International Journal on
Advanced Science
Engineering
Information Technology

Vol. 2 (2012) No. 5
ISSN: 2088-5334

Incorporating Prioritized User Preferences in Search System

Mashal Alqudah[#1], Yuhanis Yusof[*2], Shahrul Azman Mohd Noah[#3,] Alaa Almabhouh[*4]

[#]*Universiti Kebangsaan Malaysia, Malaysia*
E-mail : [1]mashal_alqudah@yahoo.com, [3]samn@ftsm.ukm.my

[*]*Universiti Utara Malaysia, Sintok, Kedah, 06010, Malaysia*
E-mail : [2]yuhanis@uum.edu.my,[4]amabhouh@yahoo.com

***Abstract*— Web services are increasing daily and users are looking to find relevant services in a quick manner. Browsing irrelevant pages presented in a retrieval hitlist would consume time and effort. Hence, there is a need to reduce the search space which will then present users with a higher retrieval precision. The idea carried out by a priority based retrieval system is that, when one attributes links to another, it is basically indicating the importance of the other attribute. The higher the value, the more important the attribute is. In this work, the search requirements defined by users are accompanied by priority values. Relevant documents identified from various resources are sorted based on the priority values. Results obtained indicate that by using priority values, users of a retrieval system are better satisfied. The undertaken precision analysis shows that relevancy of the retrieved results is improved.**

***Keywords*— information retrieval, priority-based, document ranking**

I. INTRODUCTION

Most of the people search the web to retrieve information which considered a daily activity. So searching and communication are the most popular uses of the computer. Information retrieval is the science that covers the structure, analysis, organization, storage, searching and retrieving of information [1]. Moreover, Information retrieval (IR) is the art of searching for text, for information within text and for metadata about text, as well as that of searching the World Wide Web and relational databases. An information retrieval system can help users to retrieve documents relevant to the users' queries. As the internet is growing rapidly, internet users are presented with excessive information once a search is performed. In order to reduce the search space, data on user preferences should be included in retrieving the required information.

The idea carried out by a priority based approach is that, when one attributes links to another, it is basically casting a score for an attribute. The higher the number of score that is given for an attribute the higher the significance of the attribute and the meaning of 'goodness' for each attribute can be different for each user (or group of users).

This paper proposes a priority-based retrieval architecture which is later being implemented in a car rental application system. Using conventional method, a potential customer of a car rental company would personally go the premise and identify the desired vehicle to be hired. Such an approach requires extra time, effort and money. Nevertheless, as information and communication technology is expanding tremendously, existing car rental company have offered their services through websites. This includes the Mayflower [2], Shajasa Travel & Tours [3] and Hawk [4] car rental companies. Related information on car rental services (for example car model and distance charge) is presented in the company's website. Customers can visit this website and make queries about services offered by the company, such as the company's branches and types of cars and models that can be hired. Nevertheless, these online car rental business do not provide the opportunity for customer to prioritize their personal preferences in hiring the desired vehicle. For example, a customer could not indicate which one of his requirement is the most important factor to be considered in hiring a vehicle. Therefore, we see this as a loop hole in existing information retrieval and/or search systems. Thus, in this paper, we overcome such problem by including information on user preferences in identifying the required documents.

II. RELATED WORK

According to [5], gathering information doesn't mean that you have retrieved the information of your use and that satisfy users' needs. The main problem is to get the right information with proper quality, reliability and timeliness

and to get only information that has been requested: we will call this 'knowledge'. This can be achieved by clustering the data according to the properties of data/information. Examples of clustering applications are finding the uniform sub-populations or classification of sub-categories. Our work is based on this type of information retrieval where we intend to retrieve 'specific' documents based on a list of requirement.

An inconclusive study was carried out as described in [6] on query expansion which never proved effective except for the so-called "open domain question answering" task. They provided interesting evidence suggesting new guidelines for future research. Word sense disambiguation is in fact only one of the problems involved with sense based query expansion. The second is how to use sense information (and ontologies in general) to expand the query. They showed that expanding with synonyms or hyperonyms has a limited effect on web information retrieval performance, while other types of semantic information derivable from ontology are much more effective at improving search results. Therefore they developed an algorithm that may be tuned to produce high precision, possibly at the price of low recall.

According to [7], it is difficult for users to retrieve information that are special for them, if the search is based on traditional ranking method or the page that the users queried may appear at last of result list. Authors [7] proposed a new method named Categorization-based ranking algorithm which can help the user to get the target needs from the web pages. The text categorization part uses the model created in the classifier construction stage to sort new documents.

