the difference of scores of images in a pair. In every iteration, it adjusts the weight of each pair, and put more efforts (i.e. weights) on difficult pairs. Once converged, it can rank images in each pair correctly. We define the margin ρ of RankBoost as

$$\rho = \min_{x_0, x_1} \rho(x_0, x_1)$$

Actually, RankBoost is an algorithm to maximize the margin. According to the theory of statistical learning [10], this margin is a hard margin, and the bigger the margin is, the better generalization performances will be obtained.

Unfortunately, like other AdaBoost type algorithms, RankBoost cannot avoid overfitting under noisy environment [8], although, it often tends not to overfit. If some pairs are incorrectly labeled, the completely "correct" classification boundary will be over-complex. Thus, to prevent overfitting, we generalize the ranking boosting algorithm by utilizing soft margins [8] rather than the original hard margins

$$\rho(x_0, x_1) \geq \rho - C \cdot \xi(x_0, x_1)$$

where C is a priori chosen constant, and $\xi(x_0, x_1)$ is a slack variable. In order to penalize the over emphasizing on noisy samples, in each iteration we set $\xi(x_0, x_1)$ as

$$\xi(x_0, x_1) = \frac{1}{|\vec{\alpha}_t|} \sum_{j=1}^{t} \alpha_j \omega_j(x_0, x_1) \quad (2)$$

where the subscript t means the t^{th} iteration, $\omega_j(x_0, x_1)$ is weight of pair $< x_0, x_1 >$ in the j^{th} iteration, and $|\vec{\alpha}_t|$ denotes the t-dimensional weights vector of previous weak ranking functions. $\xi(x_0, x_1)$ is the average weight of samples during the learning process. For noisy pairs, their weights will become bigger and bigger when iterations increase, and so as to its margin, $\xi(x_0, x_1)$. Consequently, the constraints on noisy pairs will become weaker and weaker. The partial margin of a pair $< x_0, x_1 >$ until the t^{th} iteration is computed by

$$\rho(x_0, x_1) = \sum_{j=1}^{t} \alpha_j (h_j(x_1) - h_j(x_0)) \quad (3)$$

The details of $RankBoost_{reg}$ algorithm is illustrated in Figure 4. If the C is set to 0, our $RankBoost_{reg}$ algorithm is exactly the original *RankBoost* algorithm. Once we get dominant scores of all images, we can determine a threshold by performing a line search to determine a threshold: images with scores above this threshold are classified as dominant images, otherwise non-dominant images.

Algorithm $RankBoost_{reg}$

Input: N ordinal pairs $X = << x_0, x_1 >_1, \cdots < x_0, x_1 >_N >$, and T the maximum number of iterations

Initialize: for each pair $< x_0, x_1 >$, set $w_1(x_0, x_1) = 1/N$

Do for $t = 1 : T$

a. Train weak ranking function with distribution D_t

b. Get a weak ranking function $h_t: X \to R$

c. Perform line search to compute α_t

$$\alpha_t = \underset{\alpha_t}{\operatorname{argmin}} \sum_{x_0, x_1} \exp\{-[\rho(x_0, x_1) + C \cdot \xi(x_0, x_1)]\}$$

d. Update weights by

$$w_{t+1}(x_0, x_1) = \frac{w_t(x_0, x_1)}{Z_t} exp\{-[\rho(x_0, x_1) + C \cdot \xi(x_0, x_1)]\}$$

Output: the final ranking function $H(x) = \sum_{t=1}^{T} \alpha_t h_t(x)$

Figure 4: The $RankBoost_{reg}$ algorithm.

5. DOMINANT IMAGE SELECTION

As we mentioned in section 1, some web pages have several dominant images, while these images may have different meanings (e.g. images in Figure 2 represent different digital cameras). Consequently, not all of them are coherent with user's queries. For example, for the page shown in Figure 2, if the query is "digital camera", all the three images are good enough to represent the page. But if the query is more specific, such as "Nikon digital camera", only the first image is relevant. That is, we need to extract the most relevant image from the dominant image set of a web page. From this point of view, our approach can leverage on the technologies of image search engines. Among them, the technologies on surrounding text extraction and relevance measurement are most important.

