Improving Web Search Using Image Snippets

XIAO-BING XUE and ZHI-HUA ZHOU
Nanjing University
ZHONGFEI (MARK) ZHANG
SUNY at Binghamton

The Web has become the largest information repository over the world, thus effectively and efficiently searching the Web becomes a key challenge. Interactive Web search divides the search process into several rounds and for each round the search engine interacts with the user for more knowledge of the user's information requirement. Previous research mainly uses the text information on web pages, while little attention is paid to other modalities. This paper shows that the Web search performance can be significantly improved if imagery is considered in interactive Web search. Compared with text, imagery has its own advantage: the time for 'reading' an image is as little as that for reading one or two words, while the information brought by an image is as much as that conveyed by a whole passage of text. In order to exploit the advantages of imagery, a novel interactive Web search framework is proposed, where the *image snippets* are first extracted from Web pages and then are provided along with the text snippets to the user for result presentation and relevance feedback and also presented alone to the user for image suggestion. User studies show that it is more convenient for the user to identify the Web pages he or she expects and to reformulate the initial query. Further experiments demonstrate the promise of introducing the multimodal techniques into the proposed interactive Web search framework.

Categories and Subject Descriptors: H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Information filtering*; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Query formulation*; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Relevance feedback*

General Terms: Algorithms, Design, Experimentation, Human Factors

Additional Key Words and Phrases: Image snippet, relevance feedback, image suggestion, term suggestion, multimodality, interactive Web search

Z.-H. Zhou and X.-B. Xue were supported by the National Science Foundation of China (60505013, 60635030, 60721002), the Jiangsu Science Foundation (BK2005412) and the Foundation for the Author of National Excellent Doctoral Dissertation of China (200343); Z. Zhang was supported by NSF (IIS-0535162), AFRL Information Institute (FA8750-05-2-0284), and AFOSR (FA9550-06-1-0327).

Author's address: X.-B. Xue and Z.-H. Zhou, National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, China; email: {xuexb, zhouzh}@lamda.nju.edu.cn.
Z. Zhang, Department of Computer Science, SUNY at Binghamton, Binghamton, NY 13902, USA; email: zhongfei@cs.binghamton.edu.

1. INTRODUCTION

With the explosive development of the Internet, the Web has become the largest information repository over the world. Due to its huge volume, users easily feel lost in this repository. Web search engines attempt to help users find a needle in the haystack. It is not surprising that Web search techniques have attracted more and more attention in the related communities.

The fact that most Web users give only short queries makes Web search challenging. Recent analyses of search engine logs revealed that the average length of Web queries is about 2.3 words [Silverstein et al. 1998]. It is difficult for search engines to perceive the user's information requirement from such short queries. A possible solution is to divide the search process into several rounds and, for each round, the search engine interacts with the user to obtain more information about the user's requirement. Such paradigm is called interactive Web search.

Relevance feedback and *term suggestion* are two widely used techniques for interactive Web search. Relevance feedback requires the user to select relevant Web pages returned in the previous round of search as the feedback to allow the search engine automatically reformulate the query for the next round of search. For term suggestion, the search engine automatically generates some terms as suggestions and then the user chooses some from those suggested terms to allow the search engine to refine the query. These techniques successfully facilitate the interaction between the search engine and the user and improve the performance of Web search.

However, in the previous research for interactive Web search, imagery is almost not considered. As a different modality from text, imagery has its own advantages. On one hand, imagery is easy and fast to perceive by humans. It is reported that humans can get the gist of an image in 110ms or less [Coltheart 1999], while in this period of time, humans can only read less than 1 word, or skim 2 words (The average English reader can read about 4.2 words per second, and can skim or scan at roughly 17 words per second [Chapman 1993]). On the other hand, imagery can provide abundant information just as some people say 'an image is worth a thousand words'. In general, the time for 'reading' an image is as little as that for reading one or two words, while the information brought by an image is as much as that conveyed by a whole passage of text. Considering the above advantages, this work is thus motivated to bring imagery into interactive Web search to develop an effective and efficient search framework.

Text snippet is widely used in Web search. It is a summarization of the retrieved Web page, that helps a user identify the content of the page. Considering the complementarity between text and imagery, this paper proposes the similar *image snippet* concept for a Web page. Image snippet is one of the images in a Web page that is both representative of the theme of the page and at the same time closely related to the search query. A Web page may or may not necessarily have an image snippet as in an extreme case a Web page may not even have images at all. However, it is obvious that image snippets do exist in many Web pages. Based on this consideration, a novel interactive Web search framework, WebSIS (Web Search using Image Snippets), is proposed in this paper.

In order to introduce WebSIS, the process of the traditional interactive Web search is briefly described as follows: First, a search query is posed by a user;

second, the search result consisting of text snippets of the retrieved Web pages is returned to the user for browsing; third, if the search result is satisfactory to the user, the search process is completed; otherwise, which is the typical case, the user may reformulate the query through techniques such as relevance feedback and term suggestion. This process may repeat until the final result becomes satisfactory to the user. In WebSIS, images are used in the second and the third step, which will be respectively described below.

For the second step, that is, result presentation, the image snippets are provided along with the text snippets to the user. On one hand, since imagery attracts the user's attention more easily than text, it is natural for the user to first focus on the image snippets. Furthermore, since humans 'read' imagery much faster than text, the user could quickly restrict the potentially relevant Web pages to those with the relevant image snippets. After further checking the corresponding text snippets, the user can find the desired Web pages easily. Thus, it is expected that the user can locate his/her desired Web pages much faster with the help of the image snippets. On the other hand, the user identifies the relevance of a Web page based on both the text snippet and the image snippet. Since imagery conveys abundant information, it is anticipated that the image snippet could provide somewhat complementary information, which helps the user make the decision more accurately. Note that, since imagery can be perceived quickly by humans, the additional information brought by the image snippets imposes little cognitive load on the user. In contrast, if this information is brought by displaying more detailed text snippets, the cognitive load of the user will be significantly increased. Since careful attention to the cognitive load of the user is essential for interactive Web search, using image snippets is much more convenient than using more detailed text snippets.

For the third step, that is, query reformulation, *relevance feedback* and *term suggestion* both benefit from the introduction of the image snippets.

For *relevance feedback*, in the traditional search environment, the user labels the retrieved Web pages only according to the text snippets of Web pages. Since providing the image snippets along with the text snippets further helps the user label the Web pages more accurately, the performance of the relevance feedback in WebSIS is also expected to be improved due to more accurate labels.

For *term suggestion*, the user is not required to label the retrieved Web pages but instead to choose the relevant terms from the ones suggested automatically by the search engine. The selected terms are then added to the query for a further search. Compared with relevance feedback, term suggestion decreases the user load at the cost of the decreased amount of information available to the search engine, since the relevance of the text snippets is replaced by the relevance of terms. Considering the characteristics of imagery, i.e., quick perception and abundant information, when compared with term suggestion, using the image snippets as suggestions may provide more information to the search engine and at the same time the user load is almost the same[1]. Thus, the image suggestion scheme provided in WebSIS is also expected to be more effective than term suggestion.

Presumably, multimodal techniques are promising to improve the performance

[1]As indicated above, the user spends nearly the same amount of time to read an image and to read one or two words.

of interactive Web search. Unfortunately, it is somewhat difficult to directly incorporate these techniques into the traditional interactive Web search framework. For WebSIS, however, these techniques can be utilized easily and naturally. It is known that, besides text, there are typically other modalities on a Web page, which also convey the important information about the content of the Web page. Thus it is helpful for a search engine to rank Web pages according to not only text content but also the information provided by other modalities. Nevertheless, as the query posed by the user usually consists of words, in most cases it is difficult to obtain the user's information requirement expressed by other modalities. Take imagery as an example. When the user poses a text query, it is difficult for the search engine to automatically find an image corresponding to the text query and it is also impractical to ask the user to provide an image along with the text query. In WebSIS, the image snippets labelled as relevant by the user can be naturally used as the required image query with which a search engine can conduct a search based on both text content and visual features.

In general, there are several advantages of using the image snippets in interactive Web search. First, the image snippet can help the user locate his/her desired Web pages more quickly and more accurately. Second, since the image snippet can help the user label the retrieved Web pages more accurately, it is also expected to improve the performance of relevance feedback. Third, the proposed image suggestion provides more information about the user's information need while the user load is almost the same when compared with term suggestion. Fourth, introducing the image snippets helps to incorporate the multimodal techniques into interactive Web search.

The rest of this paper is organized as follows. Section 2 briefly reviews related work. Section 3 presents the image snippet based interactive Web search framework. Section 4 describes the experiments conducted. Section 5 reports on a user study. Finally, section 6 concludes the paper.

2. RELATED WORK

2.1 Interactive Web Search

Since this work focuses on introducing imagery into interactive Web search, in this section, only the techniques of interactive Web search are reviewed. The review of the techniques used in general Web search, such as Web page crawling, Web page indexing, Web page cleaning and Web page ranking, can be found in [Chakrabarti 2003]. The techniques used in interactive Web search exploit the information provided by the user to refine the initial query. *Relevance feedback* and *term suggestion* are two widely studied techniques in interactive Web search.

