Research Paper
Impact Factor: 3.567
Peer Reviewed Journal

IJMDRR
E- ISSN –2395-1885
ISSN -2395-1877

CUSTOMIZED SEARCH ENGINE USING TASK TRAIL

M. Nishanthi
Computer Science and Engineering, Panimalar Institute of Technology.

Abstract
Personalized search refers to search experiences that are tailored specifically to an individual's interests by incorporating information about the individual beyond specific query provided. Especially people working in a software development organization find it increasingly difficult to get relevant results to their searches. Difficulty in web searches has given rise to the need for development of personalized search engines. Personalized search engines create user profiles to capture the users' personal preferences and as such identify the actual goal of the input query. Since users are usually reluctant to explicitly provide their preferences due to the extra manual effort involved, the search engine faces the entire burden of predicting the user's preferences and intentions behind a query in order to yield more relevant search results. In this paper we combine task level analysis of user's search logs and semantic analysis of the user's query in order to personalize to user's search results. Most personalization methods focuses on the creation of one single profile for a user and applies the same profile to all of the user's queries. We believe that different queries from a user should be handled differently because a user's preferences may vary across queries. We are creating a personalized search for a software development organization by creating TASK or domain based profile rather than individual user based profile. We then leverage this task based profile and perform semantic analysis to optimally personalize the user's searching process.

Key Words: Task Trail, Quest Trail, Meta Search Engine.

1. Introduction

Pers**o**nalised search refers to search experiments that are tailored specifically to an individual's interests. It aims to resolve the ambiguity of query terms .To know more about the ambiguity that arises in search engines let us take the instance of "Java". When the user searches about Java there are three possibilities of results (i.e.) the results can be about Java Sea in Indonesia or about the Java coffee bean or the programming language. This is an example for ambiguity.

Difficulty in web searches has given rise to the need for development of personalised search engine .It is important to introduce personalization in a software organization where the employees are reluctant to provide information. There are two types of user behaviour (i.e.) search behaviour and browser behaviour. Search behaviour is everything the user enters in the search engine to search for the information needed. Browser behaviour involves surfing, user types a URL address in the browser, clicking a bookmark or forward page in the browser etc.

Searches can be analysed in three levels, (a) query level, (b) task level and (c) session level. In query level it fails to capture the interleaving relationships between tasks. If we analyse the search logs based on session (i.e. session level) the tasks will be interleaved .It is difficult to identify what the user is doing because the sessions are chronologically ordered. If we analyze in task level the topics will be more consistent and relevant to each other. This will help us to understand the intentions behind a user's search.

In our paper we bring in two studies semantic analysis and genetic algorithm for personalizing the search process in the search engine. To get a clear picture of our study we also discuss about Meta search engine, personalization and search.

1.1. Search Engine

A search engine is a type of computer software used to search data in the form of text or a database for specified information. Search engines normally consist of spiders (also known as bots) which roam the web searching for links and keywords. They send collected data back to the indexing software which categorizes and adds the links to databases with their related keywords. When you specify a search term the engine does not scan the whole web but extracts related links from the database. Search is the heart of the web. It is how we navigate the web. All the information available in the web will become inaccessible if we don't have a search engine to enter our queries.

1.2. Semantic Analysis

Semantic analysis is nothing but filters that progressively eliminate more and more input strings until you are left with only valid data. Semantic search is different from Boolean search as apples are different from oranges. The transition to semantic search also marks the transition on the web as we go from websites to people .The web continues to be made of websites. In websites we get to find information, consume news and buy stuff. In order to understand natural language and search queries,

it has to understand what these words really mean .To do that semantic search uses a number of techniques which include using,

1. Resource Description Framework Attributes (RDFa)
2. Keyword to Concept Mapping
3. Graph Pattern Recognition
4. Entity Extraction
5. Fuzzy Logic

1.3. Metasearch Engine

Meta search engine is a search tool that uses other search engine's data to produce their own results from the Internet. Meta search engines take input from a user and simultaneously send out queries to third party search engines for results. Sufficient data is gathered, formatted by their ranks and presented to the users.

Meta search engine accepts a single search request from the user. This search request is then passed to multiple search-engine's database. Meta search engine does not create a database of webpages but generates a virtual database to integrate data from multiple sources.

