DOI: 10.21917/ijsc.2013.0086

# A NOVEL RELEVANCE METRIC PREDICTION ALGORITHM FOR A PERSONALIZED WEB SEARCH

**J. Jayanthi[1], M. Ezhilmathi[2] and S. Rathi[3]**
*Department of Computer Science and Engineering, Sona College of Technology, India*
E-mail: [1]jayamithu2002@gmail.com, [2]srkezhil@gmail.com and [3]rathi@gct.ac.in

***Abstract***
***Software metrics are the key performance indicators, using which the performance of a system can be assessed quantitatively. Metrics can also be applied for personalized web search which can be used to retrieve relevant results for each individual user depending on their unique profile. Although personalized search based on user profile has been under research for many years and various metrics have been proposed, it is still uncertain whether personalization is unswervingly effective on different queries for different user profiles. We present a framework for personalized search which retrieves result based on user profile and query type. Also we evaluate the performance of proposed system using relevance evaluation metrics.***

***Keywords:***
***Personalized Web Search, P-Click, G-Click, Profile Convergence***

## 1. INTRODUCTION

In rapid development of internet technologies, search engines plays pivotal role in information retrieval. Personalized search can be used to provide different search results depending on the user's preference. Various personalization strategies have been proposed so far, but a significant problem is that most algorithms are applied uniformly to all users and queries. Personalized web search has different levels of effectiveness for different queries, users and context, consequently a single personalization algorithm cannot improve accuracy of ranking for all queries and it may even affect the accuracy of search under certain circumstance. Hence it is substantial to evaluate the performance of various proposed strategies. Metrics can be applied to assess the performance of information retrieval systems and various personalization strategies. In personalized web search it is very essential to measure user satisfaction based on relevance metrics.

Predominantly relevance evaluation is done based on either implicit relevance judgments or explicit relevance judgments. Implicit data can be generated by users' interaction with their service. Implicit measures are easier to collect and allow us to explore many queries from vast variety of searchers. The trouble-free way to verify whether a result retrieved for a query is relevant to the user is to explicitly ask that user. Explicit judgment allows us to scrutinize the uniformity in relevance assessments across judges in a controlled setting. In the long run, distinct measures, metrics and algorithms been proposed to perform relevance evaluation by taking either implicit or explicit judgments as input. Yet there is no clear guideline regarding the type of evaluation metric to be used under various situations. As a solution to these problems, we develop an evaluation framework to predict the appropriate algorithm to be applied based on different criterion as well as metrics to evaluate the performance of the system. We provide a strategy to:

i. Gather and model user's search history
ii. A rule engine to deduce appropriate metrics and algorithms for each query and each user
iii. Metrics to evaluate the performance of the proposed system.
iv. Improve web search effectiveness by using these metrics and algorithms.

## 2. RELATED WORK

There have been many schemes for building user profiles, re-ranking/personalizing search results and for evaluating the efficiency of such re-ranked results. Most of them model user profiles represented by bag of words without considering term correlations [9, 10]. A simple taxonomic hierarchy considered as a tree structure has been widely accepted to overcome the drawbacks of the bag of words in [11, 12]. Studies [9,10] suggests that relevance feedback and machine learning techniques show promise in adapting to changes of user interests and reducing user involvements, while still overseeing what users dislike and their interest degradation [8].

### 2.1 PRECISION

Precision is one of the important metrics in the field of Information Retrieval to evaluate the performance of a retrieval system, say Search Engine. The number of documents that are relevant within the retrieved set of results for a given query is called as precision. Precision ($P$) is the fraction of retrieved documents that are relevant. The formula for precision is given as,

$$\begin{aligned}\text{Precision} &= \frac{\#(\text{relevant items retrieved})}{\#(\text{retrieved items})} \\ &= \text{P}(\text{relevant} \mid \text{retrieved})\end{aligned} \tag{1}$$

The contingency table can be given as,

Table.1. Contingency Values

| | **Relevant** | **Non relevant** |
|---|---|---|
| **Retrieved** | True positives (TP) | False positives (FP) |
| **Not retrieved** | False negatives (FN) | True negatives (TN) |