Relevance feedback is a classical technique in the field of information retrieval. The user is required to label the top retrieved documents and then these labels are used to refine the initial query by the system. Given the relevance information, two methods are available to refine the query, that is, query reweighting which updates the weights of query terms, and query expansion which adds new query terms. These techniques have been widely discussed according to the information retrieval model used, i.e., the vector space model and the probabilistic model. The readers who are interested in these topics can refer to [Baeza-Yates and Ribeiro-Neto 1999].

Recently, researchers discussed to what extent the user should be involved in the process of query expansion. After the feedback from the user is collected by the system, a group of candidate terms can be generated. Should the system directly add the top ranked terms to the query or just display these candidates and let the user make the final decision? The former choice is called Automatic Query Expansion (AQE) and the latter one is called Interactive Query Expansion (IQE). Koenemann and Belkin [1996] conducted a user study of 64 novice searchers. Four versions of INQUERY systems were provided and each one was offered a different level of user involvement in the process of query refinement. This study showed that the searchers using the IQE-version system did the best and the searchers using other versions of systems would like to have more control of the terms added to the query. Recently, Ruthven [2003] reexamined the effect of AQE through a well designed large scale experiment. The experiments simulated a user's all possible choices of the candidate terms and showed that on average the user made a worse choice than the system did.

Although the technique of relevance feedback succeeded in improving the performance of retrieval, the fact that most users were reluctant to carefully label the retrieved documents limited its practical application to Web search. Term suggestion waives the requirement of labelling the retrieved documents and just needs the user to select the candidate terms suggested by the search engine. The core of this technique is how to generate the candidate terms automatically without the interaction of the user, which is often called *automatic query refinement*[2].

Early research can be divided into two categories: global analysis and local analysis. For global analysis, the candidate terms and the information used to rank them were both from the whole corpus while for local analysis the top ranked documents replaced the whole corpus. The techniques of global analysis included term clustering, Similarity Thesauri and Phrasefinder, while the examples of local analysis included local clustering and local context analysis. Almost all these methods were based on the analysis of the co-occurrence relationships between the query and the candidate terms. Besides this basic consideration, some other important factors were considered, such as the selection of a candidate term should be based on its relationships with all terms in the query, not just the most similar term; the co-occurrence relationship should be calculated within a smaller range such as passages rather than the whole document; and the candidate terms should not be treated equally, where the noun words may be preferred to other words [Jing and Croft 1994; Qiu and Frei 1993; Xu and Croft 2000]. Pseudo-feedback can also be regarded as a local technique of automatic query refinement, where the top retrieved documents were simply considered as relevant and then the standard relevance feedback techniques were used.

The following research solved the problem of automatic query refinement from different views. Phrasier [Jones and Staveley 1999] and Paraphrase [Anick and Tipirneni 1999] used linguistic techniques. Specifically, Phrasier automatically extracted

[2]The automatic query refinement is easily confused with AQE, where the former discusses how to refine the query without the relevance information of the retrieved documents while the latter concerns directly adding to the initial query the terms generated using the relevance information of the retrieved documents.

'keyphrases' from the content of the stored documents and used these keyphrases to rank documents and to facilitate browsing and query refinement. Paraphrase assumed that the key concept was more likely than other terms to appear in a series of semantically related lexical compounds. Based on this assumption, Paraphrase extracted key concepts from the retrieved documents through calculating lexical dispersion and used these key concepts to organize the lexical compounds. These key concepts and their corresponding lexical compounds were provided to the user as the suggested terms. Instead of analyzing the content of a web page, Kraft and Zien [2004] proposed to extract the suggested terms from the anchor text of a web page. The experiments showed that the anchor-text based query refinement could generate the suggested terms with higher quality than those generated by content-based refinement.

Recently, implicit feedback [Kelly and Teevan 2003] was proposed and attracted much attention, which collected some useful information from the user's behaviors during the interaction with search engines to refine the query without the explicit user involvement. Joachims [2002] exploited the clickthrough data collected by the search engine to reflect some partial relations between documents and used these relations to train a retrieval function using SVM. Huang et al. [2003] utilized the search engine logs to calculate the co-occurrence relationships between queries based on all stored query sessions and suggested some queries closely related to the initial query. Shen et al. [2005] used both the query history and the corresponding click history in a query session to develop a language model for the context-sensitive information retrieval. Some important user studies were also conducted to show what factors would affect the performance of implicit feedback. Kelly and Belkin [2004] showed that the displaying time differed significantly with different users and also varied noticeably with different tasks for any single user. They also indicated that there was no significant relationship between the displaying time and the relevance of documents. Thus, it should be careful to use the displaying time as a kind of implicit feedback. White et al. [2005] designed a careful user study to show when and at what circumstances the implicit feedback would perform well and their experiments showed that there were three factors affecting implicit feedback: the search task complexity, the search experience of the user, and the stage in the search process.

Note that although the terms generated by the techniques of automatic query refinement mentioned above can be directly added to the initial query without any involvement of the user, in practice, the suggested terms are usually displayed to the user to let the user make the final decision. Some user studies based on the practical Web search engines have confirmed the effect of term suggestion [Anick 2003; Dennis et al. 2002].

2.2 Multimodality

Researchers have noticed that different modalities could provide some complementary information to each other. Barnard and Johnson's work [Barnard and Johnson 2005] gave an good example. In their work, images were introduced to augment the text information to complete the task of word sense disambiguation. Specifically, a statistical model for the joint probability of image regions and words was learned to automatically annotate images. The probabilities of all the possible senses of a

word were compared and the most probable sense was determined. Better results were reported when this strategy was used in conjunction with the traditional word sense disambiguation.

Considering the promise of multimodal techniques and the popularity of multimodal information on the Web, multimodal information retrieval has attracted much attention recently.

Yang et al. [2002] proposed a mutlimodal retrieval framework called Octopus. In this framework, objects from different modalities were modelled as nodes in three graphs, which expressed the relations between objects through user interpretation, structure analysis and content similarity calculation, respectively. The user could pose a query from any modality, and then after seed generation, candidate spanning and result distillation, the framework returned a list of relevant object from different modalities. The major advantage of Octopus lied that a consistent framework was provided for multimodal retrieval, which facilitated cross-media retrieval.

Fan et al. [2005] proposed a Photo-To-Search system which allowed the user to input mutlimodal queries. This system supposed that such queries came from the mobile phone with camera, with which the user can easily obtain the image query he/she expected to pose. With an image and some text message, the system first used the text message to find some text relevant web pages and then the candidate images in text relevant web pages were compared with the input image according to visual features to find the returned images.

Cai et al. [2004] used multimodal information to help cluster image search results. For each image, the text description, the visual features and the link information were obtained. Correspondingly, the text graph, the visual graph and the link graph were constructed. First, the text graph and the link graph were used to cluster images into different semantic groups. Then, the visual graph was used to further divide the images from the same semantic group into different visual groups.

Our WebSIS framework is different from the above work from two aspects. First, in WebSIS, the imagery information is used as the complement to the text snippet during the interactions with users, instead of the search targets as in the above work. This strategy is based on the characteristics of imagery, i.e., quick perception and abundant information. Second, in WebSIS, it is natural and effective to generate an image query. In the panel of MIR05 [Jaimes et al. 2005], a panel member indicated that 'I think one of the challenges for multimedia information retrieval is a simple but effective way of forming a query'. In WebSIS, the relevant image snippets selected by the user during the image suggestion can be naturally used as the user's query expressed in the imagery modality, which can be further used in multimodal image suggestion. In contrast, the Photo-To-Search framework solved this problem through a mobile phone with camera and Octopus didn't address this problem directly.

The most related work to WebSIS is Woodruff et al.'s research [2001]. In their work, an enhanced thumbnail was proposed to help the user search the Web. This enhanced thumbnail was an image, where the whole Web page was resized to fit into this image and important words were highlighted in order to be readable to the user. It was reported that with this enhanced thumbnail the user could find the answer in less time than with the text snippet. The major difference lies that the

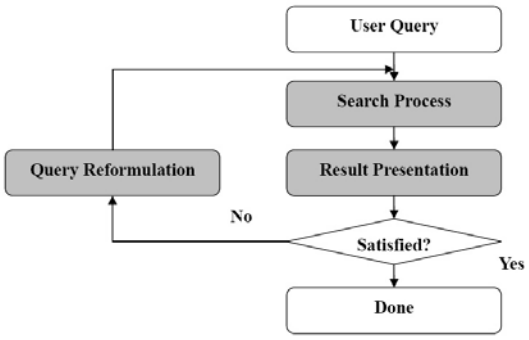


Fig. 1. The WebSIS framework

enhanced thumbnail is used to replace the text snippet in [Woodruff et al. 2001], but in WebSIS the image snippet is provided along with the text snippet to provide complementary information. Considering the abundant information expressed by the text snippet and the user's familiarity with the text snippet, it is better to augment the text snippet instead of replacing it.