1.4. Genetic Algorithm

In the field of artificial intelligence, a genetic algorithm (GA) is a search heuristic that mimics the process of natural selection. This heuristic (also sometimes called a Meta heuristic) is routinely used to generate useful solutions to optimization and search problems. Genetic algorithms belong to the larger class of evolutionary algorithms (EA), which generate solutions to optimization problems using techniques inspired by natural evolution, such as inheritance, mutation, selection, and crossover. It is a very powerful and non-traditional optimization technique. It is based on the Darwinian Theory "Survival of the Fittest". Only the fittest will survive and reproduce and successive generations will become better and better compared to previous generations.

2. Related Works

Web search logs record user activities on search engines, such as queries and clicks. Search trails record the footprints left by users in their search processes. In the literature of studying search trails, much of previous work have applied them in the applications of user satisfaction analysis, ranking function evaluation, query suggestion, etc. Here we classify related works into four categories: (1) web page utility (2) user satisfaction analysis (3) query suggestion and (4) web page ranking.

2.1. Web Page Utility

Session and task are two primary types of user search behaviour segmentations. The term *session* was proposed in [7], [8]. Cat ledge [8] analysed user browsing logs captured from client-side user events. He found that 25.5 minutes timeout is good for separating consecutive user activities into different sessions. Silverstein [7] defined "session" as "a series of queries by a single user made within a small range of time" in his study of AltaVista search logs, where they used 5 minutes as the timeout threshold. He [10] proposed to detect session boundary based on time and query reformulation patterns and found that 10 to 12 minutes timeouts are good. Jansen [11] clarified the session as "a series of interactions by the user towards addressing a single information need" and found that about 30 minutes timeout is better than others. As a result, later studies [4]–[6], [11], [18] often used 30 minutes timeout for session segmentation. Considering the multitasking behaviours within a session, Jones and Klinkler [12] proposed to classify query pairs into a same task via features based on time, word, web search results, etc. Their approach achieved about 90% accuracy in task boundary detection and same task identification. Boldi [19] applied the *query flow graph* in finding logical session and query recommendation. They formulated the problem of mining logical sessions as an Asymmetric Travelling Salesman Problem.

2.2. User Satisfaction Analysis

Hassan [14] proposed to formulate the search process by Markov models. The experiment results on 2,712 labelling goals showed that their approach could model the search process well and have a better prediction of user goal success than discounted cumulative gain (DCG). White [2], [6] conducted extensive studies on search and browser logs. They found that following the query trails, users can find more useful information. White [22] found that short-term user interests could be well captured by the previous submitted queries and visited web pages within the same session. Olson and Chi [23] proposed Scent Trail to combine searching and browsing activities into a single interface, and they found it can help users in finding information faster than by only searching or browsing alone.

2.3. Query Suggestion

The related queries mined from sessions and click through bipartite graphs can be used for query suggestion. Beeferman [30] proposed to group queries and URLs in the click graph for query suggestion. Huang [15] proposed to use co-occurred query

pairs from sessions as suggestions. Jones [16] generated query substitution for sponsored search and applied log likelihood ratio (LLR) to measure the correlation of query pairs. Boldi [19] proposed *query flow graph* and applied query flow graph in query recommendation. Mei [17] proposed to use hitting time of query pairs on a click graph for query suggestion. Cao [4] combined both click-through and session logs to mine concept sequences for context-aware query suggestion. After grouping similar queries into concepts via their efficient algorithm, suggestions can be generated at concept level. Song [31] mined the term transition graph from consecutive query pairs and applied term transition graph into tail query suggestion. Jain [32] proposed to generate high utility query suggestions based on session logs, click-through logs, and web corpus.

2.4. Web Page Ranking

Utilizing search and browser logs to enhance ranking is a promising and important direction. There are several methods to enhance ranking by web logs: (1) improving the page importance estimation of documents [18], (2) improving the relevance of query-document pairs [24], [25], (3) improving the evaluation of ranking functions using log-based measures [26]. Craswell and Szummer [24] proposed a backward random walk on click graph and validated its effectiveness in image retrieval. To address the sparseness problem of the click-through bipartite graph, Gao [25] proposed two smoothing techniques for estimating the relevance of query document pairs. Implicit metrics extracted from web logs can be used for measuring ranking functions, which can save the cost of traditional ranking metrics (e.g., NDCG) based on human annotation. Recent work [26], [28] showed that results of interleaving experiments can be used to measure ranking functions. They showed that Interleaving experiments are reliable, sensitive, and also have high correlation with standard measures. Hassan [29] showed that a task level metric can be sensitive to tell different ranking functions apart.

