

## Personalization of the Web Search

Sandesh Pare<sup>1</sup>, Bharati Vasgi<sup>2</sup>

<sup>1</sup>Department of Information Technology, Sinhgad College of Engineering, Pune, India

<sup>2</sup> Department of Information Technologies, Sinhgad College of Engineering, University of Pune, India

**Abstract:** Web search engines help users find useful information on the WWW. However, when the same query is submitted by different users, typical search engines return the same result regardless of who submitted the query. Generally, each user has different information needs for his/her query. Therefore, the search results should be adapted to users with different information needs. So, there is need of several approaches to adapting search results according to each user's need for relevant information without any user effort. Such search systems that adapt to each user's preferences can be achieved by constructing user profiles based on modified collaborative filtering with detailed analysis of user's browsing history.

There are three possible types of web search system which can provide personalized information: (1) systems using relevance feedback, (2) systems in which users register their interest, and (3) systems that recommend information based on user's history. In first technique, users have to provide feedback on relevant or irrelevant judgments which is time consuming and the second one needs registration of users with their static interests which need extra effort from user. So, the third technique is best in which users don't have to give explicit rating; relevancy automatically tracked by user behavior with search results and history of data usage. It doesn't require registration of interests; it captures changing interests of user dynamically by itself. The result section shows that user's browsing history allows each user to perform more fine-grained search by capturing changes of each user's preferences without any user effort. Users need less time to find the relevant snippet in personalized search results compared to original results.

**Keywords:** Browsing History, Collaboration Filtering, Extraction, Re-Ranking, Scoring, User Profile, User Query

### I. INTRODUCTION

Web personalization reduces the burden of information overload by tailoring the information presented based on an individual user's needs. Every user has a specific goal when searching for information through entering keyword queries into a search engine e. g. a historian may enter the query Madonna and child while browsing Web pages about art history, while a music fan may issue the same query to look for updates on the famous pop star. In recent years, personalized search has attracted interest in the research community as a means to decrease search ambiguity and return results that are more likely to be interesting to a particular user and thus providing more effective and efficient information access. One of the key factors for accurate personalized information access is user context.

Users use Web search engines to find useful information on the World Wide Web. However, when the different users submitted the same query, current working search engines like google, yahoo, etc. return the same result regardless of who submitted the query. Generally, each user has different intentions or information needs for his / her query. Therefore, the search results should be adapted to users with different information needs. So, there is need of several approaches to adapting search results according to each user's need for relevant information without any user effort. Such search systems that can record each user's preferences can be achieved by constructing user profiles based on modified collaborative filtering with detailed analysis of user's browsing history.

Researchers have long been interested in the role of context in a variety of fields including artificial intelligence, context-aware applications, and information retrieval. While there are many factors that may contribute to the delineation of the user context, here three essential elements are considered that collectively play a critical role in personalized Web information access. These three independent but related elements are the user's short-term information need, such as a query or local context of current activity, semantic knowledge about the domain, and the user's profile that captures long-term interests. Each of these elements is considered

critical source of contextual evidence, a piece of knowledge for disambiguation of the user's context for information access.

Another novel approach is introduced for building ontological user profiles by assigning interest scores to existing concepts in domain. These profiles are maintained and updated as annotated specializations of pre-existing reference domain ontology. A spreading activation algorithm used for maintaining the interest scores in the user profile based on the user's ongoing behaviour. Re-ranking is done of the search results based on the interest scores and the semantic evidence in an ontological user profile successfully provides the user with a personalized view of the search results by bringing results closer to the top when they are most relevant to user. Allan et al. in [4] define the problem of contextual retrieval as follows: "Combine search technologies and knowledge about query and user context into a single framework in order to provide the most appropriate answer for a user's information needs."

Effective personalization of information access involves two important challenges: accurately identifying the user context and organizing the information in such a way that matches the particular context. Since the acquisition of user interests and preferences is an essential element in identifying the user context, most personalized search systems employ a user modeling component. Users often start browsing through pages that are returned by less precise queries which are comparatively easy to keep track and construct user interest model. Since the users are unaware to specify their underlying intent and search goals, personalization must pursue techniques that capture implicit information about the user's interests. This Personalized Search builds a user profile by means of implicit feedback where the system adapts the results according to the search history of the user. Many systems employ search personalization on the client-side by re-ranking documents that are suggested by an external search engine such as Google, Yahoo! Since the analysis of the pages in the result list is a time consuming process, these systems often take into account only the top ranked results. Also, only the links associated with each page in the search results is considered as opposed to the entire page content.

## **II. RELATED WORK**

Now-a-days, technology is developing rapidly and information floods. In the information explosion era, people don't care about the scale of information but the technique to obtain the needed information quickly and accurately. So, the personalized searching system is emerged to provide most personal relevant searching results. And the key problem is to make clear the needs of the users. So, the researchers combined the concepts of user interest and collaborative filtering to reorder the search results and introduced the multi-agent technology in [3].

Personalizing web search results has long been recognized as a concept to greatly improve the search experience. A personalization approach is presented which builds a user interest profile using user's complete browsing behaviour, and then uses this model to re-rank web results. Using a combination of content and previously visited websites provides effective personalization. A number of techniques are proposed for filtering previously viewed content that greatly improve the user model used for personalization.