3. THE WEBSIS FRAMEWORK

In this section, the proposed interactive Web search framework, WebSIS, is introduced in detail and the implementation is also discussed.

3.1 Framework

The framework of WebSIS is illustrated in Fig. 1. The basic process of WebSIS is similar to that of the traditional interactive Web search except that different operations are required in the grey boxes in Fig. 1.

In *Search Process*, the major difference is that, for a given query, the image snippets are extracted from each Web page retrieved. As defined, an image snippet should be representative of the theme of the Web page, and at the same time also relevant to the query. In order to identify the image snippet, a Web page segmenter is first used to segment the original Web page into several blocks and then these blocks are ranked according to their importance and relevance respectively. The importance of a block is evaluated by a Web block evaluator and the relevance of a block is measured by the similarity between the query and the content of this block. These two ranks can be combined to form the final rank of blocks. Given this final rank, the highest-ranked block with at least one image is extracted. Note that some constraints can be imposed on the extracted block in order to avoid noisy blocks such as advertisements. The image appearing in this block is considered as the image snippet. If multiple images appear, some heuristic rules are applied to break the tie to select one of the images. The algorithm for the image snippet extraction is summarized in Table I.

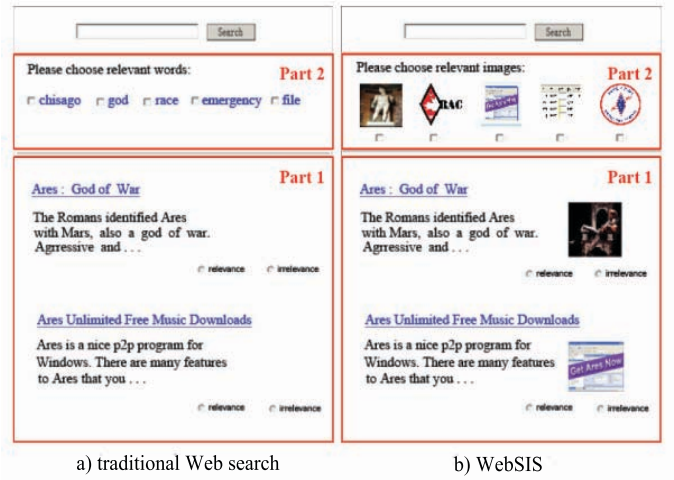
After *Search Process*, the *result page* is obtained, which is the basis for *Result Presentation* and *Query Reformulation*. Fig. 2 illustrates the differences between the result page of WebSIS and that of the traditional interactive Web search.

Part 1 of Fig. 2 is the part used for *Result Presentation*. Clearly, in WebSIS,

Table I. Algorithm for image snippet extraction

ALGORITHM: Image Snippet Extraction
INPUT: Web page *Page*, search query *Query*
OUTPUT: image snippet *Image*
PROCESS:

(1) Segment *Page* into blocks using a Web page segmenter and store the blocks in B.
(2) Rank the blocks in B according to the importance evaluated by a Web block evaluator and get $Rank_{Imp}$
(3) Rank the blocks in B according to the similarity between the *Query* and the text content of each block and get $Rank_{Sim}$
(4) Combine $Rank_{Imp}$ and $Rank_{Sim}$ to form the final Rank $Rank_{Final}$.
(5) The highest-ranked block in $Rank_{Final}$ with at least one image and at the same time satisfying the predefined constraints is extracted as *Block*.
—IF *Block* is empty, return NULL.
—ELSE select *Image* from the images appearing in *Block* according to the specified heuristic rule and return *Image*.



a) traditional Web search b) WebSIS

Fig. 2. Comparison of the result page

the image snippet is provided along with the corresponding text snippet. With the help of the image snippet, users can identify the Web pages they expect easily and accurately.

In *Query Reformulation*, relevance feedback and term suggestion are both provided. Relevance feedback requires the user to label whether the retrieved Web pages are relevant or not and then all these labels are collected as the feedback information to automatically reformulate the query. With the explicit help of image snippets, it is easier for the user to more accurately determine the relevance of the retrieved Web pages to the query. Consequently, this type of query reformulation is expected to be more effective.

Part 2 of Fig. 2 indicates the part used for term suggestion. In WebSIS, a new

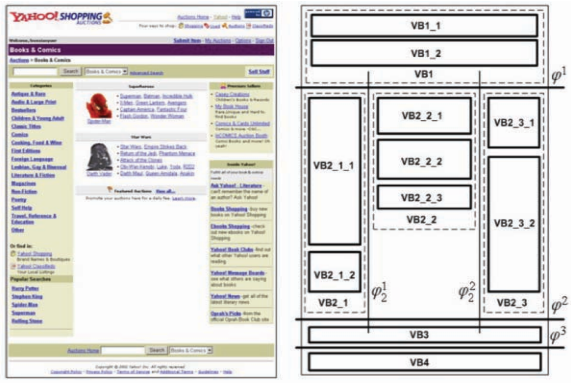


Fig. 3. A simple example for the VIPS algorithm.

technique called image suggestion is provided. Instead of terms, the image snippets extracted from Web pages are provided to the user serving as the suggestions. The user simply selects the relevant images as the feedback to allow the system automatically reformulate the query.

3.2 Implementation

While the WebSIS framework is a general approach to Web search, in this subsection, the implementation of a prototype of the WebSIS framework is discussed.

The major work of *Search Process* is to implement the algorithms of Image Snippet Extraction as described in Table I. The discussions refer to the steps defined in the algorithm in Tabel I.

For Step 1, many techniques can be used to segment the Web page. Here, the VIPS (VIsion-based Page Segmentation) algorithm [Cai et al. 2003] is used as the Web page segmenter. The basic idea of this technique is to emulate how a user understands the Web page layout based on the user's visual perception. Clearly, when a user browses a Web page, many spatial and visual cues help the user divide the Web page into several semantic regions of the page. Such cues include lines, blanks, images, colors and so on. Usually, the visually closely packed regions are likely to share the same semantics. Such regions are thus extracted as a block of the page. A simple example is given in Fig. 3[3]. The detailed description about the VIPS algorithm can be found in [Cai et al. 2003].

For Step 2, the Web block importance model proposed by [Song et al. 2004] is used as the Web block evaluator. In this model, two types of features, spatial features and content features, are extracted for each block. Spatial features include the position of the block and the size of the block. Content features include the number and size of images, links, interaction elements and forms. In the original model, importance of four different levels is assigned to each block in the training set by visual inspection. These instances are then fed to a neural network or SVM to learn the importance model. The detail of this model can be found in [Song et al. 2004]. In WebSIS prototype, however, since each block is either important

[3]Cited from http://www.ews.uiuc.edu/~dengcai2/VIPS/VIPS.html

Table II. Process of relevance feedback

ALGORITHM: Relevance Feedback
INPUT: database of Web pages *DB*, search query *Query*, *FeedbackNum*
OUTPUT: a rank of Web pages *Rank*
PROCESS:
$P \leftarrow Query$, $N \leftarrow \emptyset$;
Rank ← the initial rank of Web pages by using Eq. 1 on *DB*;
In each round of relevance feedback:
(1) Ask the user to label the top *FeedbackNum* Web pages in *Rank*. $P \leftarrow P \cup$ relevant Web pages, $N \leftarrow N \cup$ irrelevant Web pages
(2) *Rank*← the updated rank of Web pages by using Eq. 1 on *DB*

or not, a simplified, binary importance model is used and the importance of each block is measured using the confidence value output by this model.

For Step 3, the standard vector space model is used to represent the query and the text content of each block. The widely used cosine function is used as the similarity meausre.

For Step 4, the two ranks can be combined with different weights and three types of combinations are tested in the following experiments, that is: using $Rank_{Imp}$ only, using $Rank_{Sim}$ only, and combining them with equal weights.

For Step 5, the constraints to exclude noisy blocks could be that a block's probability of being important must be larger than the one of being unimportant. Typical heuristic rules used to select one from a group of tied images include identifying the image with the largest size or identifying the image with its ALT field containing the query term. In the WebSIS prototype, the image appearing first in the block is selected as the image snippet.

The two types of *Query Reformulation* are implemented in the prototype respectively and are discussed below.

For Relevance Feedback, the standard relevance feedback technique in text retrieval is used to automatically reformulate the query. For each query, the user labels the *FeedbackNum* (the number of Web pages used for feedback) Web pages as relevant or not. Since the vector space retrieval model is used in WebSIS to rank Web pages, a variation of Rocchio method is used to refine the query, which is shown in Eq. 1.

$$L(x, P, N) = \frac{\sum_{y \in P} Sim(x, y)}{|P|} - \frac{\sum_{y \in N} Sim(x, y)}{|N|} \tag{1}$$

Here, P is the set for storing relevant documents while N is the set for irrelevant ones. $|A|$ is the size of the set A. x and y represent documents. $Sim(x, y)$ is the cosine similarity between documents. $L(x, P, N)$ is the ranking function which assigns a value to each document x in the collection. Initially, P only contains the input query and N is empty. With relevance feedback conducted, the documents labelled by the user are added into P and N respectively according to the assigned labels. This process is shown in Table II.

