The Potential of User Feedback Through the Iterative Refining of Queries in an Image Retrieval System

Maher Ben Moussa, Marco Pasch, Djoerd Hiemstra, Paul van der Vet, and Theo Huibers

University of Twente
P.O. Box 217, 7500 AE Enschede, The Netherlands
{m.benmoussa,m.pasch,d.hiemstra,p.e.vandervet,
t.w.c.huibers}@cs.utwente.nl

Abstract. Inaccurate or ambiguous expressions in queries lead to poor results in information retrieval. We assume that iterative user feedback can improve the quality of queries. To this end we developed a system for image retrieval that utilizes user feedback to refine the user's search query. This is done by a graphical user interface that returns categories of images and requires the user to choose between them in order to improve the initial query in terms of accuracy and unambiguousness. A user test showed that, although there was no improvement in search time or required search restarts, iterative user feedback can indeed improve the performance of an image retrieval system in terms of user satisfaction.

1 Motivation

One problem of image retrieval is that users utilize inaccurate or ambiguous expressions in their queries. If a user has an image of a three story building in mind but just types in "house", the system will probably give results that satisfy the query but not the user because he will see a lot of houses that do not resemble the one he had in mind. On the other hand the word "bank" can refer to a financial institute as well as a dam protecting the country from a river or sea. Existing retrieval systems require the user to have certain knowledge about the operators the system employs and that he or she is required to use a certain precision and unambiguousness in his queries. Or as Baeza-Yates and Ribiero-Neto put it:" The user of a retrieval system has to translate his information need into a query in the language of the system" [1]. When a search result turns out insufficiently the user has to restart his search with a new, refined query. It can be a long process until the query finally matches both what the user has in mind and the representation of the retrieval system.

We assume that by implementing iterative user feedback on the result of a search we can improve the searching process. Relevance feedback helps the user refining the query without requiring sophisticated usage of the system's query language [2, 3]. Our goal is to develop a system in which the user is not required to type a very specific query. Instead the system will guide him through a number of feedback steps where he can refine his search by simply clicking on a category of items that is close

S. Marchand-Maillet et al. (Eds.): AMR 2006, LNCS 4398, pp. 258–268, 2007.

until the results are satisfactory. The user does not have to care about the phrasing of his query and simply has to click on a particular item. In terms of user friendliness, the system "forgives" errors (i.e. here: imprecise input) and this improves the usability of the image retrieval system as a whole with user satisfaction in particular.

We implement this idea by adding a categorization system on the collection of an image retrieval system. When the user types in a request the result is a number of categories that seem promising to contain the image that he is searching for. The user can then easily make a distinction for example for the word "bank" between "sea/bank" and "organization/bank". In this way, the query becomes iteratively unambiguous and more precise. As such, the approach bears some resemblance with browsing concept hierarchies [4] such as those provided by for instance the Open Directory Project [5]. A similar approach was followed successfully by Sieg et al. [6] for web search. Like Sieg et al., we use static categories, but categories might as well be taken from content classifiers as we have done recently for text search [7]. In the near future, image content classifiers will be available by collaborative efforts [8], which can be used directly in our approach to iterative user feedback.

The paper is organized as follows. In Section 2, we introduce our research questions. In Section 3 we describe our interactive image retrieval system and our approach to answer the research questions. Section 4 presents the experimental results, which are discussed further in Section 5. Finally, Section 6 concludes the paper.

2 Research Questions

In this study we investigate the potential of iterative user feedback in an image retrieval system. We think of potential here as an improvement in terms of the time needed to conduct a search, the attempts that a user has to make to conduct a search and the user's personal attitude of the retrieval system. This leads to the following research questions that we want to investigate:

1. Does iterative user feedback improve the performance of an image retrieval system in terms of the time that is needed for a search?
2. Does iterative user feedback improve the performance of an image retrieval system in terms of the results of a search and the need to restart a search?
3. Does feedback improve the quality of an image retrieval system as it is perceived by the user?

3 Methods

3.1 Prototype

In order to test our assumptions that were stated above we built a system that incorporates user feedback. For control reasons we also built a system that resembles a "traditional" image retrieval system like Google or AltaVista. Both systems feature the same search engine, built on top of Lucene of The Apache Software Foundation [9], and access the same collection. In preparation of the test we also created a domain-specific collection of image data.

