International Journal of Future Generation Communication and Networking
Vol. 13, No. 1, (2020), pp. 125 - 136

Web Search Personalization Using Query Optimization

Sunny sharma[1], Vijay Rana[2], Gurpinder Singh[3]
[1]Chandigarh University, Punjab, India
[2,3]Sant Baba Bhag Singh University, Jalandhar, India
Sunny202658@gmail.com

Abstract

Web search queries are the main source to retrieve the information online. However it is cleared that queries that we search on the web are frequently unclear or ambiguous which fail to retrieve the desired information. In this paper we proposed a methodology for identifying the intended meanings of queries by enriching the original query into the proficient query using past behavior of the user. Our "Web Search Personalizer" consists of three modules: Interface module, User Profiling module and Meta- Search module. These three modules interact together through well- defined interfaces to generate personalize results. The proposed model utilizes the current search engines for personalizing the web search results. Our results prove that query expansion based personalization is indeed effective.

Keywords: *Web Search Personalization, User Profiling, Query Reformulation, Meta Search Engine*.

1. Introduction

The Information Retrieval community has given a lot of attention to web search personalization. Search Personalization is the development of offering the most closed results intended by the user for his/her query. Recently a number of methodologies for providing the personalized results have been projected. Some of these methodologies [21, 22, 23] are based on the users' geographical location. These geographical location based approaches retrieved results related to the user's language and the other attributes with no interest on the user's preferences. Other some methodologies rely on re- rank the web results retrieved from search engines [24, 25]. One of many disadvantages of these approaches is they rely on original search query without taking it into considerations. The most effective personalization approaches rely on designing a rich user profile [26, 27]. This user profile is consisting of all preferences like previous submitted queries, past clicked documents, time spend on a particular document, and user feedback.

More generally, Personalization can be achieved by two ways: (i) by reformulating the original query submitted by the user, (ii) by reranking the search results provided by the search engines. The existing retrieval systems often fail for ambiguous queries (e.g. apple) that can refer to multiple entities. For such queries, non-personalized search engines retrieve results on the basis of importance of the web pages like ranking of the page [1]. According to our idea, this process might not be always good as shown in the figure 1. The figure 1.1 shows the results of the search term "fruit". The figure 1.2 shows the results of search term "apple" in Google search engine in that retrieved links for apple computer due to higher ranking of the pages. However the user might not be interested in the results as apple phone. Here the search engine should relate the user query to the user context and provide the results based on his past behavior.

On receiving the documents from the search engine, the users not only satisfy their requirements but also provide an implicit feedback to the search engine. The search engine maintains all these interactive information in a log file which includes search queries submitted by the users and the documents clicked by the users. Since a log file contains rich information about the users, the process of analyzing these log file becomes an active research area. The information in the log a file has used for many tasks, one of which is query

reformulation [2, 3, 4, 5]. Query reformulation can help to resolve the ambiguity by changing the original query to the query that is a better match to the relevant documents.

In this paper, an advanced multi-agent method called "Web Search Personalizer" (WSP) is proposed that modify the original query to personalize query which is further submitted to the syntactic search engines in order to get personalized results. The proposed model depends upon constructing a user profile that speaks to client interests and utilizing it in the web search process [6, 7, 8, 9].

The proposed model is used here by creating several agents to resolve the various issues and phases of personalization to improve the accuracy of information retrieval and recall assessment criteria. The model gains the benefits of working with the current search engines to access and defuse the effects of web searches. In the proposed model, a user profile is built from initial and specific information and retained through the implied user feedback derived from the click-through technique. Consequently; the proposed model implicitly maintains the user profile dynamically and keeps up-to-date based on the user profile preferences related to both queries.