For image suggestion, the implementation is described as follows. First, the source of the candidate images used for image suggestion is determined, that is

Table III. Process of image suggestion

ALGORITHM: Image Suggestion
INPUT: database of Web pages *DB*, search query *Query*, *SugImgNum*
OUTPUT: a rank of Web pages *Rank*
PROCESS:
$P \leftarrow Query$, $N \leftarrow \emptyset$, $U \leftarrow$ image suggestions extracted from *DB*;
Rank $\leftarrow$ the initial rank of Web pages by using Eq. 1 on *DB*;
In each round of relevance feedback:

(1) *ImgRank*$\leftarrow$ a rank of image suggestions by using Eq. 1 on U
(2) Ask the user to label the top *SugImgNum* image suggestions in *ImgRank*.
$P \leftarrow P\cup$ relevant image suggestions, $N \leftarrow N\cup$ irrelevant image suggestions,
$U \leftarrow U-$ lablled image suggestions
(3) *Rank*$\leftarrow$ the updated rank of Web pages by using Eq. 1 on *DB*

document-based or block-based. The former simply uses the image snippets extracted from each Web page as the candidates while the latter selects an image from each block, where the image that first appears is preferred, and uses these block-based image snippets as the candidates. Second, the text description is extracted for each candidate image. The content of the whole document is used to describe the document-based image snippet and the content of the block is used to describe the block-based image snippet. Third, the candidates are ranked according to the similarity between the query and their text descriptions. Finally, the top-ranked *SugImgNum* (the number of the image snippets used for suggestion) images are selected to be put into the feedback pool. After obtaining the feedback of the user, the same strategy as relevance feedback is adopted, except that the text descriptions of the labelled image snippets are added into P and N instead of the labelled documents. The process of image suggestion is shown in Table III. In the following experiments, the document-based image suggestion and the block-based one are both tested. Note that if an image snippet is labelled by the user in the current round, it will not be displayed again in the subsequent rounds.

As is discussed in the Introduction section, the use of the image snippets makes it possible to obtain the user's information need expressed by another modality, which is promising to improve the retrieval performance. In order to use the visual features of the image snippets, a variation of the standard image suggestion, called multimodal image suggestion, is implemented. Its process is shown in Table IV.

Initially, no feedback is provided for image suggestions, where P_{img} and N_{img} are both empty, thus in this situation $ImgRank$ is totally determined by $ImgRank_{txt}$. When ranking Web pages based on visual features, the image snippet of each Web page is used to represent each Web page. When using visual features, $Sim(x, y)$ is the similarity between two visual feature vectors corresponding to x, y respectively.

4. EXPERIMENTS

4.1 Experimental Configuration

In order to design realistic Web search experiments, three types of queries are designed: *ambiguous* query, *unambiguous* query and *hot* query. 10 queries are respectively designed for ambiguous query and unambiguous query. 9 queries chosen

Table IV. Process of multimodal image suggestion

ALGORITHM: Multimodal Image Suggestion
INPUT: database of Web pages *DB*, search query *Query*, *SugImgNum*
OUTPUT: a rank of Web pages *Rank*
PROCESS:
$P_{txt} \leftarrow Query$, $N_{txt} \leftarrow \emptyset$, $P_{img} \leftarrow \emptyset$, $N_{img} \leftarrow \emptyset$,
$U \leftarrow$ image suggestions extracted from *DB*;
$Rank \leftarrow$ the initial rank of Web pages by using Eq. 1 on *DB* based on P_{txt} and N_{txt};
In each round of relevance feedback:

(1) $ImgRank_{txt} \leftarrow$ rank elements in U using Eq. 1 based on P_{txt} and N_{txt}.
$ImgRank_{img} \leftarrow$ rank elements in U using Eq. 1 based on P_{img} and N_{img}.
$ImgRank \leftarrow$ the combination of $ImgRank_{txt}$ and $ImgRank_{img}$

(2) Ask the user to label the top *SugImgNum* image suggestions in *ImgRank*.
$P_{txt} \leftarrow P_{txt} \cup$ relevant image suggestions described by text
$N_{txt} \leftarrow N_{txt} \cup$ irrelevant image suggestions described by text
$P_{img} \leftarrow P_{img} \cup$ relevant image suggestions described by visual features
$N_{img} \leftarrow N_{img} \cup$ irrelevant image suggestions described by visual features
$U \leftarrow U -$ labelled image suggestions

(3) $Rank_{txt} \leftarrow$ rank elements in *DB* using Eq. 1 based on P_{txt} and N_{txt}
$Rank_{img} \leftarrow$ rank elements in *DB* using Eq. 1 based on P_{img} and N_{img}
$Rank \leftarrow$ the combination of $Rank_{txt}$ and $Rank_{img}$

Table V. Three types of queries

Type	Query
ambiguous	tiger, apple, dove, eagle, jaguar, jordan, newton, aaai, cambridge, trec
unambiguous	tiger beer, apple fruit, dove chocolate, eagle bird, jaguar car, jordan basketball, newton physics, aaai artificial intelligence, cambridge university, trec information retrieval
hot	digital camera, hurricane katrina, ares, ipod nano, janet jackson, brad pitt, myspace, orkut, xbox

from Google 2005 top searches[4] are used for the hot query category. The details of these queries are documented in Table V. Note the one-to-one correspondence between the queries in the ambiguous category and the unambiguous category. For example, the query 'jaguar' in the ambiguous category and the query 'jaguar car' in the unambiguous category correspond to the same information requirement. A passage of text is provided for each query, which describes the information requirement of this query in detail.

For each query, the first 200 Web pages returned by the Google search engine and their corresponding text snippets are downloaded. These 200 Web pages consist of a small corpus and experiments for each query are conducted on this corpus. The search strategy used here is the standard Vector Space Model and the retrieval function is shown in Eq. 1.

Nine volunteers are involved in the following experiments. Seven of them have the background of computer science and two of them have the background of math-

[4]http://www.google.com/intl/en/press/zeitgeist2005.html

Table VI. A list of tasks for the volunteer

(1)	Label the provided text snippets as relevant or not.
(2)	Label the provided images as relevant or not.
(3)	Label the provided combinations of the text snippet and the image snippet as relevant or not.
(4)	Choose some terms suggested by the system to refine the query.
(5)	Label the retrieved 200 Web pages as relevant or not.

ematics. All of them are familiar with search engines such as Google. Each query is conducted for three volunteers. Specifically, for each query, the tasks that a volunteer needs to complete are listed in Table VI.

These tasks are required to complete in sequence and a long time break is guaranteed between tasks. For Task 1-3 and Task 5, the list provided to the user is randomly shuffled. For Task 1, the text snippet of each Web page where the image snippet is available is provided. The time the user spends on Task 1 is recorded. For Task 2, all images appearing in Web pages are provided. The time the user spends on Task 2 is also recorded. For Task 3, according to the implementation of the Step 4 in the algorithm of Image Snippet Extraction, there are three types of image snippets available and thus three types of combinations are required to be labelled by the volunteer. Note that it is rather expensive to let volunteers complete such large amount of work. On average, it takes about two to three weeks for a volunteer to complete all the work including the break time.

4.2 Experiments on Result Presentation

This series of experiments are designed to test the helpfulness of introducing image snippets to the result presentation. The information gathered in Task 1 of Table VI is used to show the effect of the text snippets, and the time the user uses to complete this task is used to calculate the average speed the user reads a text snippet. The information gathered in Task 2 of Table VI is used to show the effect of the image snippets, and the time the user spends on this task is used to calculate the speed of reading an image snippet. Also, the information gathered in Task 3 of Table VI is used to show the effect of the combination of the text snippet and the image snippet.

Note that, in this experiment, the text snippet, the image snippet, and their combination would be presented to each volunteer in different stages. Some people may doubt that in the last stage, the volunteer has seen the snippets before, which may hurt the validity of this experiment. A clarification is given as follows. For the first two stages, the user has never seen the snippets provided to them, so the labels for the snippets and the time spent on labelling them are both valid. For the third stage, the user indeed has seen the text snippets and the image snippets provided to him/her, thus the time spent on labelling the snippets is invalid which is not used in this experiment. However, the user has never seen the combination between the image snippet and the text snippet which is guaranteed by the random order of presenting snippets in each stage, such that the labels based on the combination of text snippets and image snippets are valid.