3. Proposed Work

Nowadays, computers and internet has become inseparable parts of our life. Throughout the world, web has become the best source of abundant information. Search engines play a key role in finding out the information; they are enhanced with new advanced search technologies. Though search engines find much information with one key word, fail to provide the accurate, exact data that is required. Through research it has found that users will not have much interest if it is delayed and can't afford to spend time with queries. This time constrain can be observed in many situations. Hence the most significant point in the applications of the search engines is to find accurate information immediately. This aspect of accurate and immediate information for a search can be solved by personalized web environments. There is an increasing importance to the personalized web environment. In this paper we propose to personalize the searching process using semantic analysis and task analysis.

In previous works much importance is given to areas like better browsing, localization, question and answer methodology, visual results presentation and modified web search. Each technology relates to a different aspect of people's information behaviors; showing that different technologies should be developed to answer different demands. In our paper we mainly focus on personalizing the searching process for people working in a software development organization (analysts, developers testers, maintenance team members), who find it increasingly difficult to get relevant results to their searches. We build group profiles based on either the domain in which software product is to be developed or project basis.

Till now not much development is observed in web personalization field because individual web search behavior has not changed much. The main challenge in web personalization, is to read the mind of the users. This imposes a very big challenge because the words used for any search are limited to two or three words. Some of the issues in Web searching are (1) *Structuring Queries* i.e. the difficulty faced by users are properly structuring queries, namely applying the rules of a particular system, especially Boolean operators e.g., AND, OR, NOT and term modifiers e.g. '+', '!'. (2) *Spelling* i.e. the user tends to misspell their queries without even realizing it. (3) *Query Refinement* i.e. Many times the searchers do not refine their query, even if there may be other terms that relate directly to their needed information. (4) *Managing Results* i.e. mostly, the user queries are extremely broad, resulting in an unmanageable number of results. Few searchers view more than the first ten or twenty documents from the result list.

3.1. Task Analyais

We define a task to be an atomic user information need (goal), whereas a task trail represents all user activities within that particular task, such as query reformulations, URL clicks. Previously, Web search logs have been studied mainly at session or query level where users may submit several queries within one task and handle several tasks within one session. Task level analysis of search log provides a better understanding of user's interests or goal, since it performs better in modelling user's profile. Thus the user behaviour can be studied and noted from the tasks he performs in the search engine. Thus task identification is important. We make use of the same task extraction algorithm used in [42].

3.2. Semantic Analysis

We analyse the user's queries at a semantic level using vocabulary or ontology based system like ODP or yahoo! Directory. Optimal results from semantic analysis are chosen using genetic algorithm, where only the results that are most suitable to the users profile and interests are presented to the user. Genetic algorithm aids with machine learning and supports the search engine to understand the user's mind while searching. Optimality of the results from semantic analysis is based on the user's profile that is built and the results of task analysis.

4. Other Techniques

Some of the other techniques that are currently being used for personalization are briefly listed below. Query rewriting, Semantic content filtering, Re-ranking, semantic celebrative filtering, User modelling or profiling and analysis of search logs are techniques that are used to improve the relevancy in the search results for the user, while reducing their effort.

4.1. Personalisation by Query Rewriting

Here the query is elaborated by the user to personalize the search result. For example the query is to find the Chinese restaurants located in the city of New York. Here in this technique New York is added to the search query and taken as "Chinese restaurants New York "and the search results are given for this query. The search engine will now give the results for Chinese restaurants that are in New York. The problem over here is there are chances where the user may not be clear about the location.

4.2. Semantic Content Filtering

It is based on semantic relation. It helps in addressing the two most significant problem which is encountered during traditional content based filtering.

1. Cold start problem
2. Over specialization problem

4.3. Personalization by Reranking

Page re-ranking is used mainly to take the advantage of user's location. Initially some 'k' documents are taken that checks if the user location matches with original query if it matches it rearranges them in increasing ranks. Users of same location are put into one group. The re-ranking occurs by their ranking score and text matching that checks with the user location and its variations.

4.4. Semantic Collaborative Filtering

Semantic collaborative filtering uses semantic match. It enhances the performance of traditional collaborative filtering recommendation system. By recommending items that have high semantic similarity, it reduces the cold start problem. To reduce item scattering problem in semantic collaborative filtering users and items are mapped.

