Dynamic Grouping of Semantically Similar User Search Histories

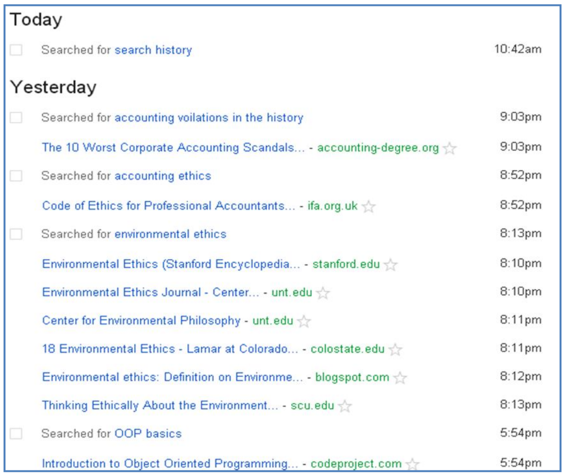
A.S.Saleem Basha[#1], V.Trilik Kumar[*2]
[#1]M.Tech, Department of CSE, KVSRIT, kurnool, Andhra Pradesh, India
[#2]Associate Professor, KVSRIT,kurnool, Andhra Pradesh, India

ABSTRACT--Users over Internet make queries continuously for various kinds of information. Such information might be about various tasks and that is done through existing search engines. When queries are made by users continuously, over a period of time, the queries are plenty. The existing search engines organize such queries only in chronological order. However, when the quires are grouped together based on the relevancy that might be very useful to users as they can reuse queries with ease. Hwang et al. studied this problem recently and proposed mechanisms that help in grouping or organizing user search histories in useful fashion. This organization of user search histories can have various real time utilities such as result ranking, query alternations, query suggestions, sessionization and collaborative search. In this paper we implement algorithms that are used to group user search histories. We built a web based prototype that demonstrates the proof of concept. The empirical results are encouraging.

Index Terms--Search engine, search history, click graph, query grouping

I. INTRODUCTION

World Wide Web is rich in information as it accumulates vast data every day from various sources. As there is content of all walks of life, people make searches in order to get required information. Thus the search engines are playing a great role in obtaining required information. Search engines like AltaVista [1] and Yahoo [2] witness 20% of navigational queries while other queries are transactional. Task oriented searches are made by users for their needs such as travelling, finances, purchases and so on. It is very common that users give input to search engines in the form of key words. Search engines take the queries as input and come up with results. The users can make queries that can be reused when they are organized well. At present the search engines organize the history of searches in hierarchical fashion and in chronological order. Figure 1 shows how the search engines organize queries.



Today

☐ Searched for search history 10:42am

Yesterday

☐ Searched for accounting voilations in the history 9:03pm

The 10 Worst Corporate Accounting Scandals... - accounting-degree.org ☆ 9:03pm

☐ Searched for accounting ethics 8:52pm

Code of Ethics for Professional Accountants... - ifa.org.uk ☆ 8:52pm

☐ Searched for environmental ethics 8:13pm

Environmental Ethics (Stanford Encyclopedia... - stanford.edu ☆ 8:10pm

Environmental Ethics Journal - Center... - unt.edu ☆ 8:10pm

Center for Environmental Philosophy - unt.edu ☆ 8:11pm

18 Environmental Ethics - Lamar at Colorado... - colostate.edu ☆ 8:11pm

Environmental ethics: Definition on Environme... - blogspot.com ☆ 8:12pm

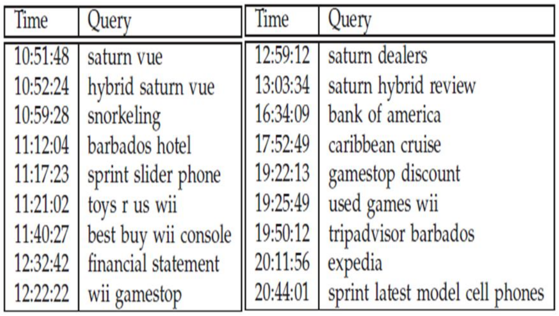
Thinking Ethically About the Environment... - scu.edu ☆ 8:13pm