Every user has a distinct background and a specific goal when searching for information on the Web. The goal of Web search personalization is to tailor search results to a particular user based on that user's interests and preferences. Effective personalization of information access involves two important challenges: accurately identifying the user context and organizing the information in such a way that matches the particular context. There are three possible types of Web search systems which can provide personalized information: (1) systems using relevance feedback, (2) systems in which users register their interest or demographic information, and (3) systems that recommend information based on user's browsing history [2]. In first technique, users have to register personal information such as their name, e-mail id, and so on, beforehand, and users have to provide feedback on relevant or irrelevant judgments. The discovery of patterns from usage data by itself is not sufficient for performing the personalization tasks. Other systems designed to realize such adaptive systems have been proposed in [5, 6] that personalize information or provide more relevant information for users. According to second technique, user has to give their interests and its ratings on a scale from bad to good. This type can become time consuming and users prefer easier methods. So, the third technology is better than others. In this, User's browsing history allows each user to perform more fine-grained search by capturing changes of each user's preferences without any user effort.

Although personalized search has been under way for many years and many personalization algorithms have been investigated, it is still unclear whether personalization is consistently effective on different queries for different users and under different search contexts. A large-scale evaluation framework is presented in [1] for personalized search based on query logs and then evaluated personalized search results

using query logs of live Search. By analyzing the results, it is revealed that personalized Web search does not work equally well under various situations. It represents a significant improvement over generic Web search for some queries, while it has little effect and even harms query performance under some situations. Click entropy proposed in [8] is a simple measurement on whether a query should be personalized. Several features also proposed to automatically predict when a query will get benefit from a specific personalization [9, 10]. Experimental results show that using a personalization algorithm for queries selected by prediction model is better than using it simply for all queries. So, it is conclude that personalization gives best result but not all the time. Its overall performance is totally dependent on taking the right decision of when personalization should occur.

### III. METHODOLOGY

The search engine is responsible to provide the best search results to every query submitted by the user. The different search engines use different methods to extract the search results for user submitted query. The search engines use number of various techniques to represent the search results to user. Based on the methodologies used by the system, they are differently categorized. Both the existing methodology and proposed methodologies are further elaborated in next section.

#### 3.1 Existing Methodology

In this section, the working of the current search engine is explained. Whenever the user submits a query to currently working search engine, it crawls on WWW. The query is in form of the keyword. While crawling on WWW, search engine selects some of the documents or websites as a relevant, and they are presented to the user in the form of snippets in search result. The process of selecting the document or website as a relevant is totally dependent on matching to query. Some of the search engines select the website as a relevant which contains query (or keyword in a query) in title tag or in meta-name tag, etc. And some of the search engines select the document as a relevant document which has no. of occurrences of query (or keyword in a query) in it. Based on the position of user query on the website or no. of its occurrences in the document, its ranking in search result is finalized. Besides, there are some financial issues to keep the website or the document on the high ranking in the search result. The specific user interests or preferences are not taken into consideration i.e. the same search results are provided to the every user on a same query. This can degrade the quality of search result from user’s point of view.

#### 3.2 Proposed Methodology

In this section, the framework of these systems reviewed with regard to “Personalization”. Links, structure and contents of Web pages are often used in the construction of a personalized Web site. This scheme involves selecting the links that are more relevant to the user for the different queries. Most of the applications use link personalization to recommend results based on the buying history of clients or some categorization of clients based on ratings and opinions. Users who give similar ratings to similar documents are presumed to have similar preferences, so when a user seeks recommendations about a certain query, the search engine suggests those recommendations that are most popular for his/her class or those that best correlate with the given query for that class. At the E-commerce site, this approach has been taken to an extreme by constructing a “New for you” home page and presenting it to each user, with new products that the user may be interested in. Additionally, E-commerce sites uses implicit recommendations via purchase history or explicit recommendations via “rate it” features to generate recommendations of products to purchase. This system automatically adapts links in the browsed pages and their relevance to the weighted topic is specified by users.

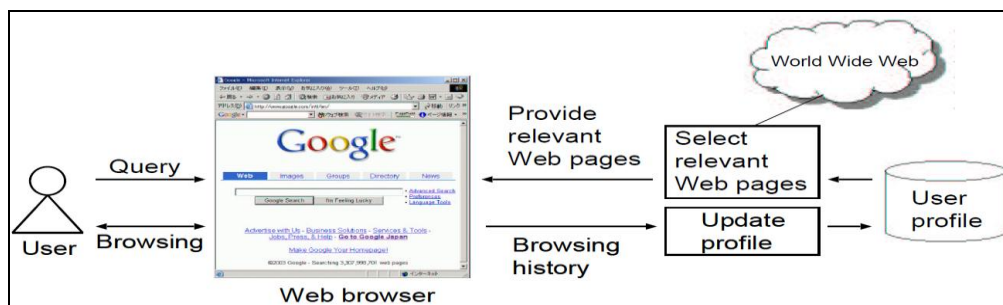


Fig. 3.1 Overview of the Personalization System

**3.2.1 User Interest model**

**a. Description of User Interest model:**

User interest model is the formalized description of the user's interest information. There are typically three kinds of models that are static model, implicit dynamic model and explicit dynamic model. Here, the weighted keyword vector model is adopted, one of the explicit dynamic models. The weighted keyword vector model is described as following:

$$Interest_i = \{(k_1, w_1), \dots, (k_n, w_n)\}$$

Where, Interest represents the interest model of user  $U_i$ .  $k_i$  is the  $i$ -th keyword which can be both extracted from the user's logs, queries and typed in by the user in advance and  $W_i$  is the weight of keyword  $k_i$  which represents how interested the user is in  $k_i$ . The weight is also called interest value.