5. Conclusion

We believe that different queries from a user should be handled differently because a user's preference may vary across queries. Users often perform multiple tasks during their search processes. Statistical results on 0.5 billion sessions from web search logs showed that: (1) about 30% of sessions contain multiple tasks, and (2) about 5% of sessions contain interleaved tasks [0]. In a software development organization there is a special need for task – specific or domain – specific ranking. Applying TASK level analysis of the search log for constructing a personalized search engine for software developers will make the searching process much easier. In this paper we propose an effective approach to personalize the searching process especially for software developers. It is clearly seen that the combination of task analysis and semantic analysis is an effective approach for personalization of searching process and it is better than any of the currently used techniques for personalization.

References

1. Fox, S., Karnawat, K., Mydland, M., Dumais, S. and White, T., "Evaluating implicit measures to improve web search," *ACM Trans. Inf. Syst.*, vol. 23, pp. 147–168, 2005.
2. White, R. and Huang, J., "Assessing the scenic route: measuring the value of search trails in web logs," ser. SIGIR '10. ACM, 2010, pp. 587–594.
3. White, R., Bennett, P. and Dumais, S., "Predicting short-term interests using activity-based search context," ser. CIKM '10, 2010, pp. 1009–1018.
4. Cao, H., Jiang, D., Pei, J., He, Q., Liao, Z., Chen, E. and Li, H., "Context-aware query suggestion by mining click-through and session data," in *KDD '08*, 2008, pp. 875–883.
5. Xiang, B., Jiang, D., Pei, J., Sun, X., Chen, E. and Li, H.,"Context-aware ranking in web search," ser. SIGIR '10. ACM, 2010, pp. 451–458.