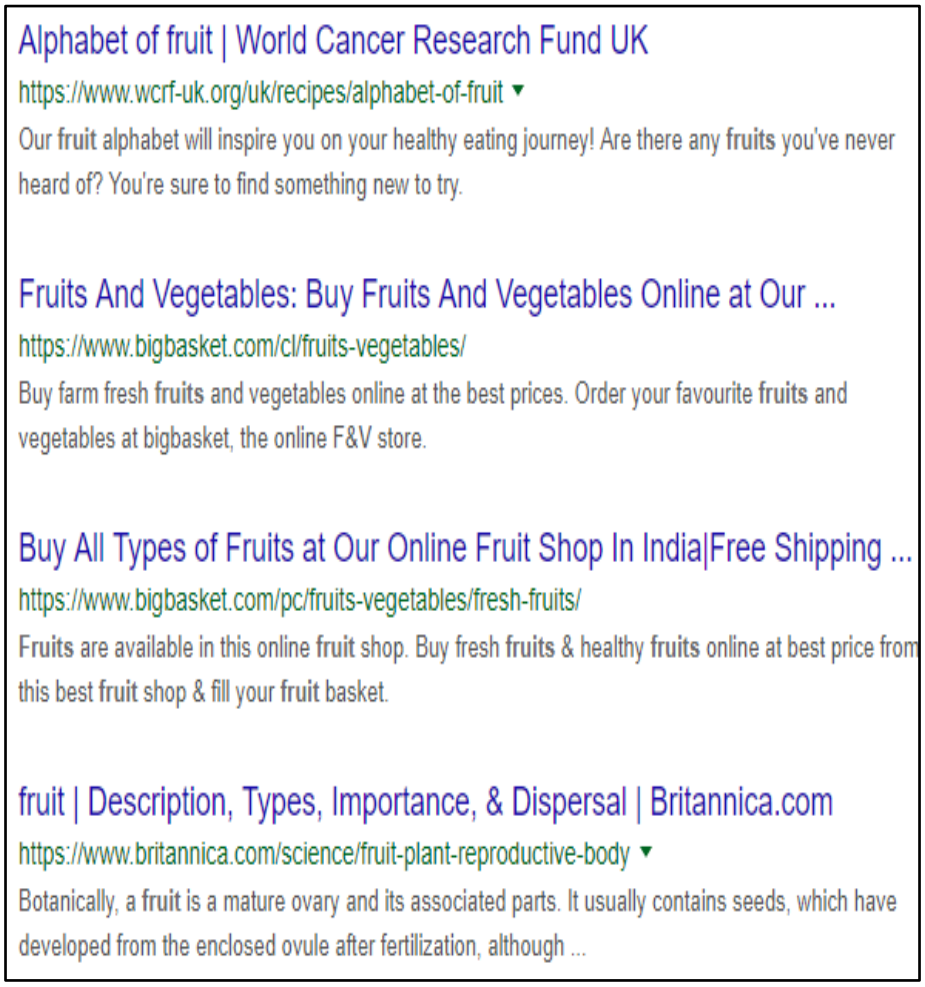

Figure 1:1: results of the search term "fruit"

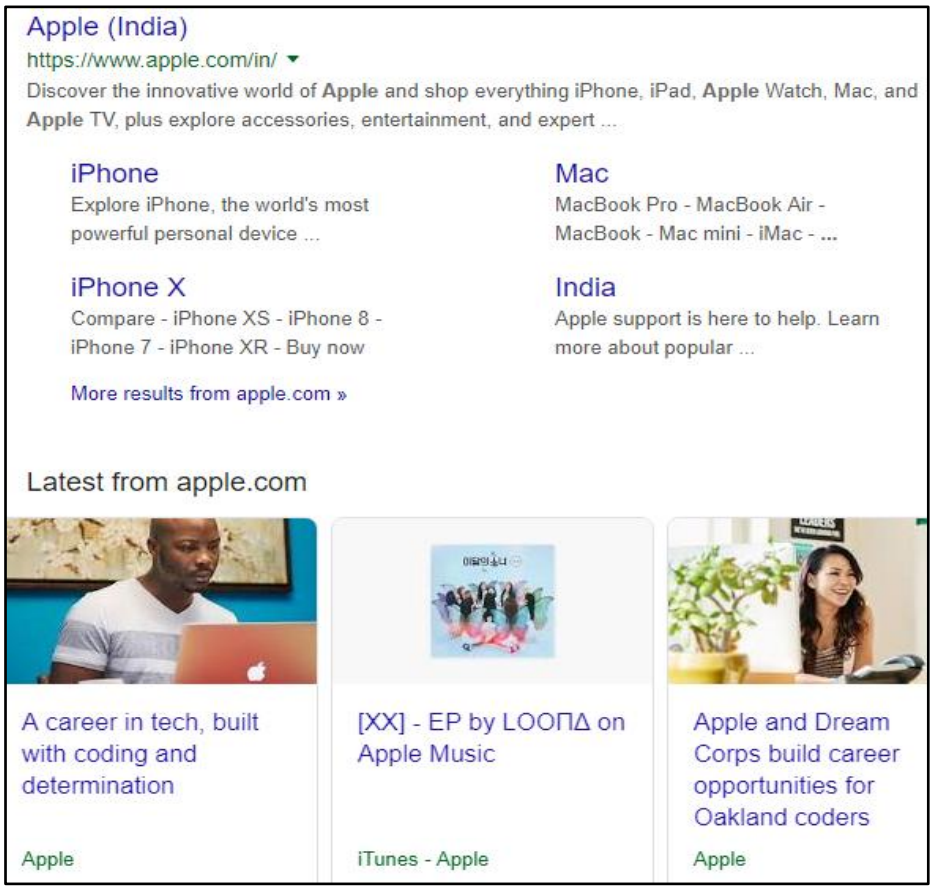

Figure 1.2: results of search term "apple"

2. Contribution

The main motivation behind our study is to retrieve personalized results based on query optimization to meet user's requirements. Our task is based on term-based user profile, which treats query optimization and eventually personalization of web search as a unified term rewriting. We describe a query rewriting algorithm for query optimization and customization based on this type of user profile. The figure 2 depicts the conceptual view of Web Search Personalizer (WSP).