**b. Update of User Interest model:**

This approach will update the user interest model dynamically. When a user  $U_i$  send a query  $k_j$ , it will first find out whether the keyword  $k_j$  is in his/her interest model. If the item  $(k_j, W_j)$  is in  $Interest_j$ , a unit score is added to  $W_j$ . Otherwise a new item  $(k_j, w_j)$  will be added into  $Interest_j$  where  $W_j$  is the default value. For any user  $U_i$ , the interest value  $W_j$  of  $Interest_i$  will decrease according to the Ebbinghaus Curve. Assume that  $W_j_{pre}$  is the interest value before decrease and  $wj_{new}$  is the interest value after decrease.

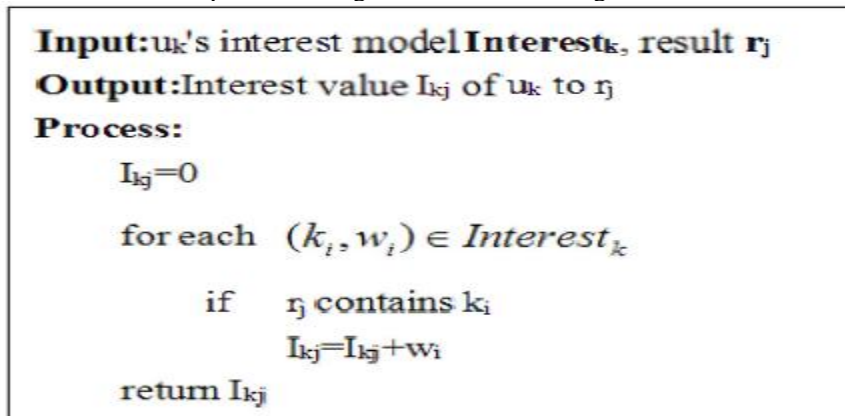
$$W_{j\_new} = W_{j\_pre} \times \lambda$$

$$\lambda = e^{\log_2 \frac{(t-t_0)}{30}}$$

Where,  $\lambda$  is the attenuate coefficient,  $t$  is the current time and  $t_0$  is the time when interest value was last updated. The item  $(k_j, W_j)$  will be removed from  $Interest_i$  if  $W_{j\_new}$  is less than the threshold.

**c. Compute the interest value:**

According to the user's query, this system will get back some search results. Then for each result  $r_i$ , the user's interest value to it is computed. The algorithm is shown as figure 3.2.



**Fig.3.2. Computation of User Interest Value for search results**

**3.3 How Personalization System Works?**

In this section, the detailed explanation about exact working of personalization system is given. Fig 3.3 is the diagrammatic representation of working of personalization system which is given below.

As shown in the fig 3.3, the users of personalized web search have to first register on the system. Users have to provide all his personal information while registration. When user registers to the system, a unique user id is assigned to that user. And, entry of that user along with his / her personal information is made to the user profile. Again, users have to sign in to the system while searching because unless user does not log in the system, system could not correctly track the person who using the system. Therefore, it is unable to present the personalized results to the user query. After Sign in process, the user can input a query to the system. The query is in the form of keyword. After query is submitted, google API comes into the picture. The

role of google API is to give the same user query to the google search engine and extract the results for that query from WWW. This phase is known as the extraction phase.