Table VII. The contingency table

		Label of Web page	
		Yes	No
Label of Snippet	Yes	TP	FP
	No	FN	TN

Table VIII. The performance of different types of snippets for result presentation

QueryType	SnipType	p	r	$F1$	acc
Ambiguous	Img	0.700	0.649	0.674	0.924
	Txt	0.682	0.635	0.658	0.920
	Img+Txt	0.738	0.807	**0.771**	**0.942**
Unambiguous	Img	0.786	0.530	0.633	0.708
	Txt	0.661	0.792	0.721	0.708
	Img+Txt	0.751	0.855	**0.800**	**0.796**
Hot	Img	0.752	0.565	0.645	0.675
	Txt	0.741	0.852	0.793	0.767
	Img+Txt	0.784	0.850	**0.816**	**0.799**
Total	Img	0.756	0.562	0.645	0.778
	Txt	0.701	0.800	0.747	0.806
	Img+Txt	0.764	0.847	**0.804**	**0.852**

The labels of different snippets are compared with the real labels of Web pages to show the effect of the image snippets only, the text snippets only, and the combination of both when used to help the user perceive the content of the retrieved Web pages. The real labels of Web pages are collected during the Task 5 in Table VI. Since this experiment requires that each Web page has an image snippet, it is only conducted for the Web pages where the image snippets are available. There are 2,573 Web pages with the image snippets among the total 5,800 ($200 \times 29 = 5,800$) Web pages, and for each Web page the process is repeated for 3 volunteers. In total, the comparisons are conducted on 7,719 ($2,573 \times 3 = 7,719$) instances.

$$p = \frac{TP}{TP+FP} \qquad r = \frac{TP}{TP+FN}$$
$$F1 = \frac{2 \times p \times r}{p+r} \qquad acc = \frac{TP+TN}{TP+FP+FN+TN}$$

The standard precision, recall, F1 measure and accuracy are used as the performance measure. Given the contingency table (Table VII), precision (p), recall (r), F1 measure ($F1$) and accuracy (acc) can be determined as follows. Table VIII shows the performance of different types of snippets, where the type of the image snippet used is to combine $Rank_{Imp}$ and $Rank_{Sim}$ with equal weights. The best result is boldfaced.

Table VIII shows that only using the image snippet is worse than only using the text snippet. However, as expected, when the text snippet and the image snippet are combined simultaneously, the result better than each component alone can be obtained, which verifies that presenting both the text snippet and the image snippet helps users identify the Web pages they expect more accurately. This observation is consistent for different types of queries. Another interesting observation is that

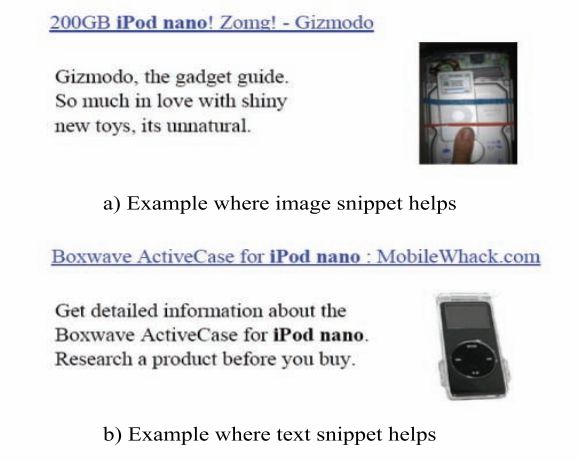


Fig. 4. Examples for combing text snippet and image snippet.

the precision of only using the image snippet is even better than only using the text snippet.

Fig. 4 illustrates the effect of the combination of the text snippet and the image snippet. These two items are both about the query 'ipod nano' and the information requirement for this query is 'any Web page that can help you buy the ipod nano mp3 player supposing you are planning to buy this product'. In Fig. 4(a), '200GB ipod nano' in the text snippet is sufficient to let most users believe that this Web page is introducing this mp3 player and should be relevant (some users familiar with ipod nano may feel strange about the huge disk volume). When the image snippet is given, things become clearer. This Web page is about how to connect the mp3 player with a hard disk and that is why the mp3 player could have 200GB volume. Obviously, this Web page is irrelevant. In Fig. 4(b), the image snippet convinces most users that this Web page is relevant, while the text snippet tells users that this Web page is introducing the plastic case used for protecting the ipod nano.

Table IX compares the performance of different types of image snippets, i.e., using $Rank_{Imp}$ only, using $Rank_{Sim}$ only, and combing them with equal weights, when they are used alone and combined with the text snippet. For the total 2,573 Web pages with the image snippet, there are 327 Web pages where the three types of image snippets are inconsistent, thus, in order to make the comparisons clearer, experiments concentrate on these 327 Web pages. Given that each Web page is labelled by three users, Table IX are obtained based on 981 ($327 \times 3 = 981$) instances. The best performance is boldfaced.

Table IX shows that in total the image snippet using $Rank_{Imp}$ only performs the best, although the other types follow closely. Since the importance-based image snippet is query independent which supports the off-line computation, it is feasible to be implemented in the practical search engine. The reported best performance further shows the promise of this type of image snippet. Due to the expensive cost of the user study, only one combination is tested which simply assigns equal weight

Table IX. The performance of different types of image snippets for result presentation

QueryType	SnipType	ImgSnipType	p	r	$F1$	acc
Ambiguous	Img	Imp	0.813	0.796	**0.804**	**0.949**
		Sim	0.848	0.571	0.683	0.930
		Comb	0.795	0.633	0.705	0.930
	Img+Txt	Imp	0.826	0.776	**0.800**	0.949
		Sim	0.944	0.694	**0.800**	**0.954**
		Comb	0.825	0.673	0.742	0.938
Unambiguous	Img	Imp	0.720	0.563	**0.632**	0.744
		Sim	0.754	0.542	0.630	**0.752**
		Comb	0.710	0.510	0.594	0.728
	Img+Txt	Imp	0.678	0.813	**0.739**	**0.776**
		Sim	0.676	0.740	0.706	0.760
		Comb	0.661	0.750	0.702	0.752
Hot	Img	Imp	0.829	0.472	0.601	**0.667**
		Sim	0.823	0.477	0.604	**0.667**
		Comb	0.789	0.497	**0.610**	0.661
	Img+Txt	Imp	0.810	0.697	0.749	0.751
		Sim	0.799	0.631	0.705	0.719
		Comb	0.795	0.754	**0.774**	**0.765**
Total	Img	Imp	0.791	0.544	**0.645**	**0.792**
		Sim	0.805	0.509	0.623	0.787
		Comb	0.766	0.521	0.620	0.779
	Img+Txt	Imp	0.766	0.741	**0.753**	**0.832**
		Sim	0.773	0.671	0.718	0.818
		Comb	0.754	0.741	0.748	0.827

Table X. The time spend on labelling different types of snippets

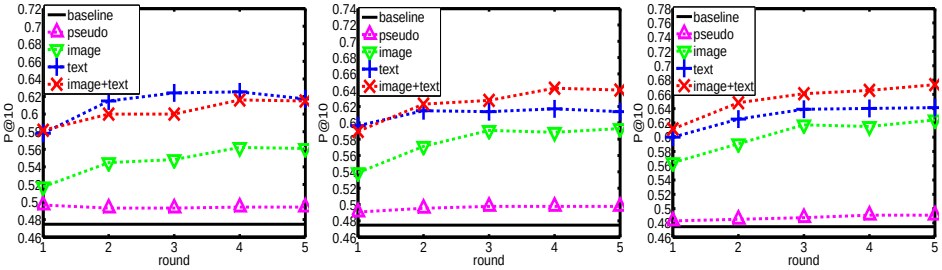
Time(second)	Text	Image
Ambiguous	5.684	1.727
Unambiguous	8.159	2.095
Hot	6.352	2.499
Average	6.657	2.130

to each component. Thus, it is difficult to say whether there are other combinations that perform better. For different query types, different types of image snippets perform inconsistently and it is hard to find a dominating type. However, it is clear that the difference of the performance is not very significant.

Table X compares the average time of labelling a text snippet and that of labelling an image snippet. It shows that labelling a text snippet is about 3 times slower than labelling an image snippet. This verifies the conclusions of previous research and supports the basic premise of this paper that humans can read 'imagery' faster than 'text'.