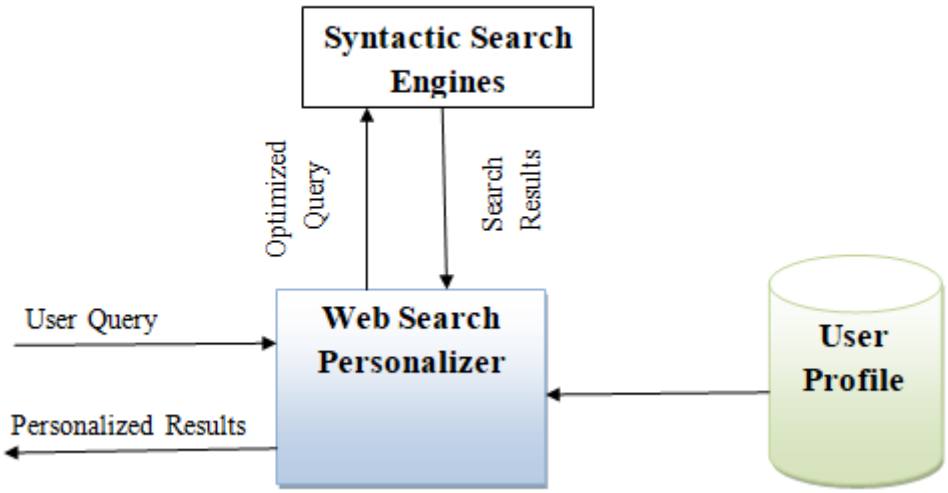

Figure 2: Conceptual view of Web Search Personalizer (WSP)

The user interacts with the WSP model by sending a search query as shown in Figure 2, which is then modified semantically to produce an optimized search query. The user query is configured according to the user profile suggested. The optimized search query is thereafter sent to syntactic search engines to obtain related search results, which will then be defused to produce the final customized results. Eventually, the WSP model implicitly gathers user feedback to refine the user profile, using the click-through technique. The remainder of this paper is structured as follows: Section 2 insights into related work. Section 3 explains the conceptual view of the proposed model, while Section 4 provides descriptions of the model architecture. Section 5 offers an in-depth case study and assessment of its simulation performance. Lastly, the work discussed in this paper is summarized in Section 6.

3. Overview of Related Work

Before developing our model, we surveyed existing work in this field and found that personalization methods can be divided into three categories: heuristic, feature-based and user-based. Heuristics based methods [28] use search logs to take some statistics decision such as number of clicks on a particular document in order to re- rank the documents already provided by the search engines. Feature based models extracts features used as input to machine learning algorithm to automatically lean personalization model. It is important to know that in feature based model, the same personalization model works for all the users. Finally, User- based model learn different model for different user as suggested by name. User- based models achieve highest personalization but require rich information about queries and clicked documents. Considering separate user profile for each user we opted to use the user- based model.

Despites it, different frameworks have proposed by researchers to personalize the search results. Some of these methodologies [10, 11, 12] are based on the users' geographical location. These geographical location based approaches retrieved results related to the user's language and the other attributes with no interest on the user's preferences. Other some methodologies rely on re- rank the web results retrieved from search engines [24, 25]. One of many disadvantages of these approaches is they rely on original search query without taking it into considerations. The most effective personalization approaches rely on designing a rich user profile [13, 14]. This paper [31] presents a framework for query search personalization for intranet search. The author in this paper created two user profiles for each session which

are clicked user profile and query user profile. For extracting topics from the user clicked documents, author uses LDA (Latent Dirichlet Allocation). For each input query, the proposed model is worked as follow:

1. Produce the n recommended search queries (q_s) by exploiting Adeyanju's method.
2. In second step, the researcher computed the similarity between q_s and clicked user profile and between q_s and query user profile in order to generate the personalized query suggestion feature.
3. After getting the suggestion features, to rerank the top n suggested queries, he employs LambdaMART to train ranking models.

