CUSTOMIZING USER SEARCH RECORDS

Ms. Vaibhavi P. Gawande[1], Mrs. Sanjivani T. Deokar[2]

[1,2] *Department of Computer Engineering, Lokmanya Tilak College Of Engineering, Koparkhairne, Navi Mumbai, Mumbai University, Maharashtra,(India)*

ABSTRACT

Users are increasingly pursuing multifaceted task-oriented goals on the Web, such as making travel arrangements, managing finances or planning purchases. Each step requires one or more queries, and each query results in one or more clicks on relevant pages. To support users in their long-term information quests on the Web, "Customizing User Search Records" helps to keep track of their queries and clicks while searching online. The aim is to organize a user's historical queries into groups in a vibrant and automated fashion. Automatic identification of query groups is helpful for a number of different search engine components and applications, such as query suggestions, result ranking, query alterations, sessionization, and collaborative search. The existing approaches rely on textual similarity or time thresholds. In this paper we use query reformulation and click graphs that will contain useful information on user behavior when searching online. Then the query fusion graph merges the information in the query reformulation graph and click graph and the approach based on probabilistic random walks over the query fusion graph outperforms time-based and keyword similarity based approaches to customize user search records in different groups. This is the robust approach that leverage search query logs.

Keywords- *Context Vector, Query Click Graph, Query Fusion Graph, Query Image, Query Reformulation Graph*

I. INTRODUCTION

As the size and richness of information on the Web grows, so does the diversity and the difficulty of tasks that users try to accomplish online. Users are no longer satisfied with issuing simple navigational queries. The various studies on query logs (e.g., Yahoo's [2] and AltaVista's [3]) disclose that only about 20% of queries are navigational. The rest are informational or transactional in nature. This is because users now follow much broader informational and task-oriented goals such as arranging for future travel, managing their finances, or planning their purchase decisions. However, the primary means of accessing information online is still through keyword queries to a search engine. A complex task such as travel arrangement has to be broken down into a number of co-dependent steps over a period of time. The important step towards enabling services and features that can help users during their complex search quests online is the capability to identify and group related queries together. Users can manipulate search history by manually editing and organizing related queries and clicks into groups, or by sharing them with their friends. The query grouping allows the search engine to better

understand a user's session and potentially tailor that user's search experience according to user needs and also assist other users by promoting task-level collaborative search.

II. PRELIMINARY

The goal is to automatically organize a user's search history into query groups, each containing one or more associated queries and their corresponding clicks. Each query group corresponds to a minute information need that may require a small number of queries and clicks related to the same search goal. The Fig 1(a) shows set of queries of a real user on the search engine over the period of one day. The user's search history is organized into a set of query groups in a robotic and lively fashion. Each query group is a collection of queries by the same user that are relevant to each other around a common informational need. These query groups are dynamically updated as the user concerns new queries, and new query groups may be created over time.

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

Figure1 (a) User's Search History

The Fig 1(b) shows corresponding query groups. The first query group contains all the queries that are related to saturn automobiles. The other groups respectively pertain to barbados vacation, sprint phone, financials, and wii game console.

Group 1	Group 2	Group 3	Group 4	Group 5
saturn vue hybrid saturn vue saturn dealers saturn hybrid review	snorkeling barbados hotel caribbean cruise tripadvisor barbados expedia	sprint slider phone sprint latest model cell phones	financial statement bank of america	toys r us wii best buy wii console wii gamestop gamestop discount used games wii

Figure1 (b) Query Groups

III. QUERY RELEVANCE USING SEARCH LOGS

A robust relevance measure is used to identify similar query groups beyond the approaches that simply rely on the textual content of queries or time interval between them. This approach is query relevance which makes use of search logs in order to determine the relevance between query groups more effectively. In fact, the search history of a large number of users contains signals regarding query relevance, such as which queries tend to be issued closely together (query reformulations), and which queries tend to lead to clicks on similar URLs(query clicks). Such signals are user-generated and are likely to be more robust, especially when considered at scale. The relevance between query groups is measured by exploiting the query logs and the click logs simultaneously.

The measure of relevance is aimed at capturing two important properties of relevant queries, such as: (1) queries that frequently appear together as reformulations and (2) queries that have induced the users to click on similar sets of pages.

The three search behavior graphs are introduced to capture the abovementioned properties. These graphs are used to compute query relevance and then incorporate the clicks following a user's query in order to enhance our relevance metric.

3.1 Search Behavior Graphs

The three types of graphs are derived from the search logs of a commercial search engine. The query reformulation graph, QRG is used to represent the relationship between a pair of queries that are likely reformulations of each other. The query click graph, QCG is used to represent the relationship between two queries that frequently lead to clicks on similar URLs. The query fusion graph, QFG, is used to merge the information in the previous two graphs. All three graphs are defined over the same set of vertices VQ, which consists of queries which appear in at least one of the graphs, but their edges are defined differently.

3.1.1 Query Reformulation Graph

The relevant queries are identified by considering query reformulations that are typically found within the query logs of a search engine. If two queries that are issued successively by many users occur frequently enough, they are likely to be reformulations of each other. The relevance between two queries issued by a user is measured by the time-based metric, sim_{time}, makes use of the interval between the timestamps of the queries within the user's search history. In contrast, the statistical frequency is defined with which two queries appear next to each other in the entire query log, over all of the users of the system.

The query reformulation graph, QRG = (V_Q, E_{QR}) is constructed based on the query logs, whose set of edges, E_{QR}, are constructed as follows: for each query pair (q_i, q_j), where q_i is issued before q_j within a user's day of activity, then count the number of such occurrences across all users' daily activities in the query logs, denoted $count_r(q_i, q_j)$. Assuming infrequent query pairs are not good reformulations of each other, so filte out infrequent pairs and include only the query pairs whose counts exceed a threshold value, ד r. For each (qi, qj) with $count_r$(qi, qj) ≥ד r, add a directed edge from qi to qj to EQR.

