A Novel Approach for Filtering Appropriate Document from Most Relevant Query Terms

Turki Alghamdi[1], Mohammad Husain[2] and Ahmad Alkhodre[3]

Department of Computer Science & Information System
Islamic University, Madinah, Saudi Arabia

Abstract : In the recent years the rapid growth has been observed in the field of information retrieval, a user generally needs to interact with the retrieval system many times to get satisfactory results for one information need, which offers good opening for the retrieval system to dynamically take part in this iterative retrieval method. Most traditional retrieval systems just passively respond to user queries and put the responsibility to refine the search exclusively on the user. But there has been evidence showing that a retrieval method can provide better results in this process. To find relevant document from heterogeneous databases in efficient and effective manner is the major issue now a days. There are number of techniques available to solve this issue.
In this paper, we have proposed an approach to improve the performance in finding the relevant document from heterogeneous databases. Here we considered two major things, one retrieval based on text mining and the query expansion based on the users feedback. The aim is to find the relevant document from the most relevant query terms, refining the query terms by introducing the users feedback in it. Our proposed architecture, algorithms and experimental results provides evidence towards significantly improvements in the precision and recall.

Keywords : Personalization; Meta-search; Heterogeneous; Feedback; Indexing.

1. Introduction

With the rapid growth of the computer technology in the recent years, digital information has been explosively increased. This tendency is especially amazing on the Web. Since the availability of the information increases, the requirement for finding more appropriate information on the web is growing day by day[1].
Peoples are relying more and more on the Web for their diverse needs of information. Now in this circumstances the role of search engine technology plays an important role to guide the users and employ such an enormously important resource. Despite the fact that keywords are not always good descriptors of contents, most existing search engines still rely solely on keyword-matching to determine the answers. Users usually describe their information needs by a few keywords in their queries, which are likely to be different from those index terms of the documents on the Web[1]. A query expansion method based on global analysis usually builds a thesaurus to assist users reformulating their queries. A thesaurus can be automatically established by analyzing relationships among documents and statistics of term co-occurrences in the documents. From the thesaurus constructed in this way, one will be able to obtain synonyms or related terms given a user query. Thus, these related terms can be used for supplementing users' original queries[1].

Information extraction is a subfield of natural language processing that seeks to obtain structured information from unstructured text. Information extraction can be used to automate the tedious and error prone process of collecting facts from the Web. Open Information extraction is a relation-independent form of Information extraction that scales well to large corpuses. Unfortunately, extraction engines, like search engines, intermix relevant information with irrelevant information. This problem is exacerbated in Information extraction systems because they use heuristic methods to extract phrases that are meant to denote entities and relationships.

A heterogeneous database system is an automated system for the integration of heterogeneous, dissimilar database management systems to present a user with a single, unified query interface. Heterogeneous database systems are computational models and software implementations that provide heterogeneous database integration. To efficiently retrieve information from heterogeneous and distributed data sources has become one of the top priorities for everyone. Information from these sources needs to be integrated into one single system such

(DOI: dx.doi.org/14.9831/1444-8939.2015/3-4/MAGNT.36)

that the user can retrieve the desired information through the integrated system by a single query.

There are several requirements that a heterogeneous management system should satisfy. Firstly, it should be able to accommodate systems which may or may not provide certain types of functionality required for them to operate properly in a concurrent heterogeneous environment. For example, a heterogeneous transaction management system may require some local concurrency mechanism with which to communicate in each autonomous system. Any autonomous system missing this functionality would have to be extended with a concurrency control mechanism of its own to be able to communicate with the heterogeneous manager. Such characteristic of heterogeneous environments requires integration processes that will help enlarge each autonomous system with the functionality that it may need.

Secondly, a heterogeneous management system must be able to integrate systems that provide incompatible implementations for the functionality that they do provide. A lack of information about variations in the internal implementation of closed architecture systems can delay this process and can make it difficult to design a heterogeneous manager which operates optimally. This characteristic of heterogeneous environments requires open, extensible architectures in which integrating processes can truly manage the heterogeneity of dissimilar implementations.