This paper [29] shows privacy issues of a user while using personalized search engines. The author presents three kinds of software architecture for personalization: client side personalization, server side personalization and client server collaborative personalization. At the last the author concludes that the personalization at client side has advantages related to privacy issues over the personalization at server side.

The paper [30] presents a web personalization method in which the user query is directly matched to set of categories which represents the user intention based on the user profile and general profile. Several learning algorithms has been performed and evaluated and Rocchio-based Algorithm is considered to be the efficient to learn the user profile. Finally, the most robust and effective approaches build a rich user profile [15, 16]. This profile contains all user preferences like previous submitted queries, past clicked documents, time spend on a particular document, and user feedback. The main disadvantage of these approaches is either they ignore to focus on vocabulary problems or entail the user to maintain and enhance the user profile.

To handle the mentioned inconveniences, we suggest a personalization framework called "Web Search Personalizer" (WSP). The proposed framework exhibits a semantic-based method to optimize the user's query. In addition, the model updates the client profile through the user's implicit feedback.

4. Architecture of the proposed approach

In this portion, we sketch about various parts of our framework. Our "Web Search Personalizer" consists of three modules: Interface module, User Profiling module and Meta-Search module. These three modules interact together through well- defined interfaces to generate personalize results. The user interacts with interface module for submitting the query and to retrieve the personalized results. In order to optimize the user's original query, the interface module interacts with user profile to access the user preference and the WordNet to find synonyms. Since the optimized query is passed to the meta-search module, the meta-search module interacts with syntactic search engines (Bing, Google, and Yahoo) to retrieve and defuse the results. The personalized results are provided to the user through the interface module which senses the user's click to get implicitly feedback from the user. The interface module sends the user feedback to the user profile module to keep the user profile update. The overall working is also illustrated in the figure 3.