☐ Searched for OOP basics 5:54pm

Introduction to Object Oriented Programming... - codeproject.com ☆ 5:54pm

Fig.1 – Search history of a user organized by Google

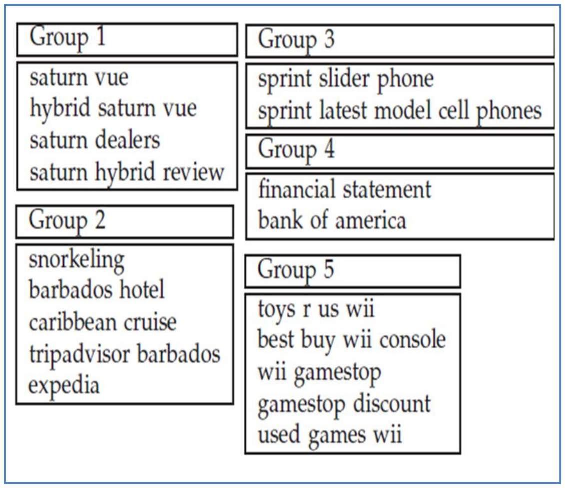
As shown in figure 1, it is evident that the Google is organizing search history of users in chronological order. Google kind of search engines is capable of organizing various search histories made by end users. However, the chronological order is not much useful to end users. They wanted to view the related queries together so that they can reuse queries. Moreover the search engines also have their own advantages when they are able to organize user search histories in different order based on relevancy. Figure 2 shows some queries that are available.



Time	Query	Time	Query
10:51:48	saturn vue	12:59:12	saturn dealers
10:52:24	hybrid saturn vue	13:03:34	saturn hybrid review
10:59:28	snorkeling	16:34:09	bank of america
11:12:04	barbados hotel	17:52:49	caribbean cruise
11:17:23	sprint slider phone	19:22:13	gamestop discount
11:21:02	toys r us wii	19:25:49	used games wii
11:40:27	best buy wii console	19:50:12	tripadvisor barbados
12:32:42	financial statement	20:11:56	expedia
12:22:22	wii gamestop	20:44:01	sprint latest model cell phones

Fig. 2 – Search history of a user (excerpt from [3])

As seen in figure 2, it is evident that the queries are in chronological order. It can be organized into meaningful groups as shown in figure 3.



Group 1

saturn vue
hybrid saturn vue
saturn dealers
saturn hybrid review

Group 2

snorkeling
barbados hotel
caribbean cruise
tripadvisor barbados
expedia

Group 3

sprint slider phone
sprint latest model cell phones

Group 4

financial statement
bank of america

Group 5

toys r us wii
best buy wii console
wii gamestop
gamestop discount
used games wii

Fig. 3 –Query Groups(excerpt from [3])

As shown in figure 3, it is evident that the related quires are grouped together. This will help users to reuse such queries easily. Besides, the search engines can make use of these lists for various operations such as sessionization, query processing, query modifications, collaborative Search and so on. This kind of approach is also followed in [4], [5] for session identification and in [6], [7] for query clustering. However, in this paper our work extends that in two ways. We use information from click graph and also query reformulation graph for capturing similarity in better way. We built a prototype web application to demonstrate the proof of concept.

The remainder of this paper is organized into some sections. Section II presents review of literature. Section III provides the proposed approach for organizing user search histories. Section IV describes prototype implementation details. Section V presents experimental results while section VI concludes the paper.

II. PRIOR WORKS

Chronological order is the only way used earlier to organize user's search histories. There might be queries that are related and may belong to single search task. Search tasks might be made up of many queries. IN [4] and [5] this has been explored. Binary classifier was used for exploiting text, time and query logs in order to organize queries. Similar search was made in [5]. The researcher in [4] has not given any provision for breaking queries into groups. Manual labeling is not required by our approach. Query fusion approach is required in some cases where random walk approach is used. This will also leads to personalization, query suggestions and sessionization. The "timeout threshold" approach was employed by many researchers as explored in [8], [9], [10], [11], [12], [13], and [14]. For grouping quires time is not good basis as the terms are overlapped in different times as explored in [11] and [15]. In [16] also refinement classes were studied for organizing search histories. Bayesian classifier was proved to be productive in such cases. In [17] query chains concept was explored that combines similar features and make use of thresholds.