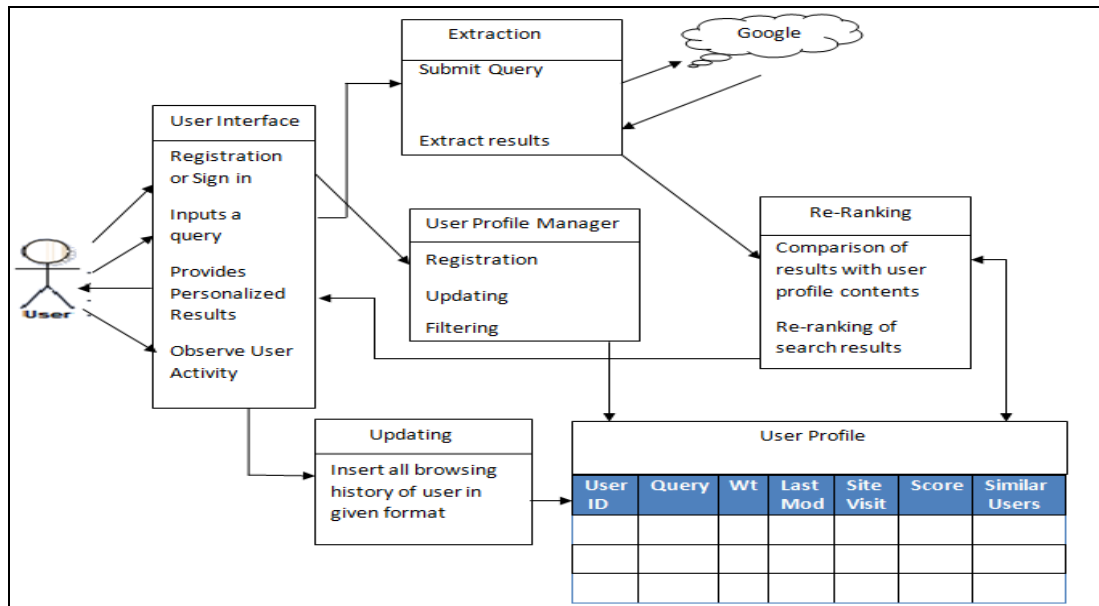


Fig 3.3 Working of the Personalization System

After extraction of search results, they are compared with the user profile contents. At the very first time, there are no logs about the user. So, google results are provided to the user as it is. But, if any user is similar to him / her, then preferences of similar user kept at top most rank in search result. After some visits, if own user profile contains certain logs, then it will compare the google results with the user profile contents. Then the re-ranking of search result is done by keeping all previously visited sites at the highest rank. If more than one snippet in google search result is present in the user profile, then the scoring of those snippets is used for re-ranking. The snippet with high score kept at the highest rank. Then it checks the entries in the user profile for similar users. If some entries found in the log of similar user, that are kept just below the highest ranking results. Weight is the special term assigned to each entry of user profile. Both the terms i.e. duration from last accessed and the no. of visits affect on weightage of entry.

The re-ranked result is then presented to the user. User then visit to particular snippet which he thinks useful. System keeps track of the user interaction with the provided result. According to the user behaviour of user, new entries are made to the user profile or existing entries are updated. So that can be useful for next searches. User profile manager is the model which performs this duty. Instead of this, it also performs filtering of user profile i.e. the entries which have weightage below than threshold value are opted out.

#### IV. PROJECT ARCHITECTURE

##### 4.1 Flow of the Project

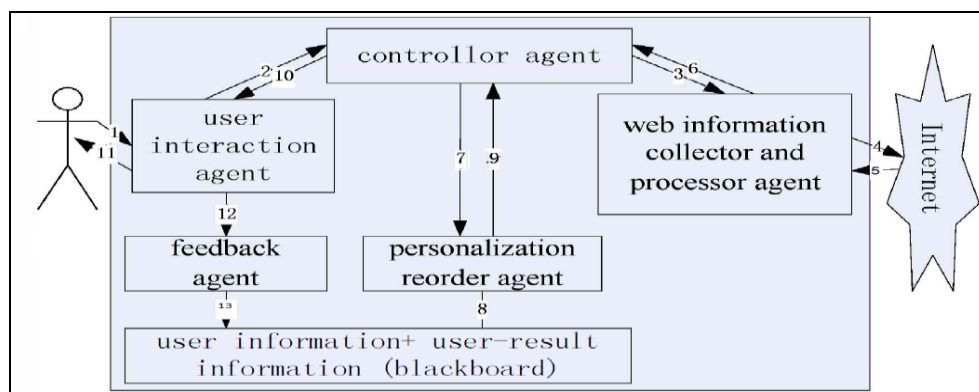


Fig 4.1 Flow of the Project



The fig 4.1 shows the flow of the project. The flow represents the sequence in which overall activities are performed. Flow diagram plays an important role in understanding the working of the system.

## 4.2 Project Architecture

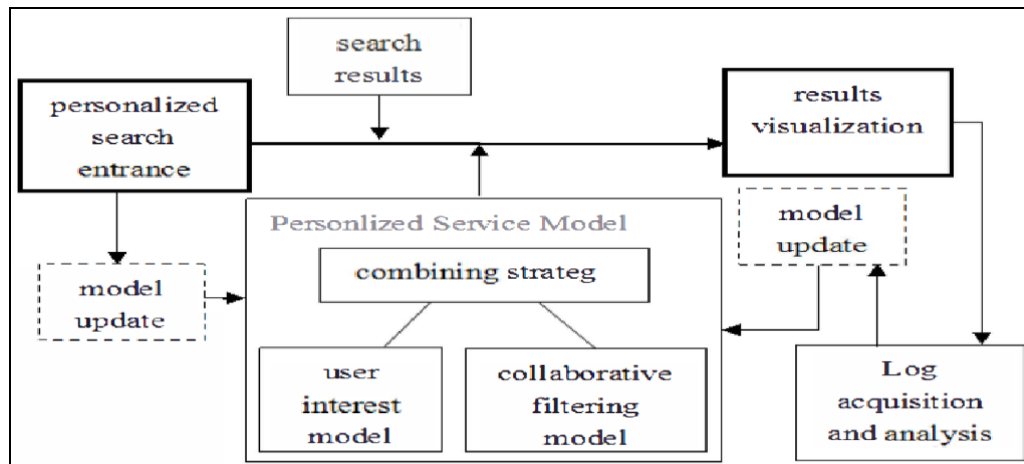


Fig 4.2 Model Diagram of the Personalization System

Firstly, a browsing history of user is captured and basic visual features such as title, metadata description, etc. are extracted from each history. And then, contextual features which are derived from basic features with time expansion are utilized to predict the probabilities of user behaviour. At last, a post-processing method is used to refine the result. The overall framework of the method is shown in Fig 4.2.

### 4.2.1 Personalization Strategies

In this section, the approach is described. The first step consists of constructing a user profile that is then used in a second phase to re-rank search results.