Lucene is a lightweight core of a search engine, with a simple and clear API for searching and indexing information. The main disadvantage of Lucene is that it is a very light API with no API for web-crawling and that it lacks support for different file formats like images or PDF files. However, because of the simplicity of its API, it can be easily customized and support for different files can easily be added. For this study, support for image files and a lightweight web crawler have been added.

$$\text{score}(d,q) = \sum_{t \text{ in } q} tf(t \text{ in } d) \cdot idf(t) \cdot boost(t.field \text{ in } d) \cdot lengtNorm(t.field \text{ in } d)$$

Although Lucene is a lightweight search engine core it contains a reasonably sophisticated scoring algorithm. This is the score formula that is used by Lucene to determine the score for each document based on a query. An explanation of the formula is given in Table 1. The formula is taken from [9].

Table 1. Score formula of Lucene

Factor	*Description*
$tf(t \text{ in } d)$	Term frequency factor for the term (t) in the document (d)
$idf(t)$	Inverse document frequency of the term
$boost(t.field \text{ in } d)$	Field boost, as set during indexing
$lengthNorm(t.field \text{ in } d)$	Normalization value of a field, given the number of terms within the field. This value is computed during indexing and stored in the index

The goal of this study is not to investigate how to improve the indexing of images. We assume that there is a search engine that can index all the images perfectly with the correct keywords and the correct description. We are aware that in reality such an image search engine does not exist. For instance, the image search of Google.com has problems with indexing images with the right keywords. To realize the goal of this study the choice was made to use a test collection of images from the stock photo provider FotoSearch.com [10]. All the images in this collection have proper description and correct keywords.

The next step was to develop an indexer that would be able to index this collection. To this end a FotoSearch.com specific indexer was developed that reads a list of URLs containing an image, its description and its keywords (e.g. http://www.fotosearch.com/BNS145/fba001/) from a file and indexes it. The indexer parses a site and retrieves only the correct information about an image and ignores the rest.

For the search JSP pages are created that use the Lucene search API for searching. To get better results, the StopAnalyser class of Lucene is used to parse the user query. The StopAnalyser class removes all English stop words from the query to decrease the change for irrelevant results. After executing the Lucene search API, the results are presented to the user in a similar way as in the Google Search, in order to provide the user with an interface that he is familiar with. This search engine serves as the control condition in the experiment. Its architecture is shown in Figure 1.

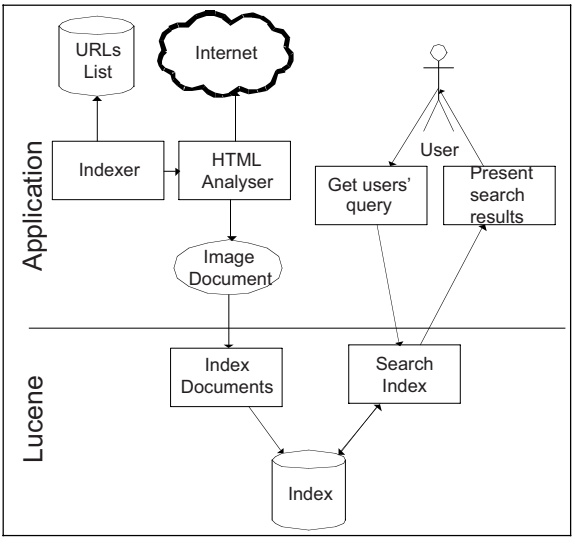

Fig. 1. Architecture of the control condition

The experimental condition was realized by extending the first prototype with support for categories. Based on the test collection of images, a category structure has been created by hand. An XML file with this category structure was built, containing category names, search queries related to these categories and images representing them. When the user types a query, the search engine does not directly pass the query to Lucene like done in the control condition, but searches in the categories XML file for the categories containing the words from this query and the words similar to them and returns the results in a list format with an image before every category that represents it. After choosing a category the user can refine his searching by continuously clicking on the desired subcategory, until he is satisfied with the results. When a (sub-) category is chosen, the search engine automatically executes the query related to this category and presents the results in the same page. The architecture of the experimental condition is shown in Figure 2.

As a test collection we created a category structure of people, with subcategories like Caucasian, Asian and African-american which have as subcategories child, teenager, young adult and older. Here the following subcategories are established by gender and the number of people in an image. To prevent confusion only images related to these categories were indexed and used by both prototypes. In total the collection consisted of 550 images. We are aware that this is a rather small collection, but we are convinced that it is sufficient for the tasks that had to be solved in the user test, which is described further below.