Query clustering is another approach to group queries together. Many researchers followed this [18], [19], [6], [7], and [20]. In [6] and [7], building Bipartite graph concept was explored for grouping. Click graphs were explored in [18] for the same. This will group queries from different users and group them in a meaningful way. In [21] and [3] ranking of results were improving using Markov random walk approach.

III. PROPOSED FRAMEWORK

The aim of proposed system is to organize users search histories more meaningfully. The search engines like Yahoo, Google, and Bing are organizing user search histories in chronological order. The end users are to view the history of queries chronologically or date wise. This may not be in the best interest of the users always. The reason behind it is that users wanted to reuse the queries that they have issued earlier. Thus it is useful to organize search histories in some meaningful way other than chronological order. Human beings issue quires based on their requirements. The queries might have repeated ones and semantically similar ones. There comes data mining handy. Data mining is a process of discovering knowledge from historical data. The trends or patterns thus discovered form business intelligence. This will help in taking well informed business decisions. In this project, the aim is to mine the queries given by end users over a period of time. Similar queries are to be grouped together. When similar queries are grouped, the results are very useful in many applications. The real world applications of our proposed work are as follows.

- Query suggestions
- Result ranking
- Query alterations
- Sessionization
- Collaborative Search

These applications are widely used in the real world for making valuable decisions. The proposed system can help in grouping meaningful and related queries. When related queries are grouped together, they can be used in such applications to gain advantages. However, we do not use only exact match while finding similarity between the queries. We use semantic meanings of a query in order to find best matches in the other queries. Finding semantic meaning is not an easy task. Many technologies came into existence in order to group such queries or finding semantic meanings together. In this project, lexical analysis is used as part of information retrieval. The lexical analysis makes use of natural language processing in order to complete the similarity search.

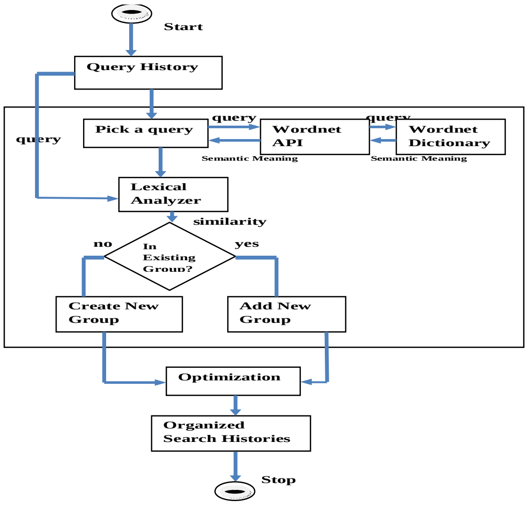
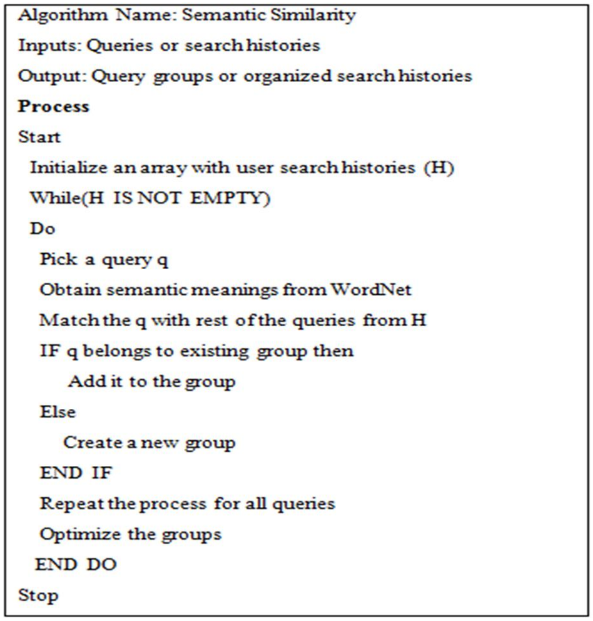