Web page sorting, known as web page categorization, might be defined as the task of determining whether a web page belongs to a category or categories. When a user puts query string into Web Server, the Web Server receive the query and take apart the query string into terms, then search the web pages which are identifying with query terms, authors select the intersection set of web pages, and get out the score of web pages, through getting the score must judge the category that the web pages belong to. If the page belongs to one category they get out the score stands on traditional algorithm, if the web page belongs to not only one category, we must check the categories it belongs to, and get out its synthetically score , then we will get result pages.

As described in [8], a difficulty through customary information retrieval (IR) systems. Users normally retrieve information without an explicitly defined domain of interest. Consequently, the system presents a lot of information that is of no relevance to the user. Hence, ontologies are used to enhance user-experience by getting the queries nearer to the user's needs.

According to [9], prior to using the User Preferences Search System (UPRE), users need to evaluate objects related to the search domain. For example, in order to provide travelling recommendations, the system provides the users a set of objects, representative, samples, and their properties (e.g. some hotels and their prices, distances from airport, pictures, etc.). The user will then evaluates/ranks these objects according to his/her belief. Using the ranking, UPRE detects the user's local preferences. The work presented in this paper is very much alike to UPRE. Nevertheless, we propose not to have a 'learning' session prior to using our search system [8] and [9]. Instead the retrieval architecture is domain specific where users just need to rank their priority.

A search broker acts as an intermediary between a user searching for information and a set of search servers [10]. It may perform automatic server selection, choosing servers which are likely to be most useful. It may also concurrently query the selected servers and present their results to the user in a single merged list. The effectiveness of a broker over a given set of servers depends on the effectiveness of its server selection and results merging methods. Its selection method must choose servers which return relevant documents. Its merging method must rank the combined results. This is similar to what we are trying to achieve in this study. We intend to combine results obtained from different resources and later sort the results in order to fulfill user needs.

Web services are increasing daily and users are looking to find relevant services in a quick manner. Browsing irrelevant pages presented in a retrieval hitlist would consume time and effort. Hence, there is a need to reduce the search space which will then present users with a higher retrieval precision. Therefore, we are proposing the priority-based retrieval architecture that incorporates prioritized preferences in a search system.

III. User Preferences-Based Retrieval Architecture

A. Architecture

In Figure 1, we present the retrieval architecture to be used in information retrieval. There are four main components namely Query Generator, Similarity Generator, Index Generator and Hitlist Generator. In addition, a database is used to store information on the analyzed documents. In this paper, the database contains information on car rental services provided by three companies.

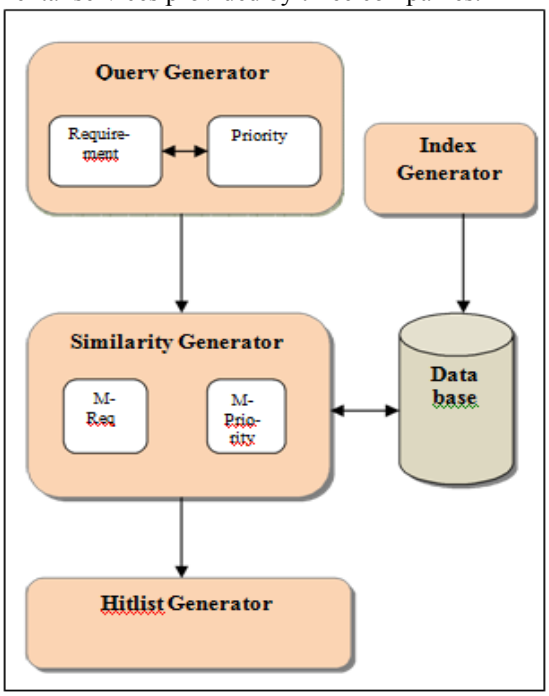

Fig.1 Priority-based Retrieval Architecture

The Query Generator contains two sub components; Requirement and Priority. The first refers to the search requirements identified by a user, for example model of the vehicle to be hired.er hand, each of the requirements will be accompanied by information of how importance it is to the user, and this is known as the score value. For example, if there are 5 search requirements included in a query, then there will be five score values being assigned to the requirements respectively. These values are given in ascending manner where value 1 indicates as the most important. In general, Query Generator produces the following:

{t1, t2, t3…tn} {s1, s2, s3,…sn} where tn indicates a search requirement and sn represents the score value for tn. An example of a search (having 5 requirements) and its priority values is {t1,t2,t3,t4,t5} {1,2,3,4,5}.