#### 4.2.1.1 User Profile Generation

A user is represented by a list of terms and weights associated with those terms, a list of visited URLs and the number of visits to each, and a list of past search queries and pages clicked for these search queries. This profile is generated as shown in Figure 4.2. First, a user's browsing history is collected and stored as (Query, URL) pairs. Next, this browsing history is processed into no. of different summaries consisting of term lists. Finally, the term weights are generated using different weighting algorithms. Below, each of these steps is explained in detail.

#### a. Data Capture:

Users have to first login to the system. Until user doesn't login to the system, it can't correctly track the user who using that system. So, it is unable to provide personalized results to the user. But, it is not compulsory to login. After that, users have to input a query to the system. This query is in the form of keyword. This query is then passed to the google API. It is the application which runs as a google search engine server. The role of google API is to accept a query from the user and search for relevant documents on the internet. The most relevant documents are selected as a result and provided to the user. The result is in form of snippets. A result page generally contains eight to ten snippets on it. Each snippet is presented in the form of its title, URL of that site and the description of its content. This phase of system is known as the data capture phase. Because, this phase capture relevant data to the query from the internet.

#### b. Data Extraction:

Once the data capture phase is over, the result is presented to the user. The users have to interact with the results. Users visit to the particular URL from the result whichever he / she thinks most appropriate to them. Then, the task of this system is to again extract the information about the snippet. So, this phase is known as the extraction phase of the system. The following summaries of the content viewed by users are considered in building the user profile:

**Title Unigrams:** The words inside any <title> tag on the html pages.

**Metadata Description Unigrams:** The content inside any <meta name=\description"> tag.

#### c. Term List Filtering:

To reduce the number of noisy terms in our user representation, we also tried filtering terms by removing infrequent keywords. Each term is assigned some weightage to it, which is explained below in this report. This weightage is the representation of no. of visits and the duration from last accessed date. Based on the value of weightage, filtering of terms is performed. In this phase, once the keyword with weight below than threshold value is found, it is opted out from the user profile contents.

#### d. Term Weighting:

After the list of terms has been obtained, the weights for each term are computed. This weight plays important role in the personalization process. Based on its value, re-ranking of the search result is performed. There are different techniques of weight assignment. Here, weight is assigned basically in two ways. They are explained below:

$$W_{j\_new} = W_{j\_pre} \times \lambda$$

$$\lambda = e^{\log_2 \frac{(t-t_0)}{30}}$$

**Fig 4.3 Weight Assignment Formula**

- i) TF-IDF Weighting: The user query is in form of sequence of keywords. Once the query is submitted from the user, this user is query is spitted in the terms. Each term in this query has its own weightage. This project uses specific formula for weight assignment, which is shown in the fig 4.3. The formula calculates  $W_{j\_new}$  value for the term. It contains  $W_{j\_pre}$  which is previous value of weight assigned to that term. Here,  $t$  denotes current date and  $t_0$  denote last modified date.  $e$  is the constant with approximate value 2.17.
- ii) TF-Weighting: This weighting scheme is used for assigning the weight to the pages visited. Here, a simple counter is maintained for weight assignment. So, each time the same page visited by the user, the weight to particular page gets incremented.

#### 4.2.1.2. Re-ranking Strategies:

Default search engine provides the results for user query. It ranks the snippets in results based on its own technique. It ranks the results based on relevancy. But, based on the user needs, personalization system re-ranks the search results and presented to the user. This re-ranking typically includes following methods:

##### a. Scoring Methods:

According to the formula discussed earlier, the different terms in the user profile gets assigned their weight. The different HTML pages visited by the user have weight just equal to no. of access to it. The scoring method is responsible to assign these weights. The rating is also scored more if the relation between two user is high i.e. more accessed keywords match between them.

##### b. Rank and Visit Scoring:

Based on the score of the terms or the pages visited by the user, re-ranking of search result is done. The results which have entry in the user profile get the top rank in the search result. From them, pages with high weight value get the top most rank in the search result i.e. Re-ranking method is responsible to the re-ranking of search result based on the weight assigned to them. This method also checks the logs of similar users. If any entries found in their record, that snippets conceived as a second most high ranking result.

## V. RESULTS & DISCUSSION

### 5.1 Introduction

There was more than 8 users take part in the experiment. Each user sends out some queries. For each query, they obtained 10 result items from the internet. Actually, they were provided by Google Web Search API. Then the search results were reordered and presented to the user. The user evaluated the result items and visits to favourite snippet according to their own judgment. Then, the 'time' required by each user to search exact favourite snippet in both the search result i.e. personalized sequence & original sequence are recorded. According to this 'time', the effectiveness of the personalized sequence & original sequence are evaluated, which are obtained from Google API.

The 'time' value is computed again and again. The minimum 'time' value is the best 'time' value. It represents the sorting effect of the top elements in the personalized sequence compared to the original sequence. All the 'time' values are computed in milli-seconds (mSec). The sequence in which user required less 'time' to find his / her favourite snippet is considered as the best sequence.

For some queries, proposed approach can obtain better reordering effect than the original order provided by Google. Here, "google" is taken as an example. In table 5.1, 8 users are chosen for experimental results and the avg. row is the average 'time' value of the corresponding column. The *timePR* & *timeOR* column shows the 'time' required by the user to search exact favourite snippet in the personalized & original search result, respectively.