5.1 Surrounding Text Extraction

Most current web image search engines take the same way as text search engines to index images [7, 27]. In them, web images are indexed by texture annotations, which are automatically extracted from web pages. Texts surrounding an image are expected to be relevant to the semantic of the image, and often termed as surrounding text [2]. Many approaches on extracting surrounding text have been proposed. Obviously, the performances of these approaches may highly depend on how precisely they can analyze the 2D structure (i.e. layout) of web pages. Most systems utilize some simple rules instead of analyzing the page layout. For example, a method is to find a passage consisting of the 20 terms before and after the image [7]. Some researchers proposed to address this problem from the point of view of image segmentation [2]. They utilize IE (Internet Explorer) to render a web page to an image just like what the user can see. Moreover, IE provides programming interfaces to retrieve positions and other attributes of images and text blocks in a page. If the position information can be got, the geometric distances between text blocks and images can be computed precisely. However, this technique encounters many practical difficulties, such as computational cost and stability (e.g. virus and malicious scripts will crash the system). These difficulties lead this attractive method to be far from real applications. Thus, in this paper, we take a DOM (Document Object Model) based approach to extract surrounding text. This method is a good trade off of this complex method and those naive methods.

DOM is a kind of syntax tree with lots of nodes and pointers, by which you can travel the tree easily. To build a DOM tree of a web page, the web page is input as a character stream, and each encountered HTML tag, text block and object (e.g. image) are inserted in this tree as the syntax of HTML. Each node of this tree is a HTML tag, an image or a text block. For a node, its parent node is a wrapper of it, and its nearest sibling nodes are physically adjacent with it. Based on these two facts, we can analyze the relative positions of nodes. In Practical, we utilize a region growing algorithm to extract surrounding text. The original points are image nodes, and the search stops once reaches a text node. On average, this approach can deal with a page in no more than 10ms. Because this approach considers some layout information, it is a good trade-off between accuracy and computational cost.

5.2 Relevance Evaluation

Actually, not only the surrounding text, but also page title, image file name (extracted from URL of image), and other words in image URL may be relevant to image semantic. Each part of the text is an independent source of evidential information [3]. But different pieces of text contribute differently to evaluate the relevance between query and image (i.e. different parts of text should have different weights). If we treat each part of the annotation text of images as a document, we get a document collection. We use the VSM (Vector Space Model) to model documents in this collection. In VSM, a document is modeled as a set of keywords that occur in its text, and then is represented as an M-dimensional vector, where the number M is the number of distinct keywords in the document collection. Each element denoted by w_{ij} is the frequency of the j^{th} word in document d_i.

Consequently, each image is represented by several vectors, and each vector corresponds to one kind of evidence. In the same way, a query is represented as a vector in the same term space. For each contents vector v of an image, we choose *cosine* to evaluate its similarity with the query vector q

$$s(v,q) = \frac{\vec{v} * \vec{q}}{|\vec{v}| * |\vec{q}|} = \frac{\sum_{j=1}^{M} w_{vj} \cdot w_{qj}}{\sqrt{\sum_{j=1}^{M} w_{vj}^2} \sqrt{\sum_{j=1}^{M} w_{qj}^2}}$$

At last, for each image p, we use a linear function to combine all similarities to get a final relevance score

$$r(p,q) = < \vec{\alpha} \cdot \vec{s} > \quad (4)$$

where $\vec{\alpha}$ is a coefficient vector and $\vec{s}$ is the similarity vector. The value of each element of $\vec{\alpha}$ reflects the importance of this type of evidence.