And finally, a heterogeneous management system must be adequate and flexible in order to accommodate requirements placed on it by end-user applications, and the autonomous configuration systems that it integrates.

A challenge here is that while people can identify what is interesting to them, it is less clear how computers can do this algorithmically. We could implement several theories of interestingness from psychology such as complexity, novelty, uncertainty, and conflict, but it is unclear which, if any, is best. The most accurate method would be if we could have people go through and specify which of the extracted assertions are of interest.[3] exploit Relevance feedback is normally used for query expansion during short-term modeling of a user's immediate information need and for user profiling during long-term modeling of a user's persistent interests and preferences. Conventionally relevance feedback methods require that users explicitly give feedback by, for example, specifying keywords, selecting and marking documents, or answering questions about their interests[7]. Such relevance feedback methods force users to make busy in additional activities instead their normal searching process. Since the cost to the user is high and the benefits are not always clear, it can be difficult to find the necessary data and the effectiveness of explicit techniques can be restricted[7].

When responding to queries, the goal of an information retrieval system – ranging from web search, to desktop search, to call center support – is to return the results that maximize user utility. So, how can a retrieval system learn to provide results that maximize utility?

The conventional approach is to optimize a proxy measure that is hoped to correlate with utility. A wide range of measures has been proposed to this effect, but all have similar problems. Most obviously, they require expensive manual relevance judgments that ignore the identity of the user and the user's context. This makes it unclear whether maximization of a proxy-measure truly optimizes the search experience for the user.[8]

As a consequence, in many cases, the documents returned by search engines are not relevant to the user information need. This raises a fundamental problem of term mismatch in information retrieval, which is also one of the key factors that affect the precision of the search engines.

Very short queries submitted to search engines on the Web amplify this problem: Many important words or terms may be missing from the queries. To solve this problem, researchers have investigated the utilization of query expansion techniques to help users formulate better queries. Query expansion involves supplementing the original query with additional words and phrases. There are two key aspects in any query expansion technique: the source from which expansion terms are selected and the method to weight and integrate expansion terms. [9]

A system can help with query refinement, either fully automatically or with the user in the loop. The methods for tackling this problem split into two major classes: global methods and local methods. Global methods are techniques for expanding or reformulating query terms independent of the query and results returned from it, so that changes in the query phrasing will cause the new query to match other semantically similar terms. Global methods include:

(DOI: dx.doi.org/14.9831/1444-8939.2015/3-4/MAGNT.36)

- Query expansion/reformulation with a glossary etc.
- Query expansion via automatic glossary generation
- Techniques like spelling correction

Local methods adjust a query relative to the documents that initially appear to match the query. The basic methods are -

- Relevance feedback
- Pseudo relevance feedback, also known as Blind relevance feedback
- (Global) indirect relevance feedback

The idea of relevance feedback is to involve the user in the retrieval process so as to improve the final result set. In particular, the user gives feedback on the relevance of documents in an initial set of results.

The success of relevance feedback depends on certain assumptions. Firstly, the user has to have sufficient knowledge to be able to make an initial query which is at least somewhere close to the documents they needed. This is required anyhow for successful information retrieval in the basic case, but it is important to see the kinds of problems that relevance feedback cannot solve alone. Cases where relevance feedback alone is not sufficient include:

• Misspellings. If the user spells a term in a different way to the way it is spelled in any document in the collection, then relevance feedback is unlikely to be effective. This can be addressed by the spelling correction techniques.

• Cross-language information retrieval. Documents in another language are not nearby in a vector space based on term distribution. Rather, documents in the same language cluster more closely together.

• Mismatch of searcher's vocabulary versus collection vocabulary. If the user searches for laptop but all the documents use the term notebook computer, then the query will fail, and relevance feedback is again most likely ineffective.