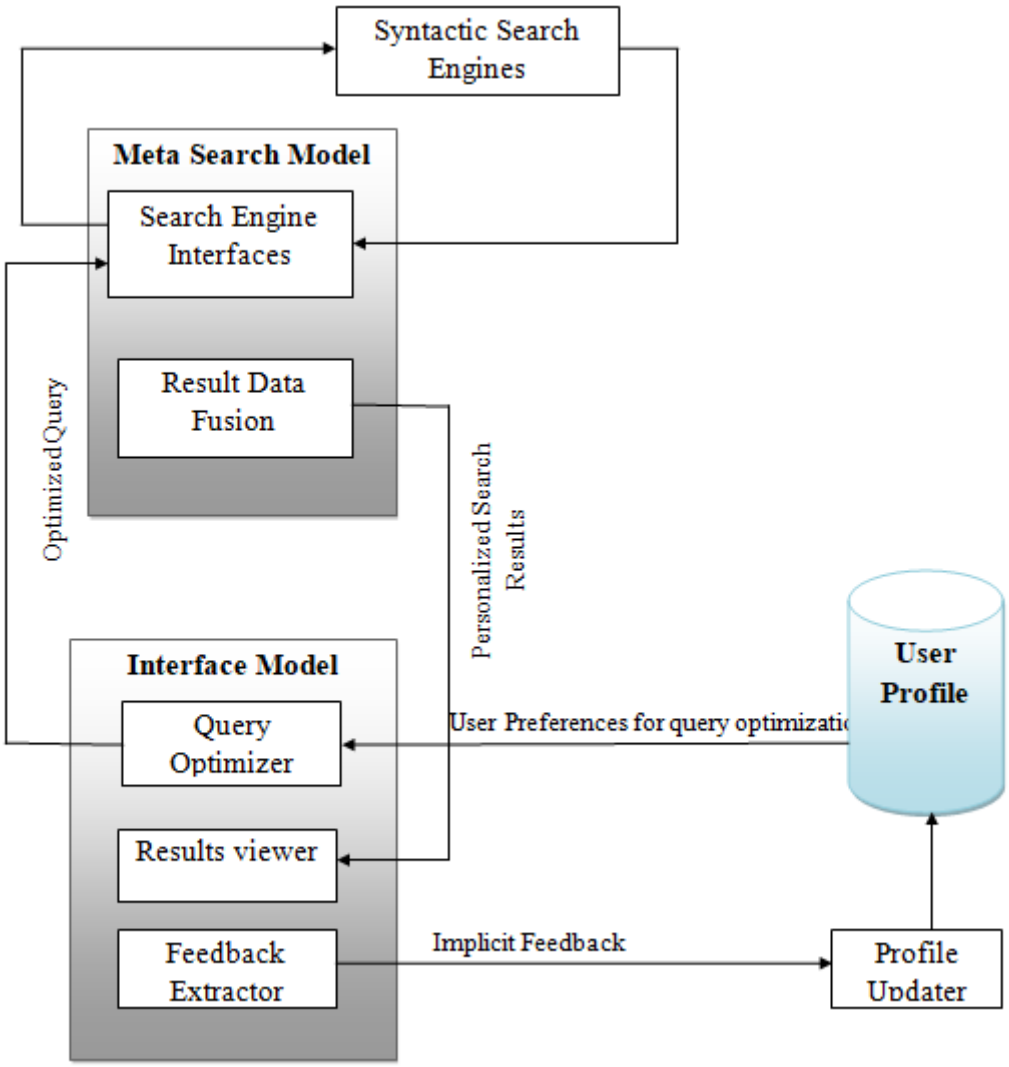

Figure 3: Web Search Personalizer

User Interface

It is liable for interacting with the user where user can submit query like "apple" or "java". It is also the responsibility of the user interface to show the results to the user and then to collect the implicit input. Therefore, there are three components to the user interface: Query Optimizer, Results Viewer and Input Extractor. To customize the user request the user interface communicates with the user profile. The query optimization algorithm is responsible for optimizing the original query based on user context.

Our objective of query optimization can be formalized as follows: For a user u who has a query Q = {t_1, t_2,.., t_m}, how to give a list of its related terms {t_{i1}, t_{i2}, ..., t_{ik}} for each term t_i **€ Q**, so that the gap between expectations of the user and system's offerings is minimized. The target is to modify Q into a query Q′ such that Q is essentially integrated in Q′, and the obtained results with Q′ be supposed to improve the precision of the results and doesn't reduce the user's satisfaction.

To achieve custom extensions of a query term t with a related term t_m, we follow two main tasks (i) the similarity between t and t_m, and (ii) the similarity between t_m and the user profile (a collection of user preferences) expressing the degree to which a tag t_m is likely to be of interest to the user concerned. We define a user profile as a weighted vector p_u = {w_{t1} ,w_{t2} ,

..., w_{tn}}, where w_{ti} is used to assess how important a term is to a user, i.e., similar to the tf-idf measure. We are finding these two required similarities as shown in the Equation 1. Once these similarities are computed using Jaccard similarity [20], a merge operation is required to obtain a final value. For this, Weighted Bords Fuse is exploited in the equation 1, where $0 \leq \alpha \leq 1$ is used to control the strength of social and semantic parts.

$$\boldsymbol{Rank\ t(t_m) = \alpha * Sim(t, t_m) + (1 - \alpha) * (\sum_{t_j \in p_u}^{n} Sim(t_m, t_j) * w_{t_j})/m} \quad (1)$$

Where Sim (t,t_m) is similarity between query term and t_m, m is the size of user profile, and w_{t_j} is the weight of term in user profile. Every term in the user profile is assigned different weight. The weight depends on the frequency of the term, and the term scanned from the title of the web page is more than the term of description and Meta term. The whole process of query optimization is shown in the algorithm 1.

Algorithm 1 Effective Query Optimization

Input: user query (Q).
Output: Optimized Query (Q').

Steps:
1: p_u [j] → extract preferences from user profile.
2: for each t_i € Q
3: l → GetWordNetDomain (t_i)
4: for each t_m € l do
5: t_m . value ← Rank t_m (t_j)
6: Sort l w.r.t to t_m. value and consider only top terms
7: Make logical OR (V) between t_i and all terms of l.
Update Q'
9: Return Q'

After getting the user preferences from the user profile (step 1) the task is to enrich the each term of the user query (step 2) with the user preferences. After that, the query optimizer retrieves the context terms of the query using the WordNet Ontology (step 3). In case we have multiple terms belong to l (step 4), we are selecting only top terms on the basis of the similarity score (step 5). Finally, the Query Optimizer joins t_i and its neighbors with the OR (V) logical connector (Line 7) and updated in Q' to generate the optimized query, which is then sent to the Meta-Search Agent. As an example, if a user submits a query Q = t_1 ^ t_2 ^ t_3 ... ^ t_m, it will be optimized to turn out to be Q′= (t_1 V t_{11} V ...V t_{1l}) ^ (t_2 V t_{21} V ... V t_{2k}) ^ ...^ (t_m V t_{m1} V ...V t_{mn}). The optimized query (Q') is then send to the meta search module.

User Profiling

Our personalization system is based on term based user profile. Every term of the document in the user profile is associated with a weight depending on the position of term in the document. We describe the method to extract the information from documents which was firstly proposed in [17]. We firstly fetch log entry from the query log. A log entry contains the user identity, submitted queries, top 10 retrieved results and clicked like with user's dwell time. We use the SAT criteria discussed in [18] to identify satisfied (SAT) clicks from the query logs.