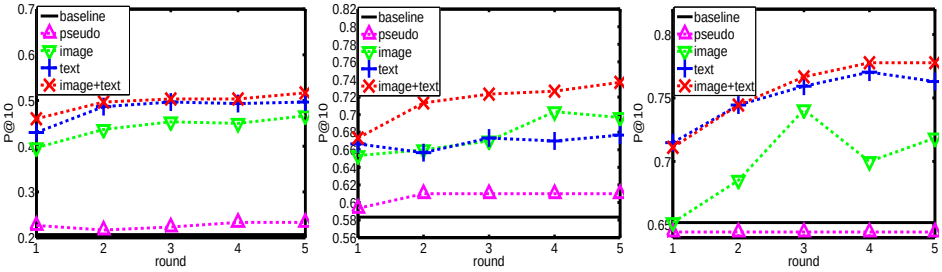
4.3 Experiments on Relevance Feedback

In the experiments on result presentation, it is shown that combining text snippets and image snippets can help the user identify the Web pages he/she expects more accurately. The following experiments are designed to further test whether the more accurate labels for the retrieved Web pages can improve the effect of relevance



(a) *FeedbackNum=5* (b) *FeedbackNum=7* (c) *FeedbackNum=10*

Fig. 5. The result of different types of snippets for relevance feedback



(a) *Ambiguous* (b) *Unambiguous* (c) *Hot*

Fig. 6. The result of different types of queries for relevance feedback with *FeedbackNum=10*

feedback. Since the volunteers have labelled the image snippets, the text snippets and the combination of them in Tasks 1-3 of Table VI, these labels can be directly used for feedback. Specifically, some highly ranked web pages are used for feedback. Since in practical scenario, the user provides his/her feedback only according to the snippet, the labels of these web pages used for feedback are provided by the labels of their image snippets, their text snippets and the combination of both, respectively. The retrieval performance is measured by P@10, i.e. the precision of the first 10 Web pages returned by the system. Here, the precision is calculated based on the relevant web pages labelled by the volunteers in Task 5 of Table VI. Note that the labels of different snippets are used for feedback and the labels of the web pages are used to evaluate performance. The above process can be continued for several rounds. The performance of 87 ($29 \times 3 = 87$) query-user combinations is averaged. This average performance with respect to different rounds is depicted in Fig. 5 and the *FeedbackNum* is set as 5, 7, 10, respectively. The retrieval performance of different types of queries with *FeedbackNum* setting as 10 is shown in Fig. 6. The performance of the initial query and the performance of pseudo feedback which simply regards all the Web pages used for feedback as relevant are also provided. Since the experiments also require that each Web page has an image snippet, it is only conducted for the Web pages where the image snippets are available.

Fig. 5 discloses that the combination of the text snippet and the image snippet performs better than each component alone in general. With *FeedbackNum* increases from 5 to 10, the effect of the combination of the two types of snippets is

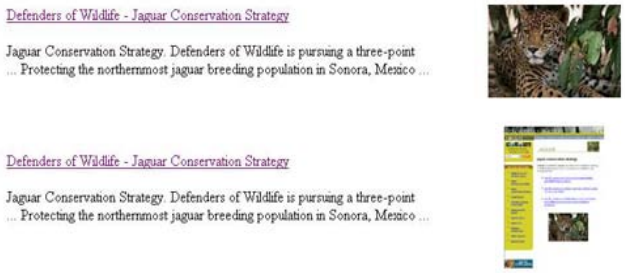


Fig. 7. The comparison of using the text snippet plus the the image snippet and the text snippet plus the thumbnail

more and more obvious. Different with Fig. 5 (b) and (c), Fig. 5 (a) shows that using the text snippet alone performs better than using the combination of the text snippet and the image snippet sometimes. A possible explanation can be given as follows. Although the experiments in the previous subsection show that using the combination of the text snippet and the image snippet helps the user to identify his/her expected web pages more accurately, this conclusion is reached from the statistical view. There are indeed some exceptions for which using the combination of the text snippet and the image snippet is less helpful than using the text snippet only. When the *FeedbackNum* is small, that is 5 here, once these exceptions appear, the performance will be influenced significantly, while when the *FeedbackNum* is big, the influence of these exceptions will be counteracted by more feedback examples for which the combination of both snippets is more useful. For different types of queries, it also performs better than each component alone consistently as shown in Fig. 6.

Note that the comparison with the combination of the text snippet and the thumbnail proposed by Woodruff et al. [2001] is not presented here. The reason lies in that when the thumbnail is used along with the text snippet, the possible space for displaying the thumbnail is limited, and in this situation the thumbnail could not provide much information about the content of the web page since it is too difficult for the user to read the thumbnail. Fig. 7 shows an example, where the first row shows an image snippet and the the second row shows the thumbnail of the same web page. All our volunteers reported that they could hardly comprehend the thumbnail. Note that, although Woodruff et al. [2001] have proposed some strategies for making the thumbnail more readable, they could not be very helpful in our situation given such a limited displaying space.

4.4 Experiments on Term Suggestion

This series of experiments are designed to compare the effect of the traditional term suggestion and the proposed image suggestion. Specifically, for each query, the suggested relevant terms and the suggested relevant images are used to reformulate the query respectively. This required information has been collected in Task 2 and Task 4 in Table VI. The implementation of the image suggestion has been discussed

in Section 3. Here, the document-based and the block-based image suggestion are both tested.

The term suggestion is implemented according to Local Context Analysis [Xu and Croft 2000]. Assume that the query to be expanded is Q, that the query terms in Q are $w_1, w_2, ..., w_m$, that the collection being searched is C, and that the set of top-ranked documents is $S = d_1, d_2, ..., d_n$. For each term t in the top-ranked documents, which doesn't appear in Q, a function $f(t, Q)$ measures how good a term t is for expanding query Q based on t's co-occurrence with w_i's in Q. All terms are ranked by $f(t, Q)$, and the *SugTermNum* (the number of terms used for term suggestion) terms with the highest ranks are displayed to the user. The user selects the relevant terms to be added to Q. In order to emphasize the initial query, when new terms are added, the frequency of the original terms in the initial query is doubled and the frequency of the newly added terms is set to one.

$$co(t, w_i) = \sum_{d \in S} tf(t, d) tf(w_i, d) \tag{2}$$

$$idf(t) = min(1.0, \log_{10}(N/N_t)/5.0) \tag{3}$$

$$codegree(t, w_i) = \log_{10}(co(t, w_i) + 1) idf(t) / \log_{10}(n) \tag{4}$$

$$f(t, Q) = \prod_{w_i \in Q} (\delta + codegree(t, w_i))^{idf(w_i)} \tag{5}$$

Here, $tf(t, d)$ and $tf(w_i, d)$ are the frequencies of t and w_i in document d, respectively. $idf(t)$ measures the inverted document frequency of t in the whole collection, N is the number of documents in C, N_t is the number of documents that contain t. $idf(w_i)$ is defined similarly.

In the experiments, S is set as the 5 highly ranked documents. Xu and Croft [2000] has reported that the performance of Local Context Analysis is relatively consistent when the size of S is within 200. Therefore in this paper the size of S is simply set to 5. δ is set to 0.1 according to the configuration in [Xu and Croft 2000]. *SugTermNum* is set to 5, since in practice there is no space on the result page to display many suggestion terms. Due to the cost of user study, other values of *SugTermNum* such as 7 and 10 are not tested in this experiment. Correspondingly, *SugImgNum* is also set to 5. The performance measure is still P@10. The same process continues for several rounds. The average result with respect to different rounds and the result for different query types are shown in Fig. 8 and Fig. 9 respectively. Besides the performance of the baseline which shows the performance of the initial query, the performance of the two pseudo methods is also provided. One is the pseudo-feedback which considers the top 5 documents as relevant and the other is the pseudo-term, which simply adds all the 5 terms suggested by Local Context Analysis to the initial query. Note that the experiments are conducted for all the Web pages.

Fig. 8 and Fig. 9 show that image suggestion performs better than term suggestion on average and for different types of queries. The main advantage of image suggestion over term suggestion is that more information is provided to the system to refine the query, since to a large extent the label of the image suggestion can

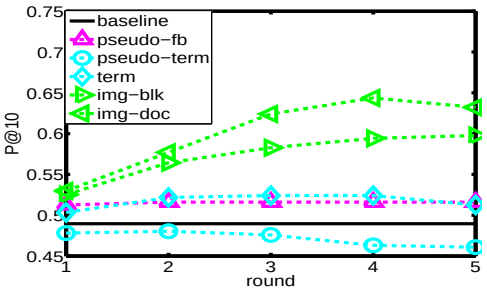
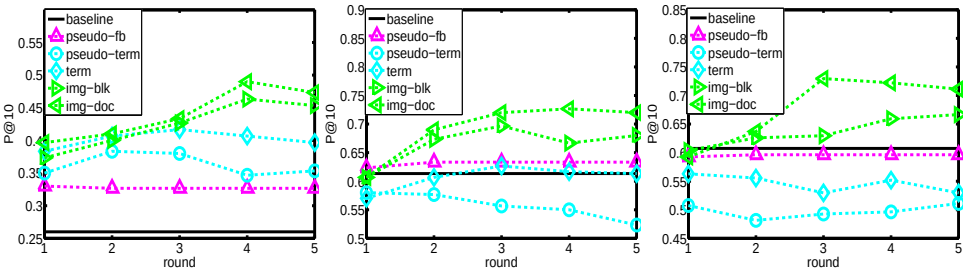


Fig. 8. The comparisons between term suggestion and image suggestion.