For each query, a table can be represented like Table 5.1. By combining the Avg. row in the tables corresponding to each query (like Table 5.1), Table 5.2 can be generated. Table 5.2 is the summary of the experimental results (the results of 10 typical queries). The avg. row is the average value of the corresponding column. The other values are all obtained from the Avg. row and corresponding column of the tables corresponding to the queries. Finally, the improving is calculated by comparing the *timePR* and *timeOR* column to evaluate the effect of this approach.

**Table 5.1 Experimental result of query "google"**

User	Current Expected Result	<i>timePR</i> (mSec)	<i>timeOR</i> (mSec)
1	earth.google.com	187	270
2	<a href="http://www.google.co.in">www.google.co.in</a>	238	184
3	images.google.co.in	221	210
4	<a href="http://www.google.co.in">www.google.co.in</a>	199	228
5	news.google.co.in	361	396
6	mail.google.com	152	214
7	translate.google.co.in	254	327
8	earth.google.com	270	322
Avg.	--	235.25	268.875

According to the table 5.2, the personalization system gives the good results to the queries like "Computer" & "Jobs" i.e. User require less time to find the personal relevant snippet in search result. This is happened because this system keeps track of user interests while browsing & put the user interested snippets at the top position.

In table 5.2, the queries like "sachin" and "anil" are ambiguous. The table 5.2 shows that the *timePR* is greater than *timeOR* for these queries. This condition can occur for mainly three reasons: (1) If the user already visited some of the snippets already for these queries, but currently not interested in them. The user wants to try some new snippets to get new information. (2) The user visits to particular snippet once, but does not get any related information there. (3) The user is submitting the query freshly i.e. profile of that user does not contain much information about it. But, the users belonging to the similar group (based on some other query) may visit some of the snippets from search result. So, these results forcefully appended at the top position in search result. But, after a long-time using, the content of our models can be richer and the effect of our approach can be improving.

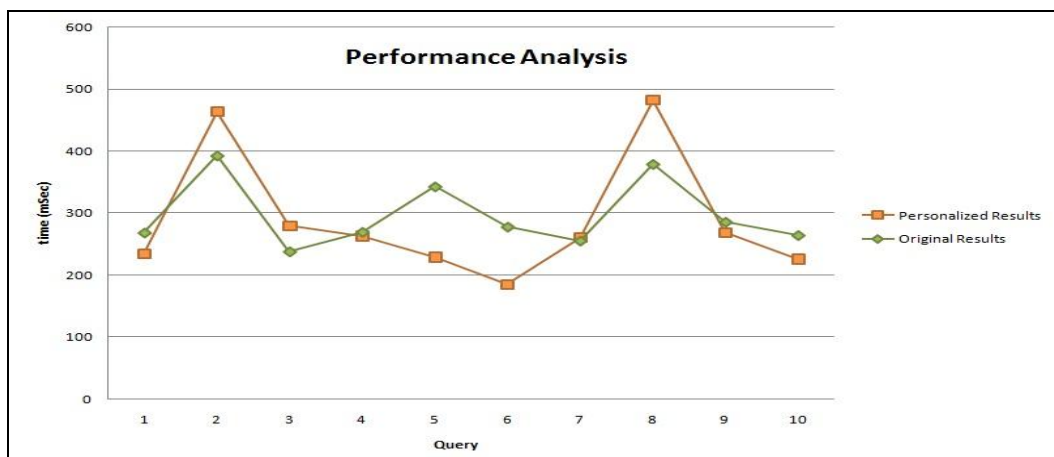
Finally, form the Avg. row in Table 5.2, it is conclude that, comparing with the order given out by Google API, our approach can do some improving by doing reordering as the value of Avg. row and improving column is greater than 0.



**Table 5.2 Summary of the experimental results**

Sr. no.	Query	timePR (mSec)	timeOR (mSec)	Improving
1	Google	235.25	268.875	14.29%
2	Sachin	463.75	393.125	-17.96%
3	Food	280.375	238.5	-17.55%
4	Flipkart	263.875	270	2.32%
5	Computer	229.625	343.25	21.03%
6	Jobs	185.875	278.375	49.48%
7	Cinema	261	255.875	-2%
8	Anil	482.125	379.5	-27.04%
9	College	269.375	286.25	6.26%
10	Yahoo	226.625	264.75	16.82%
Avg.	--	289.79	297.85	2.78%

The comparison of original ranking results and personalized ranking results is discussed. The result of its comparison is also illustrated by using graph. The graph representing the performance in terms of *time* is shown below in fig 5.1.



**Fig 5.1 The graph representing performance of system**

The above graph shows that the performance of original ranking system is too weak than proposed system. The no. of terms in user profile also affects the relevancy. More no. of terms helps to provide more personalized data to the user. This is not applicable for original case example. But because of it doesn't purely depend upon personalization ranking, its performance degrades. The original ranking refers to the currently working search engines, which can track browsing history but doesn't utilizing it. Any no. of visits to particular snippet by the user doesn't make any changes to the ranking of that snippet. So, this refers as worst case condition.

**5.2 Correlation Coefficient**

Correlation coefficient is used to find how strong a relationship is between data. The formulas return a value between -1 and 1, where:

- 1 indicates a strong positive relationship.
- -1 indicates a strong negative relationship.
- A result of zero indicates no relationship at all.

A correlation coefficient of 1 means that for every positive increase of 1 in one variable, there is a positive increase of 1 in the other. A correlation coefficient of -1 means that for every positive increase of 1 in one variable, there is a negative decrease of 1 in the other. Zero means that for every increase, there isn't a positive or negative increase. The two just aren't related.