A query log contains rich information about the user during interacting with the search engine. The most important parts of the query log are submitted queries and clicked documents for such queries.

For every clicked document, the user profiling agent extracts the main terms from the documents and assigns different weight to each extracted term based on the location of the term in the document. For example term in the heading tag has more weight than the term in the description tag.

Meta-Search Agent

Meta-Search Agent is a reactive agent [17] which responds to requests from interface agents. This functions as a meta-search engine, sending the tailored user search query to a few conventional search engines and then combining the results into a single list. This requires two elements: the search engine interface, and the subsequent data fusion modules. The component of the Search Engines Interface uses an Application Programmable Interfaces (APIs) to interact with the search engines in order to send the optimized query and receive the results. The Results Data Fusion then merges the search engine results into a single list. There is a ri(Qe) search results for each search engine, represented in a sequence of ri0, ri1, ri2, .. r1n. The sequence of search results from the different search engines are combined into a single Rm(Qe) sequence. The CombSum method is used in our model to merge the results of the search [19]. This method summarizes all the document and query similarity scores, and also normalizes the document similarity scores. This practice is finished when all crawled results are retrieved and defused to generate the search results' final single list.

5. Case Study

The project is implemented using java language. Jsoup libraries have been used to retrieve web links from third party search engines and to get the information from these web links. Jsoup is an open source library used to extract, parse and store information stored in HTML documents. The code used for retrieve link from Google as shown in the figure 4.

```
Document doc = Jsoup.connect(request).userAgent(
    "Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)")
  .timeout(5000).get();
  // get all links
 Elements links = doc.select("a[href]");
for (Element link : links) {
  String temp = link.attr("href");
  if(temp.startsWith("/url?q=")){
   result.add(getDomainName(temp));
  }
  }
 } catch (IOException e) {
 e.printStackTrace();
}
 return result;
 }
```

:

Figure 2: Jsoup code for retrieving results from the third party

For each retrieved link using Jsoup library, the title and Meta description is extracted as shown in the figure 5.

```
Document doc = Jsoup.connect("http://www.javatpoint.com").get();
String title = doc.title();
String keywords = doc.select("meta[name=keywords]").first().attr("content");
```

Figure 5: Jsoup code for getting data from the retrieved web links

6. Discussion and Evaluation of Experimental Results

In our experiments, we hired two users A and B. The user A and B has their own profile which contains their preferences. Every client is given a different computer. For every query, the user finds the suitable pages. We compare our WSP model with Google, Bing and Yahoo. The original query is submitted to these search engines to fetch their results. On the other hand, our WSP model firstly reformulates the original query and then passed the optimized query to these same search engines. The statistic of the top 30 retrieved results is shown in the table 1.

Users generally submit the ambiguous queries to the search engine while using search engine. The search engine still shows the results to the user but failed to understand the real intention of the user's quest. On the other hand, personalization of the web search offers the exact results as expected by the users even the user appears to formulate the short and vague query. After considering the short and ambiguous queries we ran our experiments particularly. Our evaluation results proves that personalization of original query based on a keyword based user profile is an effective approach. The precision is calculated to measure the effectiveness of our proposed approach in this paper. Precision is a fraction of appropriate fetched or retrieved documents. The Recall that is also used in many cases is the fraction of the relevant documents which have been fetched.

$$\text{Precision} = TP / (TP + FP) \quad (2)$$

i.e., (number of correctly fetched documents) / (Total number of documents retrieved)

$$\text{Recall} = TP / (TP + FN) \quad (3)$$

i.e., (Number of correctly fetched pages) / (Total number of relevant documents retrieved):

In our case the precision is computed as the fraction of provided documents for query that consent with preferences obtained from the human. We fetched the results from various platforms: Google, Yahoo and Bing.

Table 1: Results of WSP Model and other Search engines

System	No of Relevant Results	Number of irrelevant results	Precision
Google			
User A	25	5	0.83
User B	4	26	0.13
Yahoo			
User A	24	6	0.8
User B	6	24	0.2

Bing			
User A	21	9	0.7
User B	8	22	0.26
WSP Model			
User A	29	1	0.96
User B	28	2	0.93