To learn these coefficients, we use the click-through log of MSN image search engine. The training algorithm is also the proposed *RankBoost*$_{reg}$. We assume that users browse images in search results from top to bottom. Thus, if an image is clicked, as may indicate this image is more relevant than those images, which are ranked higher (before it) but are not clicked by the user [6]. These <*query, clicked/unclicked*> pairs constitute the input of our learning algorithm.

Table 1: The details of the experimental dataset

Wetsite	#Pages	#Images	Avg. #image Per Page
MSN.com	8111924	156775433	19.33
MIT.com	3515870	8006814	2.28
CNN.com	1976139	18407140	9.31

6. EXPERIMENTAL EVALUATIONS

In order to evaluate the performance of the proposed approach, we perform two experiments on a large-scale web dataset.

6.1 Experimental Preparations

Before reporting experimental results, we introduce the details of experimental dataset and the preprocessing of the data.

6.1.1 Preparations of Web Dataset

We crawled three typical websites. The first one is MSN.com, it composes of more than 8 million pages and 156 million images. It is noted lots of images are duplicate images (e.g. logo of MSN.com). Thus, the number of images is much bigger than the number of pages. The second one is MIT.edu. It composes of 3.5 million pages and 8 million images. The last one is CNN.com which composes of 1.9 million pages and 18 million images. More details of the three datasets are listed in Table 1.

The MSN.com is a typical commercial web site. The structures of its pages are very complex. Furthermore, there are lots of advertisement images in its pages. These images are challenges to our algorithm. On the contrary, the structures of pages of MIT.edu are very simple, and there are a few advertisements images in them, so pages from MIT.edu are the representatives of simple-structure web pages. News sites are becoming more and more important, and news images are especially critical for news articles. This is the reason why we select CNN.com as a part of our dataset.

Once the data is ready, we process data as the steps shown in Figure 2. In the first step, all web pages are parsed to extract surrounding text of images. The page level features and website level features are also extracted in this step. The text of whole pages and surrounding text of images are indexed independently [7, 27]. Thus, we actually build both a web search engine and an image search engine. At last, to accelerate the display speed of search results with dominant images, a thumbnail is generated for each image, and image level features are extracted at the same time. Based on these features, we compute the middle-level features used in our training algorithm.

6.1.2 Data Labeling

We randomly sampled 3000 pages from the dataset (pages without images are removed before sampling) to label. Since in some cases the judgment of whether an image is a dominant image for its hosting page is subjective, each page is labeled by one or two users. For each web page, the first two users independently label it. If their judgment is the same, the result will be accepted. Otherwise, the third user is asked to label this ambiguous page. The final result is got by a majority voting method. Images are labeled as three levels: *non-dominant*, *dominant* and *excellent*. Then we learn a ranking function with the proposed *RankBoost*$_{reg}$ algorithm, and consequently construct a classifier.

6.2 Experimental Results

We perform two groups of experiments to evaluate the performance of the two steps of our approach, respectively.

6.2.1 Performance of Dominant Image Detection

In order to evaluate the performance of dominant image detection, we compare our approach with three other methods. The first one is an intuitive rule based method: the biggest image is the dominant image (for all pages, exactly one image is determined to be dominant image). The second one is a SVM based ranking algorithms proposed by Joachims et al. [6], which has been used to learn the ranking functions of web search engine from click-through log. The third one is the *RankBoost* algorithm proposed by Freund et al. [4]. The four algorithms are referred to as *Biggest*, *RankSVM*, *RankBoost*, and *RankBoost*$_{reg}$, respectively. Besides the *Biggest*, all the other three algorithms utilize the permuted image pairs as input. In each image pair, the two images must come from different groups. Ordinal relationships between images in the same group are unknown.