Secondly, the relevance feedback approach requires relevant documents to be similar to each other. That is, they should cluster. Ideally, the term distribution in all relevant documents will be similar to that in the documents marked by the users, while the term distribution in all non relevant documents will be different from those in relevant documents. Things will work well if all relevant documents are tightly clustered around a single prototype, or, at least, if there are different prototypes, if the relevant documents have significant vocabulary overlap, while similarities between relevant and non relevant documents are small. Implicitly, the Rocchio relevance feedback model treats relevant documents as a single cluster, which it models via the centroid of the cluster. This approach does not work as well if the relevant documents are a multimodal class, that is, they consist of several clusters of documents within the vector space.[10]

2. Literature Review

Hang Cui et. al.[1] have suggested a new query expansion method based on user logs which record user interactions with the search systems. User logs are exploited so as to extract implicit relevance judgments they encode. In this approach, we assume that the documents that the user choose to read are "relevant documents." The log-based query expansion overcomes several difficulties of local analysis because we can extract a large number of user judgments from user logs, while eliminating the step of collecting feedbacks from users for ad hoc queries. Probabilistic correlations between terms in the user queries and the documents can then be established through user logs. With these term-term correlations, relevant expansion terms can be selected from the documents for a query. They shows that mining user logs is extremely useful for improving retrieval effectiveness, especially for very short queries on the Web.

Thomas Lin, Oren Etzioni, James Fogarty [3] have investigated that Large repositories of knowledge can enable more powerful AI systems. Information Extraction (IE) is one approach to building knowledge repositories by extracting knowledge from text. Open IE systems like TextRunner [Banko et al., 2007] are able to extract hundreds of millions of assertions from Web text. However, because of imperfections in extraction technology and the noisy nature of Web text, IE systems return a mix of both useful, informative facts (e.g., "the FDA banned ephedra") and less informative statements (e.g., "the FDA banned products").

They investigates using user-contributed knowledge from Wikipedia and from TextRunner website visitors to train classifiers that automatically filter extracted assertions. In a study of human ratings of the interestingness of TextRunner assertions, they show that their approach substantially enhances the quality of results and the relevance feedback filter raises the fraction of interesting results in the top thirty from 41.6% to 64.1%.

(DOI: dx.doi.org/14.9831/1444-8939.2015/3-4/MAGNT.36)

Hang Cui, Ji-Rong Wen, Jian-Yun Nie, and Wei-Ying Ma [9] have suggested that the proliferation of the World Wide Web prompts the wide application of search engines. However, short queries and the incompatibility between the terms in user queries and documents strongly affect the performance of the existing search engines. Many automatic query expansion techniques have been proposed, which can solve the short query and the term mismatching problem to some extent.
However, they do not take advantage of the user logs available in various Web sites, and use them as a means for query expansion.

Here they presented a novel method for automatic query expansion based on user logs. This method aims first to establish correlations between query terms and document terms by exploiting the user logs. These relationships are then used for query expansion. they have shown that this is an effective way to narrow the gap between the query space and the document space. For new queries, high-quality expansion terms can be selected from the document space on the basis of the extracted correlations.

They have tested this method on a data set that is similar to the real Web environment. A series of experiments conducted on both long queries and short queries showed that the log based query expansion method can achieve substantial improvements in performance. It also outperforms local context analysis, which is one of the most effective query expansion methods in the past. Their experiments also shows that query expansion is more effective for short queries than for long queries.

Djoerd Hiemstra, Stephen Robertson[12] have suggested that Relevance feedback in full text information retrieval inputs the user's judgments on previously retrieved documents to construct a personalised query. These algorithms utilize the distribution of terms over relevant and irrelevant documents to re-estimate the query term weights, resulting in an improved user query. Relevance feedback is especially helpful in applications where users have a long-lasting information need, with plenty of opportunity to give feedback to the system, for instance in adaptive filtering systems.
They have introduces new relevance feedback algorithms for both probabilistic approaches to information retrieval mentioned above: the language models and the binary independence model. It introduces a new relevance feedback algorithm for language model-based information retrieval systems by utilizing the expectation maximization (EM-) algorithm. The new relevance feedback algorithm for the binary independence model is a generalization of the traditional Robertson/Sparck-Jones relevance weight.
It calculates the expected weight over the ranked list of documents retrieved by a single term query. They argue that the traditional Roberston/Sparck-Jones relevance weight is not appropriate for best match versions of the model, like for instance the BM25 algorithm.