6. White, R., Bilenko, M. and Cucerzan, S., "Studying the use of popular destinations to enhance web search interaction," ser. SIGIR '07, 2007, pp. 159–166.
7. Silverstein, C., Henzinger, M.R., Marais, H. and Moricz, M.,"Analysis of a very large web search engine query log," *SIGIR Forum*, vol. 33, pp. 6–12, 1999.
8. Catledge, L.D. and Pitkow, J.E., "Characterizing browsing strategies in the world-wide web," *Computer Networks and ISDN Systems*, vol. 27, no. 6, pp. 1065–1073, 1995.
9. Liao, Z., Song, Y., He, L.-w. and Huang, Y., "Evaluating the effectiveness of search task trails," ser. WWW '12, 2012, pp. 489–498.
10. He, D., G¨oker, A. and Harper, D.J., "Combining evidence for automatic web session identification," *Inf. Process. Manage.*, vol. 38, no. 5, pp. 727–742, 2002.
11. Jansen, B., Spink, A. and Kathuria, V., "How to define searching sessions on web search engines," ser. WebKDD '06, 2007, pp. 92–109.
12. Jones, R. and Klinkner, K.L., "Beyond the session timeout: automatic hierarchical segmentation of search topics in query logs," ser. CIKM '08, 2008, pp. 699–708.
13. Lucchese, C., Orlando, S., Perego, R., Silvestri, F., and Tolomei, G., "Identifying task-based sessions in search engine query logs," ser. WSDM '11, 2011, pp. 277–286.
14. Hassan, A., Jones, R., and Klinkner, K., "Beyond dcg: user behavior as a predictor of a successful search," ser. WSDM '10, 2010, pp. 221–230.
15. Huang, C.K., Chien, L.F. and Oyang, Y.J., "Relevant term suggestion in interactive web search based on contextual information in query session logs," *Journal of the American Society for Information Science and Technology*, 2003.
16. Jones, R., Rey, B., Madani, O. and Greiner, W., "Generating query substitutions," ser. WWW '06. ACM, 2006, pp. 387–396.
17. Mei, Q. and Zhou, D. and Church, K., "Query suggestion using hitting time," ser. CIKM '08. ACM, 2008, pp. 469–478.
18. Liu, Y., Gao, B., Liu, T.-Y., Zhang, Y., Ma, Z., He, S. and Li, H., "Browserank: letting web users vote for page importance," ser. SIGIR '08, 2008, pp. 451–458.
19. Boldi, P., Bonchi, F., Castillo, C., Donato, D., Gionis, A. And Vigna, S., "The query-flow graph: model and applications," ser. CIKM '08, 2008, pp. 609–618.
20. Donato, D., Bonchi, F., Chi, T. and Maarek, Y., "Do you want to take notes? identifying research missions in yahoo! Search pad," ser. WWW '10, 2010, pp. 321–330.
21. Kotov, A., Bennett, P., White, R., Dumais, S. and Teevan, J., "Modeling and analysis of cross-session search tasks," ser. SIGIR '11, 2011, pp. 5–14.
22. White, R., Bailey, P. and Chen, L., "Predicting user interests from contextual information," ser. SIGIR '09, 2009, pp. 363–370.
23. Olston, C. and Chi, E.H., "Scenttrails: Integrating browsing and searching on the web," *ACM Trans. Comput.-Hum. Interact.*, vol. 10, pp. 177–197, September 2003.
24. Craswell, N. and Szummer, M., "Random walks on the click graph," ser. SIGIR '07, 2007, pp. 239–246.
25. Gao, J., Yuan, W., Li, X., Deng, K. and Nie, J.-Y., "Smoothing click through data for web search ranking," ser. SIGIR '09, ACM, 2009, pp. 355–362.
26. Radlinski, F. and Craswell, N., "Comparing the sensitivity of information retrieval metrics," ser. SIGIR '10, 2010, pp. 667–674.
27. Shen, X., Tan, B. and Zhai, ChengXiang, "Context-sensitive information retrieval using implicit feedback," ser. SIGIR '05, 2005, pp. 43–50.
28. Chapelle O., Joachims T., Radlinski F. and Yisong Yue, "Largescale validation and analysis of interleaved search evaluation," *ACM Trans. Inf. Syst.*, vol. 30, no. 1, p. 6, 2012.
29. Hassan, A., Song, Y. and He, L.-w., "A task level user satisfaction metric and its application on improving relevance estimation," ser. CIKM '11, 2011.
30. Beeferman, D. and Berger, A., "Agglomerative clustering of a search engine query log," ser. KDD '00, New York, NY, USA, 2000, pp. 407–416.
31. Song, Y., Zhou, D., and He, L.-w., "Query suggestion by constructing term-transition graphs," ser. WSDM '12, 2012, pp. 353–362.
32. Jain, A., Ozertem, U. and Velipasaoglu, E., "Synthesizing high utility suggestions for rare web search queries," ser. SIGIR '11, 2011, pp. 805–814.
33. Vapnik V., *Statistical learning theory*. Wiley, 1998.
34. Wang, H., Song, Y., Chang, M.-W., He, X., White, R. and Chu W., "Learning to extract cross-session search tasks," in *WWW*, 2013, pp. 1353–1364.
35. Teevan, J., Adar, E.. Jones, R. and Potts M., "Information reretrieval: repeat queries in yahoo's logs," ser. SIGIR '07, New York, NY, USA, 2007, pp. 151–158.
36. Shen, D., Pan, R., Sun, J.-T, Pan, J., Wu, K., Yin, J. and Yang, Q., "Q2c@ust: our winning solution to query classification in kddcup 2005," *SIGKDD Explor. Newsl.*, vol. 7, pp. 100–110, 2005.
37. Swain, M. and Ballard, D., "Color indexing," *International Journal of Computer Vision*, vol. 7, pp. 11–32, 1991.
38. Liao, Z., Jiang, D., Chen, E., Pei, J., Cao, H. and Li, H., "Mining concept sequences from large-scale search logs for context aware query suggestion," *ACM Trans. Intell. Syst. Technol.*, vol. 3, pp. 17:1–17:40, 2011.
39. Deng, H. and Irwin, K. and Michael L., "Entropy-biased models for query representation on the click graph," ser. SIGIR '09, 2009, pp. 339–346.
40. Haveliwala T., Kamvar, S., and Jeh G., "An analytical comparison of approaches to personalizing page rank," in *Technical Report in Stanford*, 2003.
41. Fleiss, J. L., "Measuring nominal scale agreement among many raters," *psychological bulletin*, vol. 76, 1971.
42. IEEE TRANSACTIONS O,NV OKLN.O26W, LNEOD.G2E, FAENBDR EUNAGRINYE 2E0R1I4NG,VOL.26, NO.12, APRIL 2014 "Task Trail: An Effective Segmentation of user search behaviour.