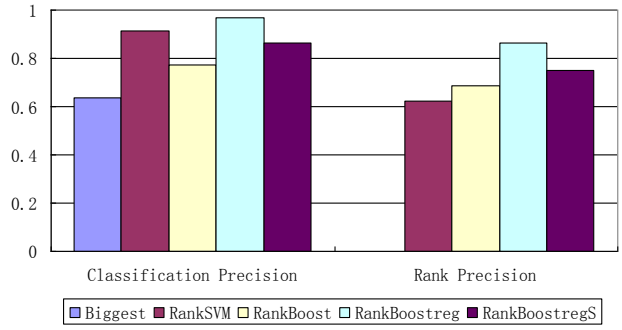

Figure 5: The accuracies of the four algorithms. *RankBoost*$_{reg}$*S* denotes our algorithm trained on a selected small feature set. The Biggest method has not ranking precision because only one image is detected as dominant image.

We randomly split the labeled data into two sets, 70% for training and 30% for testing. To prevent overfitting, we use 5-fold cross validation to find the best parameters for each algorithm. For the two boosting algorithms, we choose decision stumps (a decision tree with only two terminal nodes) as the weak rank function, and

stop iterating after 300 iterations. For *RankSVM*, we explore polynomial kernel under several different settings.

Once, we get the final ranking scores of all images, we perform a liner search for each algorithm to find its best classification point. In this process, the two classes, say *dominant* and *excellent*, are merged to one *dominant image class*. Figure 5 illustrates the error rates of the four algorithms under their own best parameter settings. The error rate of $RankBoost_{reg}$ is significantly smaller than the other three algorithms. We also examine the ranking error for the three ranking algorithms. Ranking precision is measured by the ratio of the number of correctly ranked pairs and the number of all pairs. The results are also shown in Figure 5. The classification accuracy of *RankSVM* is similar with that of $RankBoost_{reg}$ (the difference is 5.3%), but its ranking accuracy is much lower (the difference is 15.5%).

In practice, people often would like to achieve a little worse performance but with much lower computational cost. Especially, if a search engine need to process billions of pages and images (e.g. Google index about 20 billion pages), the computational cost is very critical. Thus, we explore the performance of our approach only using a few low cost features. A by-product of the $RankBoost_{reg}$ algorithm is feature selection because we use decision stumps as the weak ranking functions. In each iteration, we select the best feature to rank images. Thus, by examining these weak learners, we can predict which features are important. The performance of a classifier only leveraging on these features will not drop too much. Table 2 lists the top 7 best features and their weights. If we only use these low cost features of this best feature set to train a classifier (6 features used), its precision is just a little lower than the classifier trained with all features. The results are shown in Figure 5.

Table 2: The best features to build dominant image detector

Feature Name	Weight	Extraction Cost
Image size	0.83	Low
Photo vs. graphic	0.52	Low
Position	0.40	High
Aspect ratio	0.42	Low
Colorfulness	0.37	Low
External image	0.30	Low
Animation	0.29	Low

6.2.2 Performance of Dominant Image Selection

Given a query, the task of dominant image selection is to select the best relevant dominant images to show in the search results. In this step, we have to learn two groups of parameters. The first group is the coefficients of equation (1), and the second group is the coefficients of equation (4).

To learn the coefficients of equation (4) (i.e. a ranking function), we use the click-through log of MSN Image Search Engine. We sample almost 10,000 rows click-through log from three day's query log as training data. Like in the first experiment, we use 70% data for training, and use the left data for testing. 5-fold cross validation is adopted to select best parameters and prevent overfitting.

To learn the parameters of equation (1), we collect 40 queries from Google Trends [1], which is a bulletin board of the hottest queries submitted to Google. For each query, we only retrieve the top 20 most relevant pages from our dataset to label because usually user only has patience to examine the first two pages of search results. This is achieved by searching with the index of full-text of pages [27]. Then by the same method mentioned above, we label dominant images for each page. It is worth noticing that for each page at most one image is labeled as the final dominant image because now we have user's query in hand. If there is more than one relevant dominant image, we choose the best one (determined by $s(p,q)$). With similar settings, we learn parameters to combine the dominant score and relevance score.