They have also commented that why simple approaches to relevance feedback are inappropriate for best match retrieval algorithms. However, they were unable to show that the well-motivated algorithms perform significantly better than the simple algorithms. Apparently, the simple algorithms approximate the well motivated algorithms well enough to be realistic in real retrieval settings.

3. Proposed Methodology and Architecture

3.1 Retrieval Model

The word indexing based Information Retrieval may be better ranking than the character indexing based Information Retrieval, so to accomplish better performance rather than the character indexing based Information Retrieval. Some relevant documents does not contains segmented words in the query terms, hence it could not be retrieved, due to this the progress in the retrieval is inadequate.

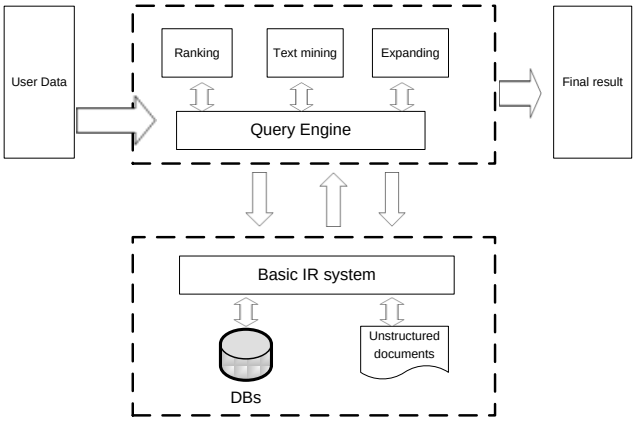

Figure 1 : Proposed Information Retrieval System

In our proposed system user sends the query to the query engine then query engine interact with the basic IR system and gathers the data from databases as well as from unstructured documents, after processing it ranks the document on the basis of the following ranking model to calculate the

(DOI: dx.doi.org/14.9831/1444-8939.2015/3-4/MAGNT.36)

ranking value of a retrieved document to determine the top R relevant documents –

$$doc_r = qt^5 \sum_{n=1}^{q} df_n^c \; X \; invdf_n^c$$

Here, q is the number of query terms, qt is the number of various query terms that appear in the document as character sequence. df_n^c is the frequency of the n^{th} term in the document and $invdf_n^c$ is the inverse document frequency of the n^{th} term in the collection. The above equation can make sure of two things: (a) the more distinct query terms are matched in a document, the higher the rank of the document. For example, a document that contains six distinct query terms will almost always have higher rank than a document that contains five distinct query terms, regardless of the query terms frequency in the document. (b) when documents contain a similar number of distinct terms, the score of a document will be determined by the sum of query terms *df-invdf* value, as in traditional information retrieval.

3.2 Architecture

In our proposed architecture when user sends the query to the system the query interact with different sources, prepared documents accordingly and ready to use for first phase, then extract the relevant document by applying similarity factor and ranked the documents after this phase user received highly ranked documents which fulfil the requirement of the users need. If it does not found suitable according to the users requirement then in this system we have proposed user feedback process by which user can provide his/her feedback and again query interact with the IR system, based on the query expansion the process goes for second phase of retrieval of documents and finally the user receives the extracted results as per their requirement.