The Similarity Generator also consists of two subcomponents; M-Req and M-Priority. The M-Req is used to identify similarity between search requirements and data stored in the database. As for now, the string matching mechanism has been adopted as similarity measurement. In order to decrease the search space, M- Priority is used to prioritize information retrieved by M- Req. If any of the search requirements (for example t3) is not being fulfilled, then the priority value that user chooses manually for the particular requirement will be changed to zero. For example, the $\{t_1,t_2,t_3,t_4,t_5\}$ {1,2,3,4,5} will changed into $\{t_1,t_2,t_3,t_4,t_5\}$ {1,2,0,4,5}.

Information obtained using Similarity Generator is sorted by Hitlist Generator. In order to rank the relevant documents, the priority values generated by M-Score for a particular document are treated as a single real number and this is known as score value. For example, the {1,2,0,4,5} is treated as 12045. All of the documents identified to include at least one of the search requirements will have their own score values. The Hitlist Generator will then generate a retrieval hitlist that contains documents which the Index Generator descended and sorted based on their score values.

B. Title and Author Details

As mentioned earlier, we developed a car rental system as a proof of concept. The search requirements included in the car rental system can be seen in the system snapshot presented in Figure 2.

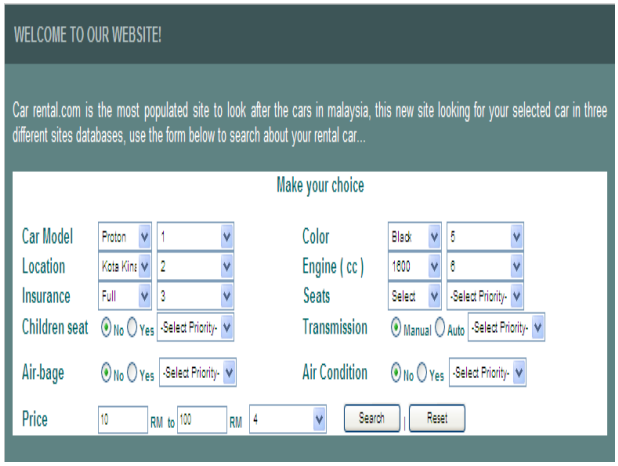

Fig.2 Snapshot on List of Requirements

A total of 30 users have participated in the experiment and information on participants background, usefulness and ease of use of the system have been collected. Participants are presented with two sets of retrieval hitlists; one (Version A) is generated without the use of weighting scheme while retrieval hitlist Version B is built based on the proposed architecture.

Version A is based on hyper-text matching which works in the following way. An example of requirement is where a user wants to hire a black Saga car that has full insurance. The system will first search the word “saga”, followed by the word “black” and finally the word “full” included under Car model, Colour and Insurance respectively. Portion of the result generation included in Version A is presented in Figure 3. The DB1, DB2 and DB3 are the car rental companies that are able to fulfill at least one of the search requirements. In Figure 3, it is noted that DB1 contains 3 Saga cars, 1 black car and 5 cars that have full insurance. With this, the total number of relevant documents in DB1 is 9. On the other hand, DB2 contains 12 relevant documents while in DB3 there are 8. Based on hyper-text matching, results from database having the greatest number of relevant documents will be presented on top of the retrieval hitlist. Hence, documents from DB2 are ranked first, followed by documents from DB1 and DB3 while the suitable DB for any user is DB1 since it contain all the information that he/she needs.

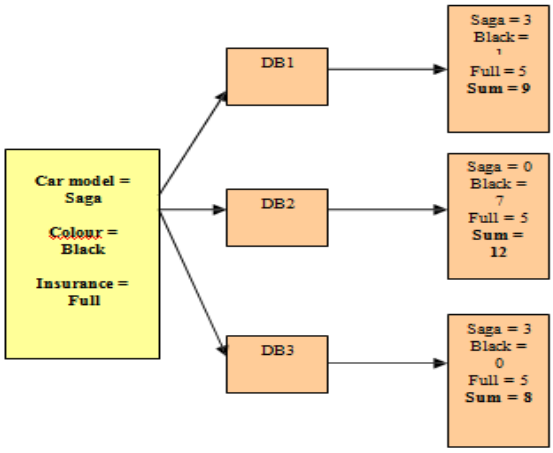

Fig.3 Results Generation (Version A)