(a) Ambiguous *(b) Unambiguous* *(c) Hot*

Fig. 9. The result of different types of queries for image suggestion

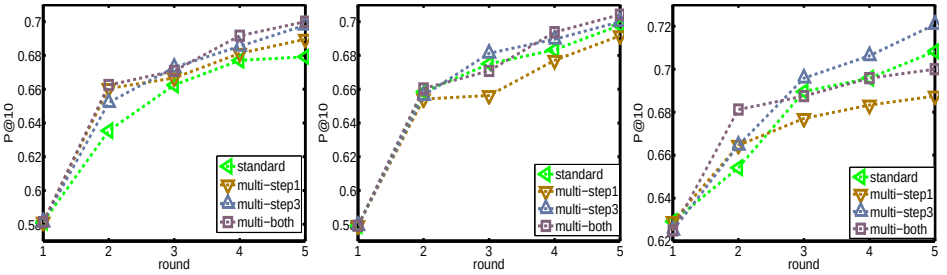
be considered as a relatively accurate label for its text description, which obviously gives more information than one or two terms. The document-based image suggestion performs better than the block-based image suggestion. A possible reason is that the top 200 Web pages returned by the Google search engine, which are used to form the dataset here, usually are about a single topic related to the query, thus the block-based image suggestion, which favors the multi-topic Web pages, naturally performs worse than the document-based image suggestion. For unambiguous and hot queries, the differences between relevant and irrelevant documents are not as significant as that in ambiguous queries. Thus, the existence of some misleading terms makes it difficult to choose appropriate terms to refine the query, which is supported by the poor performance of term suggestion on these two types of queries. In contrast, image suggestion still brings a significant improvement in this situation.

4.5 Experiments on Multimodal Image Suggestion

In this series of experiments, the promise of multimodal image suggestion is tested. As stated in Section 1 and 3, the relevant image snippets selected by the user can be naturally used as the image query which expresses the user's information need from the imagery modality. The system then can use the text query and the image query to obtain a rank of web pages, respectively. Finally, these two ranks can be combined to form a final one. The visual features used here is similar to [Zhang et al. 2005]. Specifically, a visual feature is a 144 dimensional vector (auto correlogram

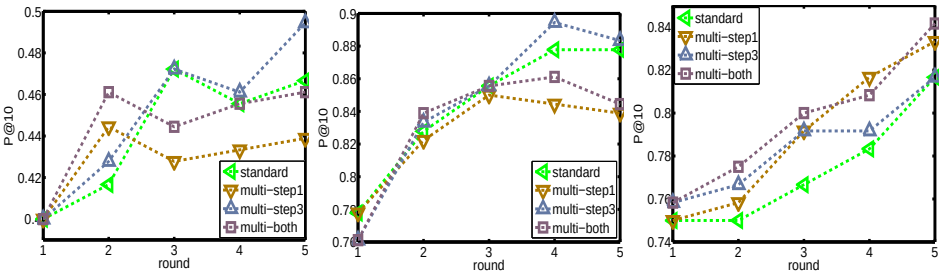
Table XI. The reduced set of queries

Type	Query
ambiguous	tiger, apple, dove, eagle, jaguar, jordan
unambiguous	tiger beer, apple fruit, dove chocolate, eagle bird, jaguar car, jordan basketball
hot	digital camera, ares, ipod nano, xbox



(a) SugImgNum=5 baseline=0.502 *(b) SugImgNum=7 baseline=0.502* *(c) SugImgNum=10 baseline=0.502*

Fig. 10. The result of multimodal image suggestions



(a) Ambiguous baseline=0.194 *(b) Unambiguous baseline=0.683* *(c) Hot baseline=0.692*

Fig. 11. The result of different types of queries for multimodal image suggestion with *SugImgNum=10*

computed over 36 quantized colors and 4 Manhattan Distances). The similarity between two images is measured by the distance between their corresponding vectors. The visual features widely used in the literature are still difficult to reflect the high-level semantics of images, thus this experiment focuses on the queries where the target has somewhat visual similarities. This reduced set of queries is listed in Table XI.

For Step 1 in Table IV, $ImgRank_{txt}$ and $ImgRank_{img}$ are combined with equal weights. For Step 3 in Table IV, the Web pages without the image snippets are simply put at the end of $Rank_{img}$. When $Rank_{txt}$ and $Rank_{img}$ are combined, a relatively conservative strategy is used: if the image snippet of a Web page ranks within top 10 in $Rank_{img}$, the final rank of this Web page is changed to the combination of its two ranks in $Rank_{txt}$ and $Rank_{imp}$; otherwise, the final rank is set as the rank in $Rank_{txt}$. The document-based standard image suggestion

is used for comparison here. Three versions of multimodal image suggestion are implemented: the first only uses the multimodal information in Step 1 of Table IV; the second only uses the multimodal information in Step 3 of Table IV; and the third uses this information in both steps. The performance of the initial query is provided as baseline. Fig. 10 shows the performance with respect to *SugImgNum* and Fig. 11 shows the performance of different types of queries with *SugImgNum* setting as 10.

Fig. 10 and Fig. 11 show that introducing the visual features indeed improves the performance of the standard image suggestion with respect to different values of *SugImgNum* and different types of queries, although sometimes the improvement is not very significant. It is not very clear which version of the multimodal image suggestion performs the best. In most cases, the two versions which uses imagery in Step 3 and uses imagery in both Step 1 and Step 3 perform better.

5. USER STUDY

In addition to the experiments reported in the previous sections, we have developed an illustration system of WebSIS[5] and deployed it to enable interested users to have some experience with WebSIS and collect their subjective feedbacks.

The illustration system was developed based on 2,573 indexed Web pages where the image snippets are available among the total 5,800 Web pages used in the previous experiments. Note that although the vector space model was used in the previous experiments, in our illustration system the search model is simply based on query word match. The standard inverted table [Baeza-Yates and Ribeiro-Neto 1999] is used to index the words appearing in Web pages. According to the results in Table IX, the importance-based image snippet is adopted. Since the importance-based image snippet is query-independent, the image snippet can be generated off-line and indexed. Therefore, the efficiency of the on-line performance of the system is not bad. Considering the results of Fig. 8 and 9, the current illustration system supports document-based image suggestion. Currently, the implementation of the image suggestion does not use the visual information of the image snippets.

After users tried the illustration system, they were invited to complete a feedback form. The feedback form consists of two parts. The first part asks for some basic information including age, occupation and familiarity with search engines. The second part asks for users' subjective opinions on the usefulness of image snippets. There are five questions in total. For each question, the user is asked to give a score in the range of 1-5. The questions are listed in Table XII.

41 users have provided their feedback till Aug.10, 2007. Among these users, one user is among the 10-20 age range, 37 users are among the 20-30 age range and three users are among the 30-40 age range. As for the occupation, 27 users are college students with different majors such as computer science, law, chemistry, accounting, finance, biology, etc. The other 14 users include software engineer, university professor, accountant, government official, lawyer, migration agent, etc. Most of these users are familiar with search engines. Users were asked to select a score from 1 to 5 (from unfamiliar to familiar) to measure their familiarity with search engine. 35 users chose 4 or 5, six users chose 2 or 3 while no one chose

[5]http://lamda.nju.edu.cn/websis

Table XII. A list of questions for users

(1) Do you think providing the image snippet for each returned web page is useful? (score: 1-5, from useless to useful)
(2) Compared with understanding text snippet, do you need more or less time to understand the image snippet? (score: 1-5, from less to more)
(3) Do you think the utility of image suggestion is useful? (score: 1-5, from useless to useful)
(4) Do you think selecting the relevant images for image suggestion is difficult? (score: 1-5, from easy to difficult)
(5) Compared with your previous search experience, do you think it is slower or faster to display the search result? (score: 1-5, from slower to faster)

Table XIII. Average Score for Questions 1-4

	Snip Useful (Q1)	Snip Time (Q2)	Sug Useful (Q3)	Sug Time (Q4)
All (41)	4.12	2.51	4.15	2.61
Student (27)	4.07	2.33	4.07	2.48
Not Student (14)	4.21	2.86	4.29	2.86
CS (25)	4.20	2.00	4.20	2.36
Not CS (16)	4.00	3.31	4.06	3.00

1. Considering these users' ages and familiarity with search engines, they properly represent the group of people who use the search engine very often and can provide a meaningful comparison between WebSIS and the style of traditional search engines. Moreover, considering the diversity of these users' specialities, they may be able to represent people with different backgrounds and cognitive styles.

The average scores of these 41 users for Questions 1-4 are shown in the first row of Table XIII. Then, these 41 users are divided into 2 groups according to occupation, one includes students and the other includes the rest. The average scores of Questions 1-4 of these two groups are reported, respectively. Similarly, these users are also divided into two groups according to speciality, one is "computer science related" and the other is the rest. The former group includes not only students majoring in computer science but also users whose job is related to computer science such as software engineers. All these results are shown in Table XIII. The numbers shown in the brackets are the number of users in the group.

From Table XIII we can know that "Not student" users need to spend more time than "Student" users to understand image snippets, and users without background on computer science need to spend more time than users with background on computer science to understand image snippets. This is not strange since "Student" and "CS" users usually use search engines more frequently than the others, thus they can get acquaint with image snippets more quickly. Although "Not student" and "Not CS" users need more time to comprehend the image snippets, the time cost is yet not more than that needed for them to comprehend the text snippets. It is impressive that for any group of users, the average scores on the usefulness of image snippets and image suggestion are higher than 4.0. In general, the results show that the users felt that the image snippet and the image suggestion were useful,

they required less time to understand the image snippet compared with reading the text snippet, and labelling the image snippet for image suggestion was not difficult.