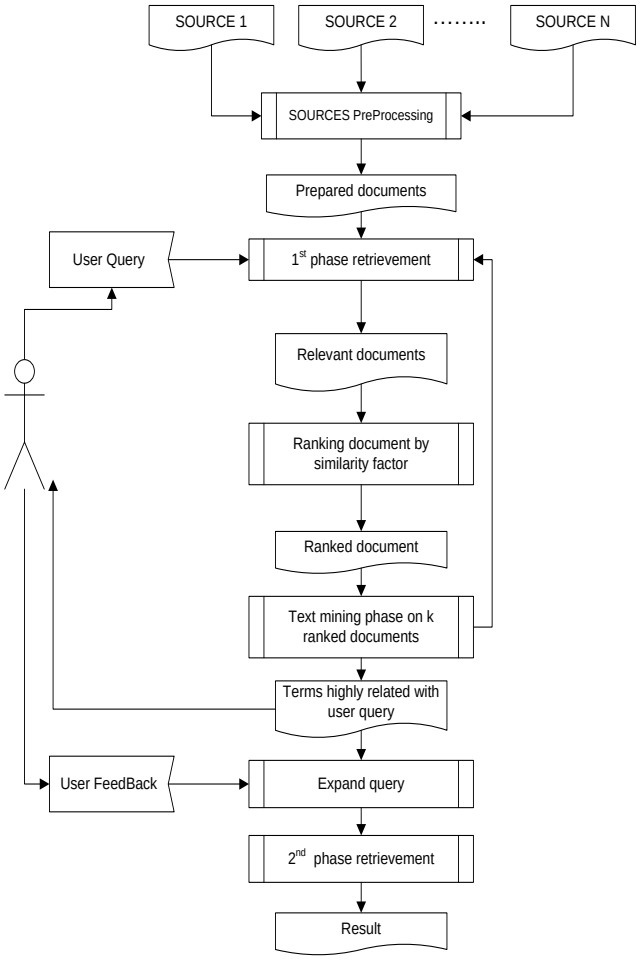

Figure 2 : Architecture for retrieving the database and document from heterogeneous databases

3.3 Proposed Algorithm

We want to select those documents from number of sources, which satisfy our query 'q'. The Basic idea of this algorithm is that we examine different type of data sources in the order $Source_1$, $Source_2$, $Source_3$, $Source_4$, $Source_5$,……., $Source_N$, until we get the Sources which contain the query 'q'. This algorithm works as follows -

Algorithm-HAT

1. Search preparation
   a. Index the sources document selected
   b. Merging the heterogonous Databases

2. Input the user query "q_1"

3. Examine each source with its documents accumulated in it. If any document of source

(DOI: dx.doi.org/14.9831/1444-8939.2015/3-4/MAGNT.36)

contains the query at least one time then we select that Data Source.

4. If not all the documents of source contain the query then that database will not be selected.

5. We select only those documents from each source in which the query ‘q1’ occurs at least one time.

6. (a) Calculate the similarity factor based on

$$siml(a,b)=1-(1/m)\sum_{i=1}^{m} Sim(a_i,b_i)*1/k \sum_{j=1}^{k} Sim(a_j,b_j)$$

(b) Calculate the ranking value based on

$$doc_r=qt^5 \sum_{n=1}^{q} df_n^c \; X \; invdf_n^c$$

7. Sort the resulted document using similarity factor and ranking value.

8. Select ‘n’ ranked document who have highest similarity factor.

9. Extract the features based on index term selection

10. Display features to user extracted from each document

11. User modify/rewrite his/her query according to features selected

12. Repeat the procedure from step3 to step8

13. Display final result.

3.4 Sequence diagram

The interaction between the components of our proposed system based on the above algorithm is represented in figure 3. The interaction are usually carried out sequentially within the components. Moreover, The sequence diagram in figure 3 shows the behavior of the proposed system from the process when the user places a query of information till the final result is returned to the user.