The performance of dominant image selection algorithm depends on three factors, i) performance of dominant image detector, ii) performance of surrounding text extraction, and iii) performances of the relevance evaluation and scores combination algorithm. Since the performance of dominant image detection is evaluated in last experiment, and the performance of surrounding text extraction is not a focus of this paper, we only report the overall performance of dominant image selection instead of distinguishing what reasons cause the wrong selection of the final dominant images. We compare the performance of our learning algorithm with a baseline approach, in which the parameters of the two ranking functions is appointed manually. We denote this naive ranking function, whose coefficients are manually appointed, as *Manual*. We also compare the results of our algorithm with *RankSVM* [6], which is originally proposed to learn ranking functions from click-through data of search engine. For practical consideration, we use the full-feature $RankBoost_{reg}$ detector and the selected features $RankBoost_{reg}$ detector as back-end to independently run experiments.

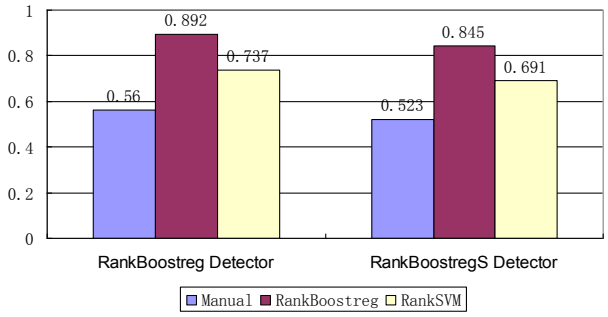

Figure 6: The accuracies of the three coefficients learning approaches using different dominant image detectors.

Figure 6 illustrates the results of this experiment. All results are reported under the best parameter settings of each algorithm. $RankBoost_{reg}$ outperforms *RankSVM* significantly. The overall performance of using selected features $RankBoost_{reg}$ is only slightly lower than the full feature one's. That is, we can achieve a low computational cost dominant image extraction solution with the overall accuracy at 0.85.

7. USER STUDY

In order to evaluate whether image excerpts can help users make quicker relevance judgment of search results, we carry out a user study. Because we do not have a real search engine which can support lots of queries, we build a meta search engine, which leverages on search results of Google.

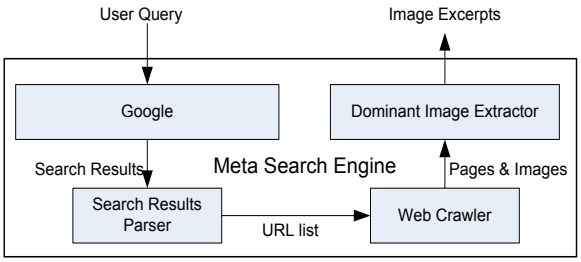

Figure 7: The workflow of our meta search engine. The input is the user's query, and the output is search results with image excerpts.

7.1 A Meta-Search Engine

Figure 7 illustrates the workflow of our meta search engine. After receiving a query, we send it to Google to request search results. Then we parse the returned HTML page to extract URL of each web page, and run a crawler to collect original web page and corresponding images. Once we get these web pages and images, we apply our approach to generate image excerpts. To accelerate the response speed of our engine, we cache the results of used queries. This method effectively avoids the dissatisfaction will caused by the long response time in user study. We give some screen shots of search results of this meta-search engine in Appendix A.

Table 3: Queries used in our user study

Categorization Methods	Query Category	Num. of Queries
Functionality	Informational	63
	Navigational	37
Semantic	Computers	11
	Entertainment	13
	Information	12
	Living	25
	Online Community	8
	Shopping	13
	Sports	18

7.2 User Study and Results

The goal of this user study is to evaluate whether image excerpts can help users to make quicker relevance judgment than traditional text snippets. To achieve this goal, we ask users, who take part in our study, to find correct answers to some queries, and observe their behaviors under search results generated by different summary methods. We have to consider two problems in this study.