Fig. 4 – Architecture of the Proposed System

As can be seen in figure 2, the proposed approach is presented. However, very important thing here is the usage of WordNet dictionary which is a well known dictionary which has semantic meanings of words in English. In this project this dictionary is used in order to get semantic meanings. The algorithm used here is presented in listing 1.



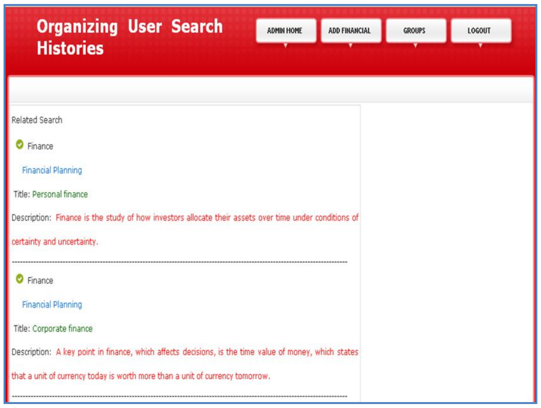
```
Algorithm Name: Semantic Similarity
Inputs: Queries or search histories
Output: Query groups or organized search histories
Process
Start
  Initialize an array with user search histories (H)
  While(H IS NOT EMPTY)
  Do
    Pick a query q
    Obtain semantic meanings from WordNet
    Match the q with rest of the queries from H
    IF q belongs to existing group then
        Add it to the group
    Else
        Create a new group
    END IF
    Repeat the process for all queries
    Optimize the groups
   END DO
Stop
```

Listing 1 – Proposed Algorithm

As can be seen in the listing 1 and the architectural diagram the framework picks a query from the search history. Give the query to lexical dictionary. The lexical analyzer gives the semantics of the search word. The semantic meanings are used to know whether the given query is meaningfully similar to other queries. If it is similar to other queries, they form a group. This process is repeated for all queries in the search history. Afterwards the groups are formed. Then the optimization process removes duplicates and eliminates empty groups in order to make the groups more meaningful.

IV. PROTOTYPE IMPLEMENTATION

The prototype application is implemented using web interface. It is to demonstrate the usefulness of grouping search history of users. The environment used for the development is a PC with 4 GB or RAM, Core 2 dual processor running Windows XP operating system. Java technologies used are Servlets and JSP. We also used MVC (Model View Controller) design pattern for its benefits like scalability, availability and maintainability. The implementation of mechanisms is made as described in [3]. An important screen of the web application the organization of user search history is presented in fig. 5.



Related Search

Finance

Financial Planning

Title: Personal finance

Description: Finance is the study of how investors allocate their assets over time under conditions of certainty and uncertainty.

Finance

Financial Planning

Title: Corporate finance

Description: A key point in finance, which affects decisions, is the time value of money, which states that a unit of currency today is worth more than a unit of currency tomorrow.

Fig. 5 – Web based UI showing grouping of users' search history

As can be seen in fig. 5, the search queries of user's search history are grouped together as per the mechanism presented in section III. The visualization of search history is also presented in fig. 6.

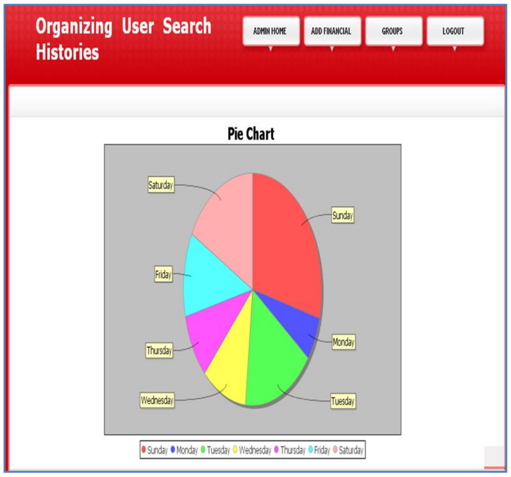


Fig. 6 – Visualization of search history

As can be seen in fig. 6, it is evident that the user's search history is broken into different days. The search volumes are presented in a pie chart. This will reflect the user's search behavior on different days of a week. However, the subsequent section shows more experimental results.