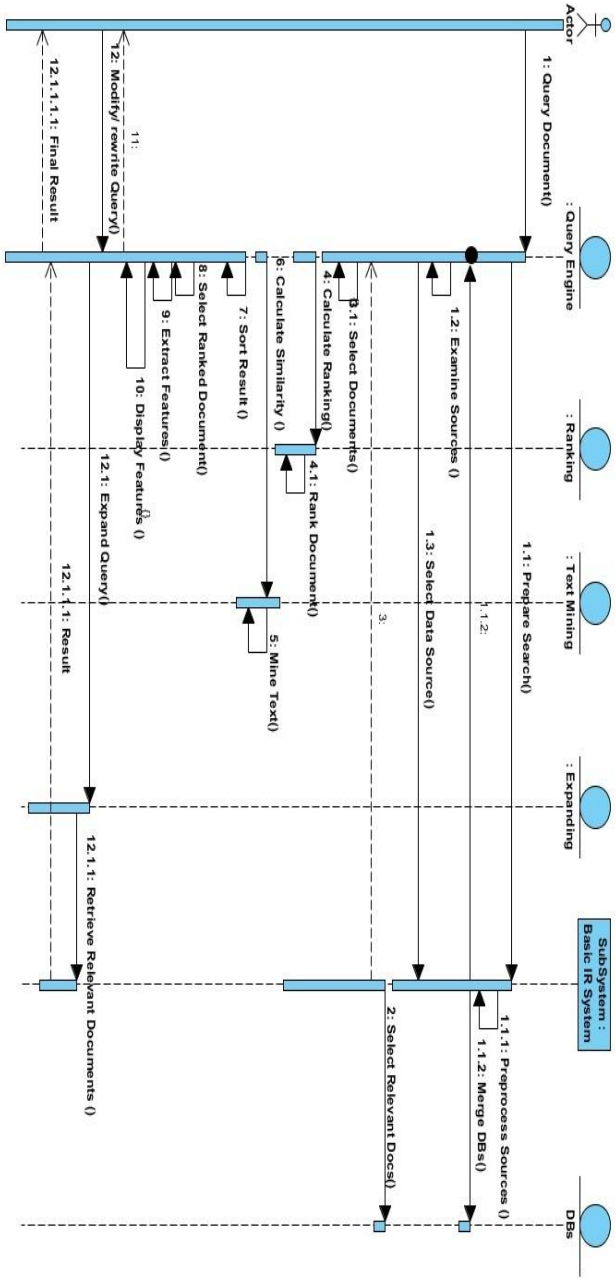

Figure 3 : Behavior of Information Retrieval System

4. Experimental Result

We have design a software which contains about 10,000 documents, we also design two indexing tables, one contain indexes of all characters exist in the documents and another contains all the words. In this we pose thirty queries and each query contains simple description which consist of one or more terms. We use the average precision and recall of the top 10,15,20,30 and 100 retrieved documents to evaluate the performance of the proposed algorithm-HAT with other techniques like - character based model

(DOI: dx.doi.org/14.9831/1444-8939.2015/3-4/MAGNT.36)

without query expansion denoted by C and the word-character based model without query expansion denoted as W-C, it is found that the performance is improved slightly from the character based model to the word-character based model, the precision is improved by 0.3% on average from 29.8% to 30.1% and the recall is improved by 0.1% on average from 33.4% to 33.5% but while we applied our proposed HAT-algorithm then we found that first phase result is equivalent to W-C model as shown in table 1.

Top N	C		W-C		HAT	
	P (%)	R (%)	P (%)	R (%)	P (%)	R (%)
10	39.7	19.6	39.9	20.1	39.9	20.1
15	35.2	26.1	35.5	26.5	35.5	26.5
20	32.7	30.9	32.9	31.3	32.9	31.3
30	27.5	36.7	27.5	35.7	27.5	35.7
100	14.1	53.8	14.6	53.9	14.6	53.9
Avg.	29.8	33.4	30.1	33.5	30.1	33.5

Table 1 : Result obtained by HAT Algorithm in 1st Phase

Now applying Algorithm-HAT for the second phase i.e. after the step 4 to step 13., compared this with the popularly known standard query expansion method Rocchio QE(R) and text mining based query expansion QE(TM). It is observed that the text mining based query expansion QE(TM) have achieved a little more improvement in comparison to Rocchio standard query expansion QE(R), the precision is improved by 1.4% on average from 32.5% to 36.4% and the recall is improved by 6.0% on average from 35.7% to 41.7%.