How to measure which summary method is better? The most straightforward measures could include i) time to finish a search, ii) number of clicks in a search. Because the two search results have the same contents, and the only difference between them is with dominant images or not, if image excerpts are helpful, the user can find correct answers to queries in shorter time and with less clicks than do the same work given traditional text snippets. In experiments, we use average search time and average number of clicks to evaluate the performance of two summaries.

How to select queries? The first concern about image excerpts is whether it is useful to all kinds of queries. Thus, the queries used in our experiments should cover a wide range of query types. Usually, there are two kinds of query classification taxonomies: by functionality and by semantic. By functionality, queries can be classified as navigational or informational [31]; by semantic, queries can be classified as sports, shopping, entertainment, science etc [30]. We use the query set released by KDD CUP'05 [30]. In this query set, there are 800,000 queries, but only 800 queries are manually categorized to 67 categories. Thus, in our user study we randomly sampled 100 queries from the 800 queries. Because this dataset does not have labels as navigational or informational, we manually labeled the 100 queries we selected. The details of our query set are given in Table 3.

In experiments, we limit the user can only find answers in the top ten search results. For all 1000 search results of 100 queries, we found 739 image excerpts. That is to say, on average there are 7 image excerpts in search results of each query. The minimal number of image excerpts for a query is 5, and the maximum number of image excerpts for a query is 10. Thus, we can conclude the number of image excerpts is not query type sensitive.

We designed two experiments. In the first experiment, queries are classified by their functionalities, and in the second experiment, queries are classified as their semantic. We want to evaluate the performance of image excerpts for different kinds of queries. 24 university students are invited to use Google and our search engine to find answers for the queries assigned to them. All these users have experience on using web search engines, and may have necessary knowledge about these queries. These students are randomly divided into two groups. Each group has 12 students. Students in the first group are asked to take part in the first experiment, and students in the second group are asked to take part in the second experiment. Their search time and clicks on search results are recorded to measure the performance of the two kinds of summaries.

Table 4: User study results of navigational and informational queries

	Text Snippet Only		Image Excerpts	
	Clicks	Time(sec)	Clicks	Time(sec)
Informational	2.77	58.3	1.86	40.6
Navigational	2.21	46.2	1.31	37.8

7.2.1 Navigational vs. Informational Queries

We randomly partition queries belonging to one category to two equal size subsets. For example, informational queries are divided into two subsets with 32 and 31 queries, respectively. The four subsets of the two queries are named as *I1*, *I2*, *N1* and *N2*. In this experiment, 12 users are asked to use search engines to search queries classified as their functionalities. The 12 users are randomly divided into 2 groups (6 users in each group). We asked the first group to use Google to search queries in *I1* and *N1*, and use our search engine to search queries in *I2* and *N2*. While the second group are asked to search queries in *I1* and *N1* by our search engine, and search queries in *I2* and *N2* by Google. In this way, we can alleviate the bias caused by the different performances of users taking part in our study. The results of this experiment are listed in Table 4. Image excerpts significantly outperform text snippets on all the two categories.

The intents of navigational queries are defined as "to reach a particular site" [31]. The intents of informational queries are defined as "to acquire some information assumed to be present on one or more web pages" [31]. According to the experimental results, the users need more time to evaluate search results of informational queries. Fortunately, image excerpts can greatly help this task: 30.4% search time can be saved for informational queries. Thus, we can conclude image excerpts can help users to make relevance judgment for both of the two kinds of intents. Images are indispensable components to represent the ideas of web pages. Images are definitely helpful to understand what a page is talking about.

7.2.2 Queries Classified by Semantic

In this experiment, we partitioned the query set and assign them to users in a similar way as in the first experiment, but in this time we classified the queries according to their semantic. The results of this experiment are listed in Table 5. Image excerpts outperform text snippets on both search time and click numbers on all query categories.