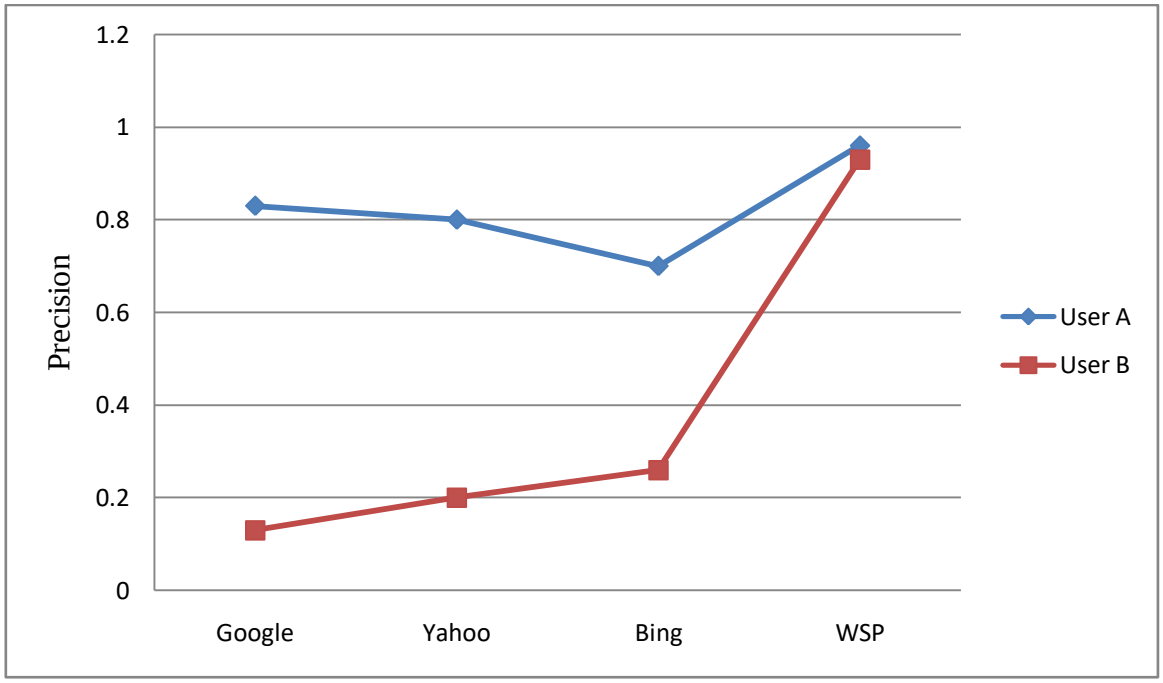

Figure 6: Comparison of WSP model and other search engines

The Figure 6 shows the precision of different search engines for both users A and B. As we can notice, the precision for both users has been increased to 0.96 and 0.93 respectively when the same query is submitted to WSP model with different profiles.

7. Conclusion

In this paper we presented a model for personalizing web search based on the optimization of queries. The model dynamically builds and manages the user profile, keeping it up to date. The original query is configured semantically during the web search process, using the user preferences and the ontology of WordNet. Also the final results of WSP model are evaluated with Google, Yahoo and Bing to prove how the precision is increased with the proposed model.

References

1. Jiang, J. Y., Liu, J., Lin, C. Y., & Cheng, P. J. (2015, October). Improving ranking consistency for web search by leveraging a knowledge base and search logs. In Proceedings of the 24th ACM International on Conference on Information and Knowledge Management (pp. 1441-1450). ACM.
2. D. Beeferman and A. Berger. Agglomerative clustering of a search engine query log. In Proceedings of KDD, pages 407-416, 2000.
3. R. Jones, B. Rey and O. Madani. Generating Query Substitutions. In Proceedings of WWW, pages 387-396, 2006.