Top N	QE (R)		QE(TM)		HAT	
	P (%)	R (%)	P (%)	R (%)	P (%)	R (%)
10	41.6	23.2	49.2	26.8	49.5	27.3
15	39.7	28.9	44.1	34.0	44.6	34.6
20	35.9	32.1	39.2	38.7	39.8	39.2
30	30.2	33.8	33.0	44.5	33.1	45.1
100	15.1	60.4	16.3	64.7	16.8	65.2
Avg.	32.5	35.7	36.4	41.7	36.8	42.3

Table 2 : Result obtained by HAT Algorithm in 2nd Phase

Where in our approach expanded query the precision is improved by 0.4% on average from 36.4% to 36.8% and the recall is improved by 0.6% from 41.7% to 42.3%.

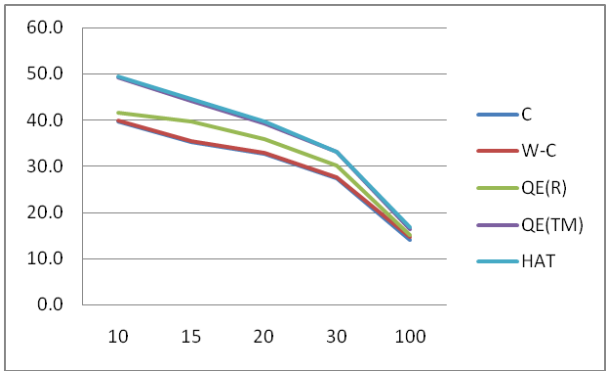

Figure 4 : Comparision of HAT Algorithm with other methods based on Precision

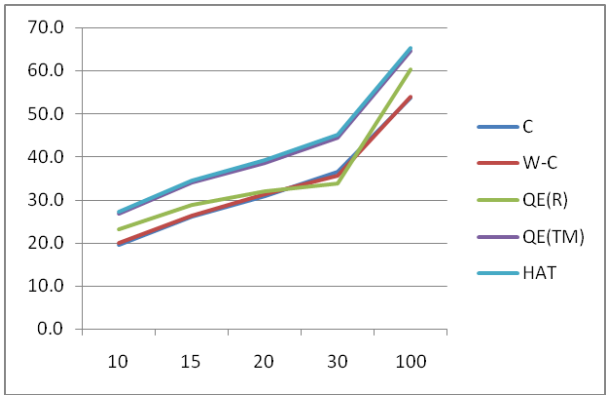

Figure 5 : Comparision of HAT Algorithm with other methods based on Recall

5. Conclusion

In ad hoc information retrieval, a user generally needs to interact with the retrieval system many times to get satisfactory results for one information need, which provides opportunities for the retrieval system to actively participate in this iterative retrieval process. Most traditional retrieval systems just passively respond to user queries and put the responsibility to refine the search exclusively on the user. But there has been evidence showing that a retrieval system can play an important role in this process. To find relevant document from heterogeneous databases in efficient and effective manner is the major issue now a days. There are number of techniques available to solve this issue.

Relevance feedback is normally used for query expansion during short-term modeling of a user's immediate information need and for user profiling during long-term modeling of a user's persistent

(DOI: dx.doi.org/14.9831/1444-8939.2015/3-4/MAGNT.36)

interests and preferences. Conventionally relevance feedback methods require that users explicitly give feedback by, for example, specifying keywords, selecting and marking documents, or answering questions about their interests. Such relevance feedback methods force users to make busy in additional activities instead their normal searching process. Since the cost to the user is high and the benefits are not always clear, it can be difficult to find the necessary data and the effectiveness of explicit techniques can be restricted.

In this paper, we proposed an approach to improve the performance in finding the relevant document from heterogeneous databases. The approach includes two aspects ie. retrieval based on text mining and the query expansion based on the users feedback. The experimental results shows that our proposed algorithm brings significant improvements in the result. In our approach expanded query based on user feedback the precision is improved by 0.4% on average from 36.4% to 36.8% and the recall is improved by 0.6% from 41.7% to 42.3%. Our proposed method can yield substantial improvements over existing techniques.