Another difficulty that had to be solved was that different users use different words to specify their search queries, which is also known as encoding specificity problem. Some people would use “caucasian” to search for Caucasian people, while others would use “white” for the same purpose. To solve this problem, WordNet had been

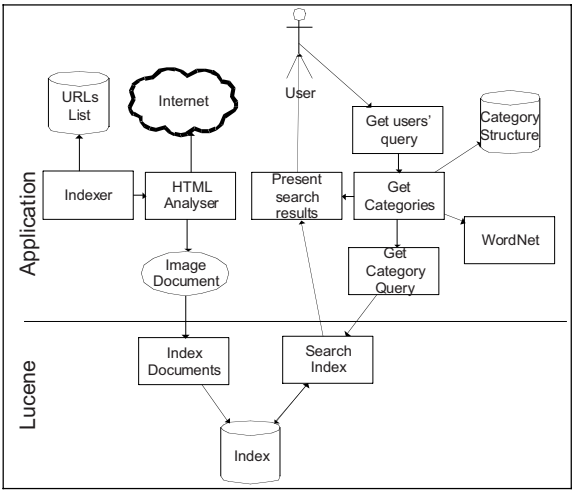

Fig. 2. Architecture of the experimental condition

integrated to the second prototype, making it possible to retrieve the category "caucasian" when typing "white". Although WordNet worked pretty well, there were some cases where it presented fewer alternatives than we expected. For instance, when searching for "white", "caucasian" was returned as one of the word senses. However when searching for "black", "african-american" was not between the related word senses. Although it is returned in the similarity list of the adjective "black", there was not enough time to develop a smarter application that checks all the words related in WordNet to the source word and also determines which ones are relevant and which are not. The developed prototype only uses the word senses relations in WordNet.

3.2 Design

To evaluate our approach we conducted a user study choosing a within-subjects setup, which means we use the same participants for both test conditions. The advantage of employing the within-subject setup is that we need fewer participants than with a between-subjects setup, where participants are only used for one condition. More important, this setup usually has a higher statistical power than using different participants for the test conditions, as we get values for each condition from the same individuals.

There was an experimental condition with the system employing user feedback and a control condition with the system without user feedback. For both conditions the participants had to search for images that they were shown before. Those images were chosen by random. The tasks had to be assigned visually because assigning the tasks verbally would have had an influence on which keywords the participants would have used to find the images.

The participants first had to search for images in the control condition and then afterwards with the experimental condition. Usually it is recommended to alternate the order of conditions to avoid learning effects of the participants [11]. In our case it did not appear useful to alternate the order of conditions because users can see the names

of categories in the experimental condition and this could have given them cues for keywords which would have influenced the scores in the control condition. On the other hand the control condition does not give cues for working with the experimental condition. This made it possible to use the within-subject setup.

3.3 Participants

In their HCI textbook, Dix et al suggest using at least 10 participants in each condition of a user test to be able to perform statistical analysis on the results [12]. A total of 12 participants, 5 females and 7 males were recruited for this study. They were all university students, with ages ranging from 20 to 25 years.

As Dix et al further point out it is not useful to test a system that is intended for the general public on a group of computer science students because they are not representative for the common user. We coped with this by relying on students from various fields. Only one participant in our study had a background in computer science, the others came from all kinds of disciplines. All of them can be described as regular Internet users.

3.4 Procedure

Participants were asked to separately enter a room where a computer running the test systems was located. They were asked to sit in front of the computer with an observer next to them. Then they were given a short introduction what the test was about without going too much into detail to avoid giving any cues that might influence their behaviour.

Then they were shown an image and asked to search for this image with the first system. In total they had to search for three images for each system. While they were searching, notes were taken on how long it took them to search and how often they had to restart their search by entering new queries. After finishing the tasks they were asked to evaluate the two systems by describing how well they could work with each system and by assigning grades. The grades were oriented on the Dutch grading scale where a 10 represents the maximum and a 1 the minimum.

In the end a short debriefing and a short discussion about the usefulness of our approach took place. Interesting points deriving from those discussions are mentioned in the discussion section of this report.

4 Results

The first research question asked whether user feedback improves the performance of an image retrieval system in terms of the time users need to search for images. Regarding this research question we can state the following hypothesis:

H_1: A system utilizing user feedback will be faster in use than a system without user feedback.

This leads to the null hypothesis

N_1: There is no difference in time when using a system with or without user feedback.