3.1.2 Query Click Graph

The relevant queries are captured from the search logs by considering queries that are likely to induce users to click frequently on the same set of URLs. This property of relevant queries is captured through a graph called the query click graph, QCG. The graph starts by considering a bipartite click-through graph, CG = ($V_Q UV_U$, EC). The click graph has two distinct sets of nodes corresponding to queries, V_Q, and URLs, V_U, extracted from the click logs. There is an edge (q_i, u_k) € EC, if query q_i was issued and URL u_k was clicked by some users. The CG is used to identify pairs of queries that frequently lead to clicks on similar URLs.

The query click graph is derived from click graph, QCG =(V_Q, E_{QC}), where the vertices are the queries, and a directed edge from q_i to q_j exists if there exists at least one URL, u_k, that both q_i and q_j link to in CG.

3.1.3 Query Fusion Graph

The query reformulation graph, QRG, and the query click graph, QCG, confine two important properties of

relevant queries respectively. The effective use of both properties is made by combining the query reformulation information within QRG and the query click information within QCG into a single graph called as Query Fusion Graph, QFG =(V_Q, E_{QF}). E_{QF} contains the set of edges that exist in either E_{QR} or E_{QC}.

3.2 Computing Query Relevance

The edges in QFG correspond to pairs of relevant queries extracted from the query logs and the click logs. This approach is not effective to use the pair wise relevance values directly expressed in QFG because it fails to capture relevant queries that are not directly connected in QFG.

A more general approach, Markov chain for q, MC_q, over the given graph QFG is used to obtain query relevance and compute the stationary distribution of the chain. The stationary distribution is referred as the fusion relevance vector of q, rel^F_q , and use it as a measure of query relevance. The stationary probability distribution of MCq can be estimated using the matrix multiplication method, where the matrix corresponding to MC_q is multiplied by itself iteratively until the resulting matrix reaches a fix point. As thousands of users issuing queries and clicks in real-time and the huge size of QFG, it is impracticable to perform the expensive matrix multiplication to compute the stationary distribution whenever a new query comes in. So that the most efficient Monte Carlo random walk simulation method is used on QFG to approximate the stationary distribution for q.

3.3 Incorporating Current Clicks

The user activities include not only query reformulations but also clicks on the URLs following each query submission. The clicks of a user help to gather his/her search interests behind a query q and thus identify queries and query groups relevant to q more effectively. The click information of the current user is used to expand the random walk process to improve query relevance estimates. This approach is independent of modeling the query click information as QCG to build QFG. The clicks of the current user is used to understand his/ her search intent behind the currently issued query, while clicks of massive users in the click logs are aggregated into QCG to capture the degree of relevance of query pairs through commonly clicked URLs.

IV. QUERY GROUPING USING THE QFG

For each query, a query image is maintained which represents the relevance of other queries to this query. For each query group, a context vector is maintained which aggregates the images of its member queries to form an overall representation. A similarity function $sim_{rel\ is}$ proposed for two query groups based on these concepts of context vectors and query images. The definitions of query reformulation graph, query images, and context vectors are crucial ingredients, which provide significant novelty to the Markov chain process for determining relevance between queries and query groups.

4.1 Context Vector

For each query group, a context vector is maintained to compute the similarity between the query group and the user's latest singleton query group. The context vector for a query group s, denoted cxt_s, contains the relevance scores of each query in VQ to the query group s, and is obtained by aggregating the fusion relevance vectors of the queries and clicks in s. If s is a singleton query group containing only $\{q_{s1}, clk_{s1}\}$, it is defined as the fusion relevance vector $rel(q_{s1},clk_{s1})$.

4.2 Query Image

The fusion relevance vector of a given query q, rel_q, captures the degree of relevance of each query q' € VQ to q. This is not effective or robust to use rel_q itself as a relevance measure for online query grouping. The new concept is introduced that is image of q, denoted I(q), that expresses q as the set of queries in VQ that are considered highly relevant to q. The query image I(q) is generated by including every query q' whose relevance value to q, $rel_q(q')$, is within top-X percentage. The queries are sorted by relevance, and find the cutoff such that the sum of the relevance values of the most relevant queries accounts for X% of the total probability mass. The size of the image of the query is typically very small compared to the total number of possible queries in QFG. The image of a query group s, I(s), is defined in the same way as I(q) except that the context vector of s, cxt_s, is used in the place of $rel_{(q,clk)}$.

V. CONCLUSION

In this paper we have discussed to organize a user's historical queries into groups in a dynamic and automated fashion. The query reformulation and click graphs contain useful information on user behavior when searching online and this information is used effectively for the task of organizing user search histories into query groups. The two graphs are combined into a query fusion graph. The system uses approach that is based on probabilistic random walks over the query fusion graph outperforms time-based and keyword similarity based approaches.

REFERENCES

[1] Heasoo Hwang, Hady W. Lauw, Lise Getoor and Alexandros Ntoulas "Organizing User Search Histories", IEEE 2012 Transactions on Knowledge and Data Engineering, Volume: 24.

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

[3] A. Spink, M. Park, B. J. Jansen, and J. Pedersen, "Multitasking during Web search sessions," Information Processing and Management, vol. 42, no. 1, pp. 264–275, 2006.

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

[7] 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.

[8] 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.

[9] 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.

[10] R. Jones and F. Diaz, "Temporal profiles of queries," ACM Transactions on Information Systems, vol. 25, no. 3, p. 14, 2007.