4. X. Wang and C. Zhai. Mining term association patterns from search logs for effective query reformulation. In Proceedings of CIKM, pages 479-488, 2008.
5. J. Wen, J. Nie and H. Zhang. Clustering user queries of a search engine. In Proceedings of WWW, pages 162-168, 2001.
6. Cai, Fei, Shuaiqiang Wang, and Maarten de Rijke. "Behavior-based personalization in web search." Journal of the Association for Information Science and Technology 68.4 (2017): 855-868.
7. Khodaei, A., Sohangir, S., & Shahabi, C. (2015). Personalization of web search using social signals. In Recommendation and Search in Social Networks (pp. 139-163). Springer, Cham.
8. Ho, S. Y., & Bodoff, D. (2014). The effects of Web personalization on user attitude and behavior: An integration of the elaboration likelihood model and consumer search theory. MIS quarterly, 38(2).
9. Eickhoff, C., Collins-Thompson, K., Bennett, P. N., & Dumais, S. (2013, February). Personalizing atypical web search sessions. In Proceedings of the sixth ACM international conference on Web search and data mining (pp. 285-294). ACM.
10. Bennett, P. N., Radlinski, F., White, R. W., & Yilmaz, E. (2011, July). Inferring and using location metadata to personalize web search. In Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval (pp. 135-144). ACM.
11. Majid, A., Chen, L., Chen, G., Mirza, H. T., Hussain, I., & Woodward, J. (2013). A context-aware personalized travel recommendation system based on geotagged social media data mining. International Journal of Geographical Information Science, 27(4), 662-684.
12. Kliman-Silver, C., Hannak, A., Lazer, D., Wilson, C., & Mislove, A. (2015, October). Location, location, location: The impact of geolocation on web search personalization. In Proceedings of the 2015 Internet Measurement Conference (pp. 121-127). ACM.
13. Ibrahim, N., Chaibi, A. H., & Ghézala, H. B. (2017). Scientometric re-ranking approach to improve search results. Procedia Computer Science, 112, 447-456.
14. Zamir, O. E., Korn, J. L., Fikes, A. B., & Lawrence, S. R. (2010). U.S. Patent No. 7,693,827. Washington, DC: U.S. Patent and Trademark Office.
15. Vallet, D., Cantador, I., & Jose, J. M. (2010, March). Personalizing web search with folksonomy-based user and document profiles. In European conference on information retrieval (pp. 420-431). Springer, Berlin, Heidelberg.
16. Matthijs, N., & Radlinski, F. (2011, February). Personalizing web search using long term browsing history. In Proceedings of the fourth ACM international conference on Web search and data mining (pp. 25-34). ACM.
17. Labrou Y, Finin T, Peng Y. Agent communication languages: the current landscape. J IEEE Intell Syst 1999;14(2):45–52.
18. Carman MJ, Crestani F, Harvey M, Baillie M. Towards query log based personalization using topic models. In: Proceedings of the 19th ACM conference on information and knowledge management (CIKM) 2010 Toronto, Ontario, Canada; 2010. p. 1849–52.
19. Lee JH. Combining multiple evidence from different properties of weighting schemes. Interest group on information retrieval (SIGIR). In: Proceedings of the 18th annual international ACM SIGIR conference on research and development in information retrieval, Seattle, Washington, USA; 1995. p. 180–8.
20. Niwattanakul, S., Singthongchai, J., Naenudorn, E., & Wanapu, S. (2013, March). Using of Jaccard coefficient for keywords similarity. In Proceedings of the

international multiconference of engineers and computer scientists (Vol. 1, No. 6, pp. 380-384).
21. Mauro, N., & Ardissono, L. (2018, March). Session-based Suggestion of Topics for Geographic Exploratory Search. In 23rd International Conference on Intelligent User Interfaces (pp. 341-352). ACM.
22. Vaynblat, D., & Domain, M. (2017). U.S. Patent No. 9,672,538. Washington, DC: U.S. Patent and Trademark Office.
23. Leung, K. W. T., Lee, D. L., & Lee, W. C. (2010, March). Personalized web search with location preferences. In 2010 IEEE 26th International Conference on Data Engineering (ICDE 2010) (pp. 701-712). IEEE.
24. Kashyap, A., Amini, R., & Hristidis, V. (2012, October). SonetRank: leveraging social networks to personalize search. In Proceedings of the 21st ACM international conference on Information and knowledge management (pp. 2045-2049). ACM.
25. Bennett, P. N., White, R. W., Chu, W., Dumais, S. T., Bailey, P., Borisyuk, F., & Cui, X. (2012, August). Modeling the impact of short-and long-term behavior on search personalization. In Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval (pp. 185-194). ACM.
26. Alaoui, S., Idrissi, Y. E. B. E., & Ajhoun, R. (2015). Building rich user profile based on intentional perspective. Procedia Computer Science, 73, 342-349.
27. Sharma, S., & Rana, V. (2020). Web Search Personalization Using Semantic Similarity Measure. In Proceedings of ICRIC 2019 (pp. 273-288). Springer, Cham.
28. Hwang, G. J., Kuo, F. R., Yin, P. Y., & Chuang, K. H. (2010). A heuristic algorithm for planning personalized learning paths for context-aware ubiquitous learning. Computers & Education, 54(2), 404-415.
29. Shou, L., Bai, H., Chen, K., & Chen, G. (2012). Supporting privacy protection in personalized web search. IEEE transactions on knowledge and data engineering, 26(2), 453-467.
30. Liu, F., Yu, C., & Meng, W. (2002, November). Personalized web search by mapping user queries to categories. In Proceedings of the eleventh international conference on Information and knowledge management (pp. 558-565). ACM.
31. Vu, T., Willis, A., Kruschwitz, U., & Song, D. (2017, March). Personalised query suggestion for intranet search with temporal user profiling. In Proceedings of the 2017 Conference on Conference Human Information Interaction and Retrieval (pp. 265-268).