During the experiment notes were taken how long it took the participants to fulfill the tasks that were given to them. Table 2 shows the results. Note that the table shows the accumulated times for the three tasks that had to be solved with each of the conditions.

Table 2. Time needed to fulfill tasks (in seconds)

Subject	*Control C.*	*Exp. C.*	*Diff.*	*Diff.^2*
1	235	120	115	13225
2	380	315	65	4225
3	180	260	-80	6400
4	80	130	-50	2500
5	80	80	0	0
6	165	90	75	5625
7	390	400	-10	100
8	120	180	-60	3600
9	120	280	-160	25600
10	240	190	50	2500
11	220	120	100	10000
12	210	120	90	8100
Sums	2420	2285	135	81875
Means	201,67	190,42	11,25	

The time needed to solve the given tasks with the system that employs user feedback was slightly shorter ($\bar{x}$ = 190.42) than without feedback ($\bar{x}$ = 201.67). However, when running a student's t test, the difference did not support the hypothesis that a system employing iterative user feedback is faster in use than a system without feedback since the obtained value t = 0.46 is not statistically significant at the 5% level.

The second research question was whether iterative user feedback improves the search results. This can be interpreted in many ways. In our experimental setup we think of an improvement as a reduced need to restart the search with a new query. Our hypothesis is that

H_2: A system utilizing user feedback reduces the need to reinitiate searches than a system without user feedback.

The according null hypothesis is

N_2: There is no difference in the number of searches that have to be carried out to fulfill a task.

Notes were taken how often participants had to enter new queries and by this reinitiate their searches. The resulting figures are given in table 3. Again the values for the three single tasks that a participant had to fulfill per condition are accumulated.

Table 3. Number of search attempts

Subject	*Control C.*	*Exp. C.*	*Diff.*	*Diff.^2*
1	7	3	4	16
2	8	7	1	1
3	3	7	-4	16
4	3	3	0	0
5	3	4	-1	1
6	5	3	2	4
7	10	9	1	1
8	4	3	1	1
9	5	5	0	0
10	5	3	2	4
11	6	4	2	4
12	7	5	2	4
Sums	66	56	10	52
Means	5,50	4,67	0,83	

The table shows that in the experimental condition less restarts took place ($\bar{x}$ = 4.67) than in the control condition ($\bar{x}$ = 5.5). However, when again running a t test, the difference did not support the hypothesis that the need to reinitiate searches in a system employing user feedback is smaller than with a system not utilizing user feedback as the null hypothesis could not be rejected at 5% significance level (t=1.45, degrees of freedom=11, p>0.05).

The third research question asked whether user feedback improves the quality of a system as it is perceived by the user. In other words we are interested in knowing whether users like to work with a system employing feedback. This can be rephrased into

H_3: Users rate a system utilizing user feedback higher than a system without user feedback.

The adequate null hypothesis is

N_3: There is no difference in user ratings of a system with and a system without feedback.

After the participants had worked with both systems in the experiment they were asked to rate them on a scale from 1 (low) to 10 (high). Table 4 shows these ratings.

As can be seen in the table the ratings of the experimental condition are higher ($\bar{x}$ = 8.25) than those of the control condition ($\bar{x}$ = 7.33). Here the difference does indeed support the hypothesis that users perceive a higher quality in a system employing iterative user feedback. The t test reaches significance at 5% significance level (t=-2.42, df=11, p>0.05). The null hypothesis is thus rejected and we can conclude that user feedback does improve the quality of a system as it is perceived by the user.

Table 4. User ratings

Subject	*Control C.*	*Exp. C.*	*Diff.*	*Diff.^2*
1	7	10	-3	9
2	8	10	-2	4
3	9	8	1	1
4	8	9	-1	1
5	8	7	1	1
6	6	8	-2	4
7	7	8	-1	1
8	6	8	-2	4
9	8	7	1	1
10	7	8	-1	1
11	7	8	-1	1
12	7	8	-1	1
Sums	88	99	-11	29
Means	7,33	8,25	-0,92	

5 Discussion

We were able to show that user feedback does improve the quality of an image retrieval system as it is perceived by the user. The majority of the participants in this study stated that they at least liked the opportunity to use the categories in addition to restart their searches. One participant even declared that he would like to solely navigate through the categories once he had started his search. This user also indicated that he liked working with similar systems like the Yahoo directories to search on the internet. Other participants said they were irritated at first, being used to minimalist interfaces like Google, but once they understood the category system most of them appreciated the extra search options offered by it.