The absolute value of the correlation coefficient gives us the relationship strength. If there is larger number, there will be stronger the relationship. For example,  $|-0.75| = .75$ , which has a stronger relationship than 0.65.

**Table 5.3 Values needed for calculating the Correlation Coefficient**

Sr. No.	X	Y	XY	X <sup>2</sup>	Y <sup>2</sup>
1	268.875	235.25	63252.84	72293.77	55342.56
2	393.125	463.75	182311.72	154547.27	215064.06
3	238.5	280.375	66869.44	56882.25	78610.14
4	270	263.875	71246.25	72900	69630
5	343.25	229.625	78818.78	117820.56	52727.64
6	278.375	185.875	51742.95	77492.64	34549.52
7	255.875	261	66783.38	65472	68121
8	379.5	482.125	182966.44	144020.25	232444.52
9	286.25	269.375	77108.59	81939.06	72562.89
10	264.75	226.625	59998.97	70092.56	51358.89
Σ	ΣX=297.85	ΣY=289.79	ΣXY=901099.36	ΣX <sup>2</sup> =913460.36	ΣY <sup>2</sup> =930411.22

There are several types of correlation coefficient formulas. One of the most commonly used formulas in stats is Pearson's correlation coefficient formula.

**Correlation Coefficient =**

$$r = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{[n\sum x^2 - (\sum x)^2][n\sum y^2 - (\sum y)^2]}}$$

To find out the correlation coefficient for the system from above table values of x are made congruent to the value of 'time' required in original sequence and values of y are made congruent to the value of 'time' required in personalized ranking. Calculation of values is shown in table 5.3.

$$\text{Correlation coefficient} = \frac{10*(901099.36) - 297.85*289.79}{\sqrt{[10*913460.36 - 88714.62][10*930411.22 - 83978.24]}}$$

Correlation Coefficient = **0.9772**

From the above calculated value of correlation coefficient it is conclude that the personalized search results are match with original search results with 'strong positive' relationship.

### 5.3 Recall & Precision

This section discusses about the performance of the personalization system through the measuring parameters-recall, precision and f-measure. There are certain issues in the proposed system. After submitting the query, if the user visits any snippet, then entry of that snippet is made into the user profile of that user. But, after 8 days of query submission, if user doesn't resubmit that query, then the weight assigned to that snippet goes on decreasing. At last, when it goes below threshold value, it gets deleted from user profile. After that if user resubmits the same query with intension to get personalized result, he / she doesn't find that snippets in search result. The proposed system provides only 10 snippets to web search and 8 snippets to image search. So, even if the user intended snippet is present below this no. of top result, it is not presented to the user. So, these snippets are referred as 'relevant but not retrieved'. Another one example of it is, when the user profile contains certain snippet, but if Google API doesn't include it in search result, then it will not appear in final result even though it is relevant one.

The snippets which are present in the user profile and presented to the user as search result with high ranking is referred as 'relevant retrieved results'. And the remaining snippets which are present in search result below personalized results are referred as 'irrelevant retrieved results'.

In Personalization, **precision** (also called positive predictive value) is the ratio of the number of relevant records retrieved to the total number of irrelevant and relevant records retrieved, while **recall** (also known as sensitivity) is the ratio of the number of relevant records retrieved to the total number of relevant records in the database. The **F-measure** is often used in the field of information retrieval for measuring search, result classification, and query classification performance. Precision, Recall and F-measure are calculated by using contingency table shown in table 5.4:-

**Table 5.4 Contingency table**

	<b>Correct</b>	<b>Not correct</b>
<b>Selected</b>	TP	FP
<b>Not selected</b>	FN	TN

Precision = TP/ (TP + FP) Recall = TP/ (TP+FN)

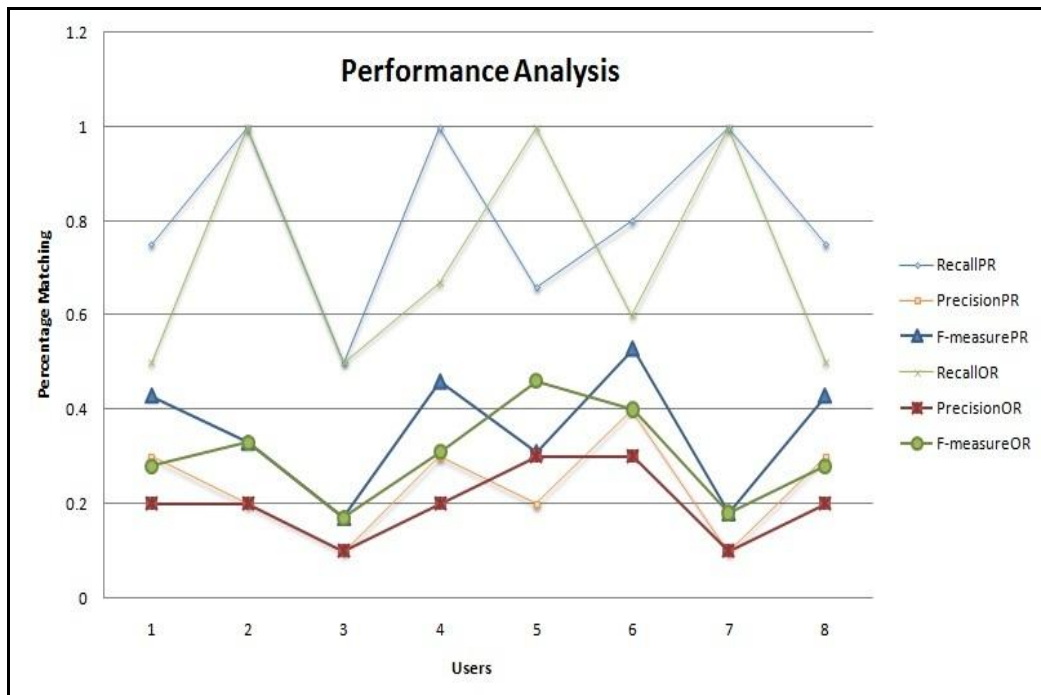
F-measure = (2\*Precision\*Recall) / (Precision + Recall)