V. EXPERIMENTAL RESULTS

Experiments are made based on different mix of click and query graphs, varying damping factor, varying click importance, varying related queries, varying similarity threshold, varying recency weight, and varying time threshold.

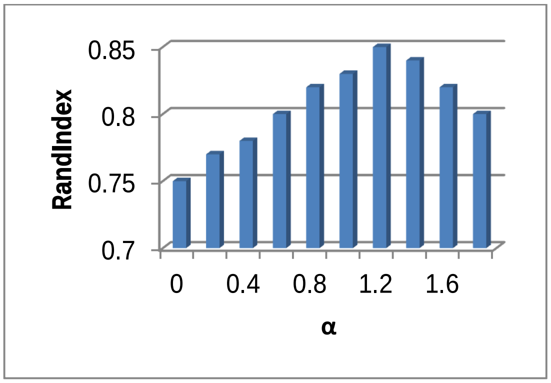


Fig. 7–Illustrates varying mix of query and click graphs

As can be seen in fig. 7, the horizontal axis represents weight of query edges that come from query reformulation graph while the vertical axis shows the performance based on RandIndex metric.

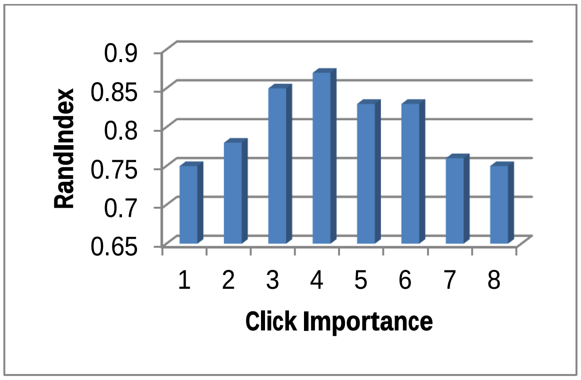


Fig. 8 – Illustrates varying the damping factor
As can be seen in fig. 8, the horizontal axis represents damping factor while the vertical axis shows the performance based on RandIndex metric.

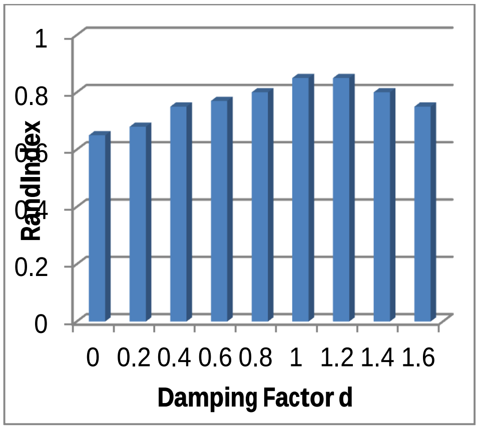


Fig. 9 – Illustrates varying click importance

As can be seen in fig. 9, the horizontal axis represents click importance while the vertical axis shows the performance based on RandIndex metric.

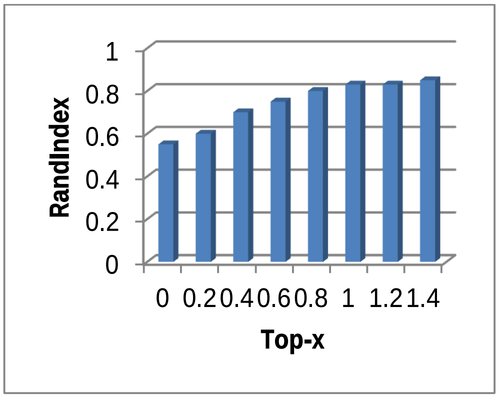


Fig. 10 – Illustrates varying the fraction of related queries

As can be seen in fig. 10, the horizontal axis represents fraction of related queries while the vertical axis shows the performance based on RandIndex metric.