This initial irritation when first using the system is in our opinion also the reason why no significant differences could be found in terms of speed and search restarts. We motivate this position on the fact that tendencies towards this, though not significant, could be found indeed and on the statements of some participants following the experiment. They indicated that they at first had problems in understanding the category system, but this improved with growing experience. We can also conclude from our observations that some of the participants had difficulties in distinguishing categories and images themselves.

Figure 3 shows a screenshot of our image retrieval system that employs user feedback. The first row shows the sub-categories that can be reached from the current category. Beneath this row pictures matching the current query are presented. The confusion between categories and images can be explained by their visual similarity. This can be seen as a minor flaw in our prototype and the first thing we would correct for further research.

Fig. 3. Screenshot of the image retrieval system employing feedback

We do think that with an improved prototype and participants that have had more experience with retrieval systems that employ iterative user feedback it is possible to find statistically significant differences for the time users need to solve tasks and for the need to reinitiate a search, i.e. entering new queries.

Our results are confirmed by similar work done by Nemeth et al [13]. They examined methods for improving users' queries, specifically interactive and automatic query expansion, coming to the conclusion that there is a difference between users' preferences and the real performance of systems using those improvement methods. While the user satisfaction was higher with the query expansion systems, the performance did not differ significantly.

6 Conclusions

This study has shown that iterative user feedback improves the perceived quality of an image retrieval system. After a short period of getting acquainted with the categorization system users prefer the additional searching options that come with it.

A significant improvement of the performance in terms of search times and search restarts could not be found. Reason for this might be a flaw in the prototype, namely that users confused categories of images and images themselves. This was identified during the user test. Another reason might be the lack of experience of the test participants with search engines that employ iterative user feedback by means of a categorization system. Observations during the experiment show that many users were

able to work more efficiently once they had figured out the categorization system, i.e. to search faster and needing less search attempts. Of course also the possibility remains that there is no necessary correlation between the perceived performance and the real performance of retrieval systems. Further research with an improved prototype and users that have a little more experience with iterative feedback systems could answer the question whether the real performance can indeed be improved analogue to the perceived user satisfaction as we were able to show in this study.

References

1. Baeza-Yates, R., Ribiero-Neto, B.: Modern Information Retrieval. Addison-Wesley (1999)
2. Salton, G., Buckley, C.: Improving retrieval performance by relevance feedback. Journal of the American Association of Information Science 41(4) (1990) 288–297
3. Harman, D.: Relevance feedback revisited. In Belkin, N., Ingwersen, P., Pejtersen, A., eds.: Proceedings of the 15th International ACM SIGIR Conference on Research and Development in Information Retrieval., ACM (1992) 1–10
4. Godin, R., Gecsei, J., Pichet, C.: Design of a browsing interface for information retrieval. In Belkin, N., Rijsbergen van, C., eds.: Proceedings of the 12th International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM (1989) 32–39
5. Open directory project. http://www.dmoz.org (2006) Date retrieved: 22 March 2006.
6. Sieg, A.,Mobasher, B., Lytinen, S., Burke, R.: Using concept hierarchies to enhance user queries in web-based information retrieval. In: Proceedings of the IASTED International Conference on Artificial Intelligence and Applications. (2004)
7. Rode, H., Hiemstra, D.: Using Query Profiles for Clarification, Proceedings of the 28th European Conference on Information Retrieval ECIR, Springer (2006)
8. Lin, C.Y., Tseng, B.L., Smith, J.R.: Video Collaborative Annotation Forum: Establishing Ground-Truth Labels on Large Multimedia Datasets, Proceedings of the TRECVID video retrieval evaluation workshop (2003)
9. Hatcher, E., Gospodnetic, O.: Lucene in Action. Manning, Greenwich (2005)
10. FotoSearch Stock Photography and Stock Footage. http://www.fotosearch.com (2006)
11. Howitt, D., Cramer, D.: An Introduction to Statistics in Psychology. 2nd ed. Pearson Education Limited, Harlow (2000)
12. Dix, A., Finlay, J., Abowd, G.D., Beale, R.: Human-computer interaction. Prentice Hall, New York (2003)
13. Nemeth, Y., Shapira, B., Taeib-Maimon, M.: Evaluation of the Real and Perceived Value of Automatic and Interactive Query Expansion. SIGIR'04, Sheffield, UK. ACM (2004)