References

1. Hang Cui, Ji-Rong Wen, Jian-Yun Nie, and Wei-Ying Ma, "Query Expansion by Mining User Logs", IEEE Transactions on knowledge and Data Engineering, Vol. 15, No. 4, July/August 2003.
2. Deepika Sharma, KirtiChoudhary, Sandeep Kumar Poonia, "Design And Implementation Of Context Based Information Retrieval System", International Journal of Engineering, Management & Sciences (IJEMS) ISSN-2348 –3733, Volume-1, Issue-6, June 2014.
3. Thomas Lin, Oren Etzioni, James Fogarty, "Filtering Information Extraction via User-Contributed Knowledge", In Proc. of WikiAI, 2009.
4. Xuehua Shen, ChengXiang Zhai, "Active Feedback in Ad Hoc Information Retrieval", SIGIR'05, August 15–19, 2005, Salvador, Brazil.
5. T. Joachims., "Optimizing search engines using clickthrough data", In Proceedings of SIGKDD, 2002.
6. White, R. W., Ruthven, I., and Jose, J. M., "Finding relevant documents using top ranking sentences: An evaluation of two alternative schemes", In Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '02), Finland 2002, 57-64.
7. D. Kelly and J. Teevan., "Implicit feedback for inferring user preference: A bibliography", SIGIR Forum, 37(2):18–28, 2003.
8. Yisong Yue, Thorsten Joachims, "Interactively Optimizing Information Retrieval Systems as a Dueling Bandits Problem", In Proceedings of the 26th International Conference on Machine Learning, Montreal, Canada, 2009.
9. Hang Cui, Ji-Rong Wen, Jian-Yun Nie, and Wei-Ying Ma, "Query Expansion by Mining User Logs", In IEEE Transactions on Knowledge and data Engineering, Vol. 15, No. 4, July/August 2003.
10. "Relevance feedback and query expansion", April 1, 2009 Cambridge University Press.
11. Robert M. Losee and Lewis Church Jr., "Information Retrieval with Distributed Databases: Analytic Models of Performance", IEEE Transactions on Parallel and Distributed Systems, Vol. 14, No. 12, December 2003.
12. Djoerd Hiemstra, Stephen Robertson, "Relevance Feedback for Best Match Term Weighting Algorithms in Information Retrieval", In Proceedings of the Joint DELOS-NSF Workshop on Personalisation and Recommender Systems in Digital Libraries, pages 37–42, Dublin, Ireland, June 2001.
13. Andreas Hotho, Andreas N¨urnberger, Gerhard Paaß, "A Brief Survey of Text Mining", May 13, 2005
14. A. Wierse U. Fayyad, G. Grinstein, "Information Visualization in Data Mining and Knowledge Discovery", Morgan Kaufmann, 2001.
15. S. Bloehdorn and A. Hotho, "Text classification by boosting weak learners based on terms and concepts", In Proc. IEEE Int. Conf. on Data Mining (ICDM 04), pages 331–334. IEEE Computer Society Press, NOV 2004.
16. I.S. Dhillon, S. Mallela, and D.S. Modha., "Information-theoretic co-clustering", In Proc. of the ninth ACM SIGKDD int. conf. on Knowledge Discovery and Data Mining, pages 89–98. ACM Press, 2003.
17. R. Gaizauskas., "An information extraction perspective on text mining: Tasks, technologies and prototype applications. http://www.itri.bton.ac.uk/projects/euromap/ TextMiningEvent/Rob_Gaizauskas.pdf, 2003.

(DOI: dx.doi.org/14.9831/1444-8939.2015/3-4/MAGNT.36)

18. S. Havre, E. Hetzler, K. Perrine, E. Jurrus, and N. Miller., "Interactive visualization of multiple query result", In Proc. of IEEE Symposium on Information Visualization 2001, pages 105 –112. IEEE, 2001.
19. J. M. G. Hidalgo., "Tutorial on text mining and internet content filtering", Tutorial Notes Online: http://ecmlpkdd.cs.helsinki. fi/pdf/hidalgo.pdf, 2002.
20. U. Nahm and R. Mooney., "Text mining with information extraction", In Proceedings of the AAAI 2002 Spring Symposium on Mining Answers from Texts and Knowledge Bases, 2002.

(DOI: dx.doi.org/14.9831/1444-8939.2015/3-4/MAGNT.36)