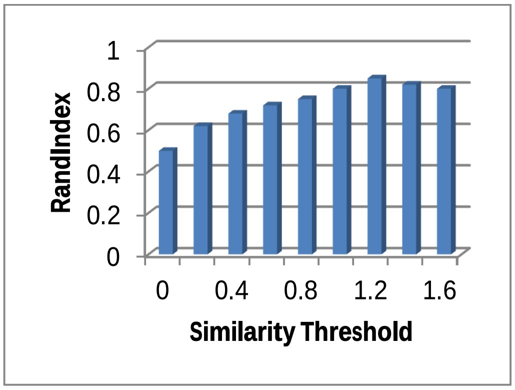


Fig. 11 – Illustrates varying the similarity threshold

As can be seen in fig. 11, the horizontal axis represents similarity threshold while the vertical axis shows the performance based on RandIndex metric.

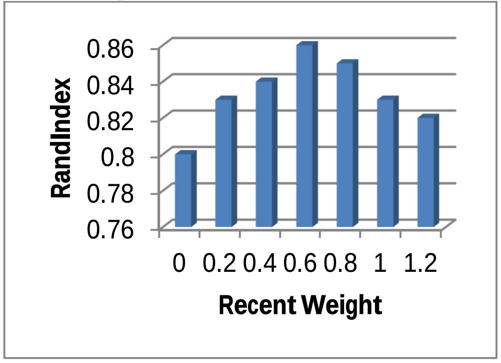


Fig. 12 – Illustrates varying the recency weight

As can be seen in fig. 12, the horizontal axis represents recency weight while the vertical axis shows the performance based on RandIndex metric.

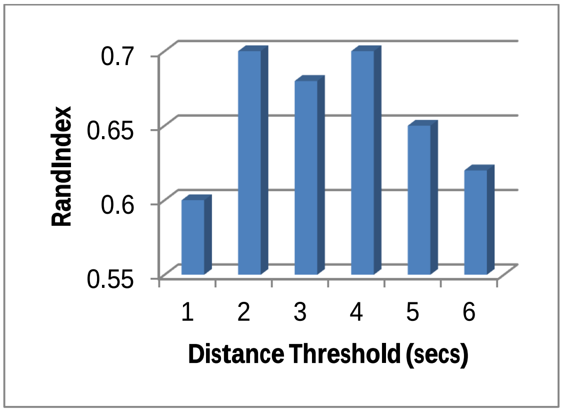


Fig. 13 – Illustrates varying the time threshold

As can be seen in fig. 13, the horizontal axis represents time threshold while the vertical axis shows the performance based on RandIndex metric.

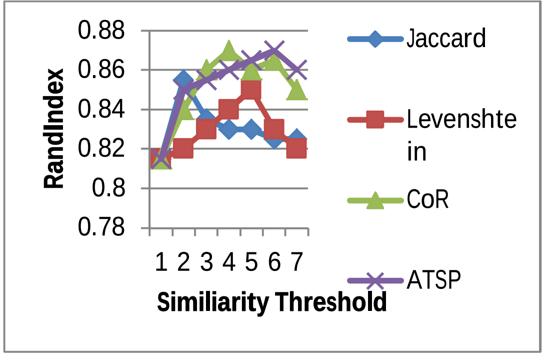


Fig. 14 – Illustrates varying the similarity threshold

As can be seen in fig. 14, the horizontal axis represents similarity threshold while the vertical axis shows the performance based on RandIndex metric.

VI. CONLCUSION

Historical search information is maintained by search engines like Google. However, the search engines organize search histories in only chronological order. It is useful if they organize the search histories in some meaningful way. For instance organizing search histories on the basis of relevancy of quires can help users to reuse searches with ease. There are many advantages to search engines as well. For instance the organized groups can be used by search engines for sessionization, collaborative search, query answering, and query modifications and so on. In this paper we built two algorithms in order to achieve this. We organize user's search histories based on the relevancy of queries and organize them well. We built a prototype application that demonstrates the proof of concept. The empirical results revealed that the application is useful to organize user's search histories into meaningful groups.