The question 5 in Table XII is designed to provide a reference for the influence of using the image snippet on the response time. The average score of 41 users is 3.51, which implies that the users felt that the speed for displaying the image snippets is comparable to displaying only the text snippets. Note that in the illustration system we have generated and indexed the image snippets off-line. If the image snippets were not generated and indexed in advance, the speed for displaying the image snippets will be much slower. Considering that all commercialized search engines use indexes to help improve the efficiency, we think that this is feasible if the proposal of WebSIS is incorporated into these search engines.

6. CONCLUSION

In the previous research on interactive Web search, little attention is paid to using other modalities, especially imagery. The comparison of two modalities, text and imagery, shows that the time for 'reading' an image is as little as that for reading one or two words, while the information brought by an image is as much as that expressed by a whole passage of text. Considering the above advantages of imagery, a new interactive Web search framework called WebSIS is proposed, which extends a recent research [Xue et al. 2006] where images are used to help improve the performance of interactive Web search. Specifically, the image snippets are extracted from Web pages and then they are provided along with text snippets to the user for result presentation and relevance feedback and also presented alone to the user as image suggestion. Experiments show that using the image snippets in interactive Web search helps users to identify the Web pages they expect and to reformulate the initial query more effectively and efficiently. Further experiments on exploiting the visual features of the image snippets have demonstrated the promise of incorporating the multimodal techniques into WebSIS.

In WebSIS, the result page consists of both text content and images, thus transferring such an Web page on Internet to respond a user's query requires more time. How much the response time will be increased and how this increased response time will affect the user's satisfaction for the search engine are both important issues to be studied in the future. In the current implementation of WebSIS, the image suggestions are ranked using some simple strategies. It is anticipated that some more principled techniques, such as active learning, are useful to further improve the performance. Moreover, the current implementation of WebSIS has not exploited visual features of the image snippets very effectively. Adopting techniques such as these described in [Zhou and Dai 2007] to improve the exploitation of image information is another interesting future work. Note that although image suggestion performs better than term suggestion in many cases, image suggestion could not fully replace text suggestion. For example, when a retrieved Web page contains no images at all, image suggestion could not help while term suggestion still performs well. Therefore, how to further combine these two techniques is also an interesting future work.

Acknowledgements

We want to thank the anonymous reviewers and the associate editor for their helpful comments and suggestions.

REFERENCES

Anick, P. 2003. Using terminological feedback for web search refinement: A log-based study. In *Proceedings of the 26th ACM International Conference on Research and Development in Information Retrieval*. Toronto, Canada, 88–95.

Anick, P. and Tipirneni, S. 1999. The paraphrase search assistant: Terminological feedback for iteractive information seeking. In *Proceedings of the 22nd ACM International Conference on Research and Development in Information Retrieval*. Berleley, CA, 153–159.

Baeza-Yates, R. and Ribeiro-Neto, B. 1999. *Modern Information Retrieval*. Addison-Wesley, Wokingham, UK.

Barnard, K. and Johnson, M. 2005. Word sense disambiguation with pictures. *Aritificial Intelligence 167,* 12, 13–30.

Cai, D., He, X.-F., Li, Z.-W., Ma, W.-Y., and Wen, J.-R. 2004. Hierarchical clustering of www image search results using visual, textual and link analysis. In *Proceedings of the 12th ACM International Conference on Multimedia*. New York, NY, 952–959.

Cai, D., Yu, S.-P., Wen, J.-R., and Ma, W.-Y. 2003. Vips: A vision-based page segmentation algorithm. Tech. Rep. No. MSR-TR-2003-79, Microsoft.

Chakrabarti, S. 2003. *Mining the Web: Discovering Knowledge from Hypertext Data.* Morgan Kaufmann: San Francisco, CA.

Chapman, A. 1993. *Making Sense: Teaching Critical Reading Across the Curriculum.* The College Board: NY.

Coltheart, V. 1999. *Fleeting Memories: Cognition of Brief Visual Stimuli.* MIT Press: Cambridge, MA.

Dennis, S., Bruza, P., and McArthur, R. 2002. Web searching: A process-oriented experimental study of three interactive search paradigms. *Journal of the American Society for Information Science and Technology 53,* 2, 120–133.

Fan, X., Xie, X., Li, Z., Li, M., and Ma, W.-Y. 2005. Photo-to-search: Using mutlimodal queries to search the web from mobile devices. In *Proceedings of the 7th ACM SIGMM International Workshop on Multimedia Information Retrieval*. Singapore, 143–150.

Huang, C.-K., Chien, L.-F., and Oyang, Y.-J. 2003. Relevant term suggestion in interactive web search based on contextual information in query session logs. *Journal of the American Society for Information Science and Technology 54,* 7, 638–649.

Jaimes, A., Christel, M., Gilles, S., Sarukkai, R., and Ma, W.-Y. 2005. Multimedia information retrieval: What is it, and why isn't anyone using it? In *Proceedings of the 7th ACM SIGMM International Workshop on Multimedia Information Retrieval*. Singapore, 3–8.

Jing, Y. and Croft, W. 1994. An association thesaurus for information retrieval. In *Proceedings of the Intelligent Multimedia Information Retrieval Systems*. New York, NY, 146–160.

Joachims, T. 2002. Optimizing search engines using clickthrough data. In *Proceedings of the 8th ACM International Conference on Knowledge Discovery and Data Mining.* Alberta, Canada, 133–142.

Jones, S. and Staveley, M. 1999. Phrasier: A system for interactive document retrieval using keyphrases. In *Proceedings of the 22nd ACM International Conference on Research and Development in Information Retrieval*. Berleley, CA, 160–167.

Kelly, D. and Belkin, N. 2004. Display time as implicit feedback: Understanding task effects. In *Proceedings of the 27th ACM International Conference on Research and Development in Information Retrieval*. Sheffield, UK, 377–384.

Kelly, D. and Teevan, J. 2003. Implicit feedback for inferring user preferrence. *SIGIR Forum 37,* 2, 18–28.

Koenemann, J. and Belkin, N. 1996. A case for interaction: A study of interactive information retrieval behavior and effectiveness. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Vancouver, Canada, 205–212.

Kraft, R. and Zien, J. 2004. Mining anchor text for query refinement. In *Proceedings of the 13th International Conference on World Wide Web*. New York, NY, 666–674.

Qiu, Y. and Frei, H.-P. 1993. Concept based query expansion. In *Proceedings of the 16th ACM International Conference on Research and Development in Information Retrieval*. Pittsburgh, PA, 160–169.

Ruthven, I. 2003. Re-examining the potential effectiveness of interactive query expansion. In *Proceedings of the 26th ACM International Conference on Research and Development in Information Retrieval*. Toronto, Canada, 213–220.

Shen, X., Tan, B., and Zhai, C. 2005. Context-sensitive information retrieval using implicit feedback. In *Proceedings of the 28th ACM International Conference on Research and Development in Information Retrieval*. Salvador, Brazil, 43–50.

Silverstein, C., Henzinger, M., Marais, H., and Moricz, M. 1998. Analysis of a very large AltaVista query log. Tech. Rep. No.1998-014, Digital Systems Research Center.

Song, R.-H., Liu, H.-F., Wen, J.-R., and Ma, W.-Y. 2004. Learning block importance models for web pages. In *Proceedings of the 13th International Conference on World Wide Web*. New York, NY, 203–211.

White, R. W., Ruthven, I., and Jose, J. M. 2005. A study of factors affecting the utility of implicit relevance feedback. In *Proceedings of the 28th ACM International Conference on Research and Development in Information Retrieval*. Salvador, Brazil, 35–42.

Woodruff, A., Faulring, A., Rosenholtz, R., Morrison, J., and Pirolli, P. 2001. Using thumbnails to search the web. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Seatle, WA, 198–205.

Xu, J. and Croft, W. 2000. Improving the effectiveness of information retrieval with local context analysis. *ACM Transactions on Information Systems 18,* 1, 79–112.

Xue, X.-B., Zhou, Z.-H., and Zhang, Z. 2006. Improve Web search using image snippets. In *Proceedings of the 21st National Conference on Artificial Intelligence*. Boston, MA, 1431–1436.

Yang, J., Li, Q., and Zhuang, Y. 2002. Octopus: Aggressive search of multi-modality data using multifaceted knowledge base. In *Proceedings of the 11th International World Wide Web Conference*. Honolulu, Hawaii, 54–64.

Zhang, R., Zhang, Z., Li, M., Ma, W.-Y., and Zhang, H.-J. 2005. A probabilistic semantic model for image annotation and multi-modal image retrieval. In *Proceedings of the 10th IEEE International Conference on Computer Vision*. Beijing, China, 846–851.

Zhou, Z.-H. and Dai, H.-B. 2007. Exploiting image contents in web search. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*. Hydrabad, India, 2928–2933.