Table 5: User study results of queries classified by their semantic

	Text Snippet Only		**Image Excerpts**	
	Clicks	Time(sec)	Clicks	Time(sec)
Computers	2.86	55.14	1.91	43.38
Entertainment	2.01	39.24	1.03	30.15
Information	3.57	68.87	2.64	45.21
Living	2.96	60.21	2.31	48.77
Online Community	3.44	63.91	2.59	42.83
Shopping	2.11	38.55	1.19	30.92
Sports	2.28	40.11	1.25	30.22

A common sense about images is that images are good at representing "concrete" concepts, such as "apple", "mountain" and "people", while they are not good at representing "abstract" concepts, such as "mutual information", "spring" and "thinking". Thus, we may predict image excerpts are not so useful for queries belonging to "abstract" categories, such as "information" and "online communities". However, we do not observe this phenomenon from our experimental results. Image excerpts work very well for almost all categories. We think the reason may lie on two facts: i) at least, adding dominant images in search results will not damage relevance judgment, ii) dominant images can make the results look vividly.

8. CONCLUSION AND FUTURE WORK

We have proposed a novel idea on using image excerpts to improve relevance judgment of web search results, and presented an effective approach to implement this idea in web search engines. According to our experimental results and the user study on a wide range of queries, we can safely conclude: i) the proposed dominant image extraction approach is effective and scalable, ii) image excerpts can be generated for almost all queries, iii) image excerpts are very helpful on accelerating user's relevance judgment to web search results. Moreover, this work is an early exploration on how to integrate existing individual vertical search engines together. Image excerpts enlighten a new way to exert the combinational power of web components (i.e. pages and images).

In future, we are going to apply this idea to other multimedia contents, such as flash, video and audio. However, for different media types the schemes to present search results should be re-designed.

APPENDIX A

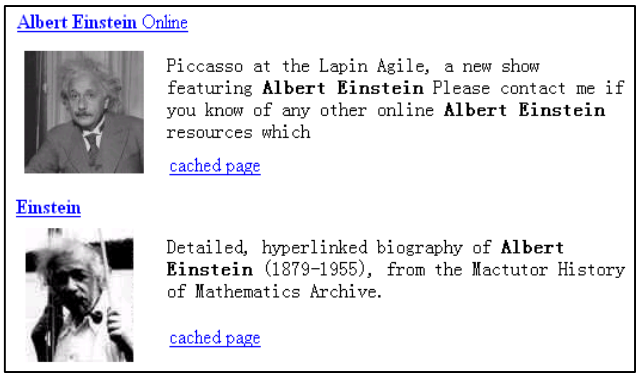

Figure 8: Search results of query "Albert Einstein"

Figure 9: Search results of query "WWW 2008"

Figure 10: Search results of query "Canon Digital Camera"

9. REFERENCES

[1] Google Trends. http://www.google.com/trends.

[2] D. Cai, X. He, Z. Li, and et al. Hierarchical clustering of www image search results using visual, textual and link

information. In Proc. of ACM international conference on Multimedia, 2004.

[3] T. A. S. Coelho, P. Calado, and et al. Image retrieval using multiple evidence ranking. IEEE Transaction on Knowledge and Data Engineering, 2004.

[4] Y. Freund, R. Iyer, R. E. Schapire, and Y. Singer. An efficient boosting algorithm for combining preferences. Machine Learning Research, 2003.

[5] A. Hartmann and R. Lienhart. Automatic classification of images on the web. In Proc. of Storage and Retrieval for Media Databases, 2001.

[6] T. Joachims. Optimizing search engines using click-through data. In Proceedings of the ACM Conference on Knowledge Discovery and Data Mining, 2002.

[7] G. Lu and B. Willam. An integrated www image retrieval system. In Proc. Fifth Australian World Wide Web Conference, 2004.

[8] G. R¨atsch, T. Onoda, and K.-R. M¨uler. Soft margins for adaboost. Machine Learning, 42(3):287–320, 2001.

[9] H. H. Tong, M. J. Li, H. J. Zhang, and et al. Learning no-reference quality metric by examples. In Proc. Of International Conference on Multi-Media Modeling, 2005.