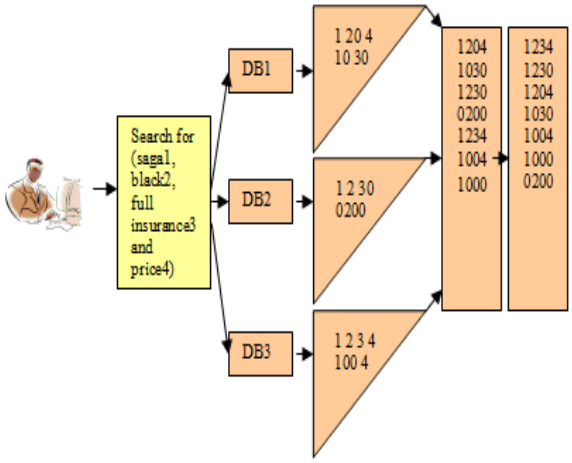

Fig.4 Result Generation using Priority-based Retrieval Architecture (Version B)

In the Version B search system, the requirements are given priority values and therefore retrieval results can be sorted based on the values. Portion of the result generation can be seen in Figure 4. Description on how the results are generated is included in previous section.

C. Finding

Respondents are required to identify if the result presented by both of the systems (Version A and Version B) are relevant to his/her requirement. Nevertheless, they are required to analyse only the top ten documents presented in the hitlists. With this, we obtained the precision value for the retrieval systems at cut-point 10. The average precision for Version A is 0.5 while Version B generated 0.73, and this is depicted in Table 1.

TABLE 1
STATISTICAL DESCRIPTION ON PERCEIVED RELEVANCY

	Version A	Version B
Mean	0.5066	0.73
Mode	0.4	0.7

Data depicted in Table 2 shows information on the perception on the usefulness and ease of use. The information is obtained upon analyzing information provided by respondents based on the 5-likert scale questions. The total mean for perceived usefulness and perceived ease of use is 4.27775. This value is larger than the mean score (point) which is three and this shows that respondents accept the proposed approach.

TABLE 2
STATISTICAL DESCRIPTION ON PERCEIVED USEFULNESS AND EASE OF USE

	N	Min	Max	Mean	Std. Deviation
Perceived Usefulness	30	2.67	5.00	4.2333	.55640
Perceived Ease of use	30	3.00	5.00	4.3222	.46719
Total	30			4.2777	

IV. CONCLUSIONS

Authors have addressed the problem of representing user preferences in retrieving required information by adopting the priority-based approach. User preferences are given the appropriate weight in order to indicate their importance. Such an approach would benefit most of the retrieval system that involves searching of several requirements. To proof the effectiveness of the approach, we have performed a comparison on the usefulness and user acceptance between existing method (solely based on string matching) and our approach. Based on the result presented in the earlier subsections, we concluded that by including user preferences in a retrieval system, users are better satisfied. This is shown by obtaining a higher precision value when compared between systems. Hence, by applying a priority-based approach, relevancy of the retrieved results is improved and at the same time fulfilled user preferences.

Nevertheless, there is still work to be done in improving the work presented in this paper. Several issues need to be addressed and this includes solving ambiguities in matching user preferences and the priority values.

REFERENCES

[1] C. D. Manning, P. Raghavan, H. Schütze, and C. Ebooks, Introduction to information retrieval vol. 1: Cambridge University Press Cambridge, UK, 2008.

[2] "Mayflower Car Rental Phd. co ", 2011. [Online]; available: http://www.mayflowercarrental.com/

[3] "Shajasa Travel and Tour," 2011. [Online]; available: http://www.malaysiacarrental.com/

[4] "Hawk Car Rental Phd.co," 2011. [Online]; available: http://www.kuala-lumpur.ws/hawk/

[5] G. Cumming, "Targeted Information Retrieval," in 7th International Conference on Computers in Education, Chiba, Japan, 1999, p. 355.

[6] R. Navigli and P. Velardi, "An analysis of ontology-based query expansion strategies," in Workshop on Adaptive Text Extraction and Mining, 2003, pp. 42–49.

[7] J. Huang, G. Wang, and Z. Wang, "Cross-subject page ranking based on text categorization," in International Conference on Information and Automation, ICIA 2008, pp. 363-368.

[8] S. Tomassen, "Research on ontology-driven information retrieval," in WorkshopsOn the Move to Meaningful Internet Systems, 2006, pp. 1460-1468.

[9] P. Gursky, T. Horvath, R. Novotny, V. Vanekova, and P. Vojtas, "UPRE: User preference based search system," in International Conference on Web Intelligence 2006, pp. 841-844.

[10] N. E. Craswell, "Methods for distributed information retrieval," Unpublished doctoral dissertation, Australian National University, 2000.