Where TP =True Positive FP = False Positive FN = False Negative

Below table 5.5 & fig. 5.2 explains the performance of the system in terms of precision, recall and f-measure. A sample query ‘google’ is submitted by the 8 regular registered users (i.e. the users which are visited the snippets on ‘google’ in past) to determine the personalization accuracy.

**Table 5.5 Precision, Recall & F-measure values using Personalization for user query ‘google’**

User	No. of Expected Relevant Search Results	No. of Actual Relevant Search Results		Recall		Precision		F-measure	
		PR	OR	PR	OR	PR	OR	PR	OR
--	--	PR	OR	PR	OR	PR	OR	PR	OR
1	4	3	2	0.75	0.5	0.3	0.2	0.43	0.28
2	2	2	2	1	1	0.2	0.2	0.33	0.33
3	2	1	1	0.5	0.5	0.1	0.1	0.17	0.17
4	3	3	2	1	0.67	0.3	0.2	0.46	0.31
5	3	2	3	0.66	1	0.2	0.3	0.31	0.46
6	5	4	3	0.8	0.6	0.4	0.3	0.53	0.4
7	1	1	1	1	1	0.1	0.1	0.18	0.18
8	4	3	2	0.75	0.5	0.3	0.2	0.43	0.28
Avg.	3	2.375	2	0.81	0.72	0.24	0.2	0.37	0.30



**Figure 5.2 Precision, recall & F-measure graph for multiuser personalization**

## VI. CONCLUSION

In order to provide each user with more relevant information, several approaches were proposed to adapting search results according to each user's information need. This approach is novel in that it allows each user to perform a fine-grained search, which is not performed in typical search engines, by capturing changes in each user's preferences. Certain experiments were conducted in order to verify the effectiveness of the approaches: (1) relevance feedback and implicit approaches, (2) user profiles based on pure browsing history, and (3) user profiles based on the modified collaborative filtering. The user profile constructed based on modified collaborative filtering achieved the best accuracy. This approach allows constructing a more appropriate user profile and performing a fine grained search that is better adapted to each user's preferences.

## Acknowledgements

The authors thank X. Wu, Y. Fu, S. Tian, Q. Zheng and F. Tian for their valuable comments and edits in refining ideas for the paper. The research was supported by the information technology department of Sinhgad College of Engineering, Pune.

## REFERENCES

- [1]. Z. Dou, R. Song, J. Wen and X. Yuan, Evaluating the Effectiveness of Personalized Web Search, *IEEE Transactions on Knowledge and Data Engineering*, 21(8), 2009, 1178-1190.
- [2]. Prabaharan S. and R. S. D. Wahidabanu, Ontological Approach for Effective Generation of Concept Based User Profiles to personalize search results, *International Journal of Computer Science*, 8(2), 2012, 205-215.
- [3]. X. Wu, Y. Fu, S. Tian, Q. Zheng and F. Tian, A hybrid Approach to Personalized Web Search, *In Proc. of IEEE 16<sup>th</sup> International Conference on Computer Supported Cooperative Work in Design*, 12(7), 2012, 214-220.
- [4]. A. Sieg, B. Mobasher and R. Burke, Learning Ontological-Based User Profiles, *IEEE Intelligent Informatics Bulletin*, 8(1), 2007, 7-18.
- [5]. Sung Ho Ha, Seong Hyeon Joo and Hyun U. Pae, Searching for Similar Informational Articles in the Internet Channels, *International Journal of Computer Science and Engineering*, 2(1), 2007, 14-17.
- [6]. M. J. Pazzani and D. Billsus, Content-based Recommendation Systems, *Springer Journal- Verlag Berlin*, 2007, 325-341.
- [7]. R. R. Deshmukh and R. R. Keole, Web Search Personalization with Ontological Profiles, *International Journal of Advance Research in Computer Science and Software Engineering*, 2(12), 2012, 156-159.
- [8]. D.Wu, I. Im, M.Tremaine, K. Instone and M. Turoff, A Framework for classifying Personalization Scheme Used On e-commerce websites, *36<sup>th</sup> Hawaii International Conference*, 5(3), 2003.
- [9]. X. Amatriain, Mining Large Streams of User Data for Personalized Recommendation, *SIGKDD Explorations*, 14(2), 2005, 37-48.
- [10]. Z. Ma, Gautam Pant and Olivia R. Liu Sheng, Interest Based Personalized Search, *ACM Transactions on Information System*, 25(1), 2007, 1-38.