[10] V. Vapnik. The nature of statistical learning theory. Statistics for Engineering and Information Science. Springer Verlag, Berlin, 2000.

[11] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli. Image quality assessment: from error visibility to structural similarity. IEEE Transactions on Image Processing, 2004.

[12] R. Xiao, L. Zhu, and H. Zhang. Boosting chain learning for object detection. In Proc. of International Conference on Computer Vision, 2003.

[13] A. Woodruff, A. Faulring, R. Rosenholtz, J. Morrison, P. Pirolli, Using Thumbnails to Search the Web. SIGCHI'01, March 31-April 4, 2001, Seattle, USA.

[14] E. Ayers, and J. Stasko, Using Graphic History in Browsing the World Wide Web. In Proc. 4th Intl. WWW Conf., December 1995.

[15] A. L. Berger, and V.O. Mittal. OCELOT: A System for Summarizing Web Pages. In Proc. of the 23rd annual international ACM SIGIR, Athens, Greece, 2000.

[16] O. Buyukkokten, H. Garcia-Molina, and A. Paepcke. Seeing the whole in parts: text summarization for Web browsing on handheld devices. In Proc. of WWW10, Hong Kong, China, May 2001.

[17] A. Chapman (ed.). Making Sense: Teaching critical reading across the curriculum. The College Board: NY, 1993.

[18] V. Coltheart (ed.). Fleeting Memories: Cognition of Brief Visual Stimuli. MIT Press: Cambridge, MA, 1999 (pp. 32-70).

[19] M. Czerwinski, M. V. Dantzich, G. Robertson, and H. Hoffman. The contribution of thumbnail image, mouseover text and spatial location memory to web page retrieval in 3D. In Proc. INTERACT '99, 1999, 163-170.

[20] J. Y. Delort, B. Bouchon-Meunier, and M. Rifqi. Web document summarization by context. In Proc. Of WWW12, 2003.

[21] S. Dziadosz, and R. Chandraseka, Do Thumbnail Previews Help Users Make Better Relevance Decisions about Web Search Results? In Proc. Of SIGIR'02, August 11-15, 2002, Tampere, Finland.

[22] Google news search. http://news.google.com

[23] S. Kaasten, and S. Greenberg. Integrating Back, History and Bookmarks in Web Browsers. In Proc. of CHI'01, ACM Press, 379-380.

[24] T. Kopetzky, and M. Mühlhäuser. Visual Preview for Link Traversal on the WWW. In Proc. 8th Intl. WWW Conf., May 1999, 447–454.

[25] J. B. Morrison, P. Pirolli, and S. K. Card. A Taxonomic Analysis of What World Wide Web Activities Significantly, Impact People's Decisions and Actions. Xerox PARC Report UIR-R-2000-17.

[26] A. Paivio. Pictures and Words in Visual Search. Memory & Cognition 2, 3, 515-521, 1974.

[27] S. Brin, and L. Page, The anatomy of a large-scale hypertextual web search engine. Journal of Computer Networks and ISDN Systems, 1998.

[28] D. Shen, Z. Chen, Q. Yang, H. J. Zeng, B. Zhang, Y. Lu, and W. Y. Ma. Web-page Classification through Summarization. In Proc. of the 27th ACM International Conference of Information Retrieval (SIGIR-2004). Sheffield, UK. July 2004.

[29] M. Wynblatt, and D. Benson. Web Page Caricatures Multimedia Summaries for WWW Documents. In Proc. IEEE Intl. Conf. on Multimedia Computing and Systems, June 1998, 194-199.

[30] D. Shen, J. T. Sun, Q. Yang, and Z. Chen, Building Bridges for Web Query Classification, In Proc. Of SIGIR, 2006

[31] A. Broder, A taxonomy of web search, SIGIR Forum, 2002

[32] B. Wang, Z. Li, M. Li, W. Y. Ma. Large-scale Duplicate Detection for Web Image Search. In Proc. of ICME, 2006