References

[1] A. Broder, "A taxonomy of web search," *SIGIR Forum*, vol. 36, no. 2, pp. 3–10, 2002.

[2] J. Teevan, E. Adar, R. Jones, and M. A. S. Potts, "Information reretrieval: repeat queries in yahoo's logs," in *SIGIR*. New York, NY, USA: ACM, 2007, pp. 151–158.

[3] Heasoo Hwang, Hady W. Lauw, LiseGetoor and AlexandrosNtoulas, "Organizing User Search Histories", IEEE 2012 Transactions on Knowledge and Data Engineering, Volume: 24 , Issue: 5.

[4] R. Jones and K. L. Klinkner, "Beyond the session timeout: Automatic hierarchical segmentation of search topics in query logs," in *CIKM*, 2008.

[5] P. Boldi, F. Bonchi, C. Castillo, D. Donato, A. Gionis, and S. Vigna, "The query-flow graph: Model and applications," in *CIKM*, 2008.

[6] D. Beeferman and A. Berger, "Agglomerative clustering of a search engine query log," in *KDD*, 2000.

[7] R. Baeza-Yates and A. Tiberi, "Extracting semantic relations from query logs," in *KDD*, 2007.

[8] P. Anick, "Using terminological feedback for web search refinement: A log-based study," in *SIGIR*, 2003.

[9] B. J. Jansen, A. Spink, C. Blakely, and S. Koshman, "Defining a session on Web search engines: Research articles," *Journal of the American Society for Information Science and Technology*, vol. 58, no. 6, pp. 862–871, 2007.

[10] L. D. Catledge and J. E. Pitkow, "Characterizing browsing strategies in the World-Wide Web," *Computer Networks and ISDN Systems*, vol. 27, no. 6, pp. 1065–1073, 1995.

[11] D. He, A. Goker, and D. J. Harper, "Combining evidence for automatic Web session identification," *Information Processing and Management*, vol. 38, no. 5, pp. 727–742, 2002.

[12] R. Jones and F. Diaz, "Temporal profiles of queries," *ACM Trans- actions on Information Systems*, vol. 25, no. 3, p. 14, 2007.

[13] A. L. Montgomery and C. Faloutsos, "Identifying Web browsing trends and patterns," *Computer*, vol. 34, no. 7, pp. 94–95, 2001.

[14] C. Silverstein, H. Marais, M. Henzinger, and M. Moricz, "Analysis of a very large Web search engine query log," *SIGIR Forum*, vol. 33, no. 1, pp. 6–12, 1999.

[15] H. C. Ozmutlu and F. C¸ avdur, "Application of automatic topic identification on Excite Web search engine data logs," *Information Processing and Management*, vol. 41, no. 5, pp. 1243–1262, 2005.

[16] T. Lau and E. Horvitz, "Patterns of search: Analyzing and modeling Web query refinement," in *UM*, 1999.

[17] F. Radlinski and T. Joachims, "Query chains: Learning to rank from implicit feedback," in *KDD*, 2005.

[18] J.-R. Wen, J.-Y.Nie, and H.-J. Zhang, "Query clustering using user logs," *ACM Transactions in Information Systems*, vol. 20, no. 1, pp. 59–81, 2002.

[19] J. Yi and F. Maghoul, "Query clustering using click-through graph," in *WWW*, 2009.

[20] E. Sadikov, J. Madhavan, L. Wang, and A. Halevy, "Clustering query refinements by user intent," in *WWW*, 2010.

[21] N. Craswell and M. Szummer, "Random walks on the click graph," in *SIGIR*, 2007.

AUTHORS

A.S.SALEEM BASHA Pursuing Mtech in Dr.KVSRIT ,KNL

V.Trilik kumar , Associate Professor