So precision can also be represented as,

$$P = \frac{TP}{TP + FP} \tag{2}$$

Seig et al. maintained models of users' context by building ontological user profiles, which were created using implicitly derived interest scores to existing concepts in domain ontology [3]. The factors considered by them for maintaining user profile are: (i) frequency of visit to a page, (ii) time spent on a page and (iii) other actions such as bookmarking. They proposed a ‘Spreading activation algorithm’ to maintain interest score based on user’s behavior. Search results were re-ranked based on interest score and semantic evidence in user profile using a re-ranking algorithm. The effectiveness of re-ranking was measured in terms of Top-n-Precision and Top-n-Recall. The precision at top 30 documents is shown to be 0.25. Experimental result shows that in this method, the precision gradually decreases with increasing number of documents.

Liu et al. maintained two categories of profiles namely user profile and general profile. User profile and search history is maintained in the form of Document-Term, Document-Category and Category-Term matrix [2]. General profile is maintained based on concept hierarchy from ODP category. General profile has interests of all users, which is useful when some users have new interests. They proposed a method in which when user issues a query, it is matched to a category based on user profile and general profile. Then the categories are re-ranked in descending order of similarity. Top three categories will be displayed to the user from which the user can select the category of their interest. If the user is not satisfied with top three categories, the next three categories in the list will be displayed. This mode is called as semi-automatic mode. When the top category is chosen by the system automatically then it is called as automatic mode. The main factors considered in maintaining user profiles are the query issued by the user, documents viewed by the user for that particular query and the ODP category of those documents. When compared to baseline mode, the retrieval effectiveness of automatic mode improved by 13 percent ($P$ = 0.49) and that of semi-automatic mode by 25.6 percent ($P$ = 0.55).

The relevance feedback given by the user for a set of documents returned by the search engine for a query issued by the user can be used to improve retrieval effectiveness. Lv and Zhai [5] proposed an adaptive relevance feedback method to predict an optimal balance coefficient between query and feedback information using machine learning. They also proposed three heuristics to characterized feedback coefficients:

i. Discrimination of query,
ii. Discrimination of feedback documents and
iii. Divergence between query and feedback documents, for each of which several measures are proposed to quantify them.

Query length, Entropy of query and Clarity of query for discrimination of query; Feedback length, feedback radius, entropy of feedback documents and clarity of feedback documents for discrimination of feedback documents; Absolute divergence and Relative divergence for divergence between query and feedback documents. Logistic regression model is used as learning algorithm to combine all the above measures to generate a score for predicting feedback coefficients. The experimental results shows that the precision of AdaptFB method to be 0.552.

Kajaba et al. proposed a method in which personalization is done by augmenting the query with additional keywords, extracted from Interested Person’s (IP) profile, which is represented by a set of weighed keywords called as keyword cloud. In order to maintain the size of the profile, it is updated at regular intervals using an algorithm [4]. They proposed a middleware called “Finder”, which augments the query with keywords from IP’s profile and also suggests keywords to IP that can be added to original query. Profile is updated based on behavior of user. Main factor considered is the clicks made by the user. The precision of the system is shown to be 0.87 from the experimental results made by them.

One of the personalization method used by Dou et al. is L-Topic, which is based on long-term topical interest of the users. The user interest is represented as a vector of 67 predefined topic categories [7]. Each web page, returned by the system for a query issued by the user is mapped to a category vector. Using cosine similarity, the similarity between user profile vector and page category vector can be computed. User profile is based on past clicks made by the user. The similarity between user interests and a web page is used to re-rank search results [6]. The precision of L-Topic is shown to be 0.8917.

**Advantage** - It doesn’t require any estimate of the size of the set of relevant documents.

**Disadvantage**

- Precision is least stable.
- It doesn’t average well since total number of relevant documents for a query has a strong influence on precision at $k$.

## 2.2 DISCOUNTED CUMULATIVE GAIN (DCG)

Two main factors to be considered while examining the ranked result list of a query are:

- Highly relevant documents are more valuable than marginally relevant documents and
- Greater the ranked position of the relevant document, less valuable it is for the user since the probability of user viewing the document is less.

In cumulative gain, each document is rated according to the gain values/relevance score from 0-3, where 3 indicates highly relevant and 0 completely irrelevant. A discounting function is needed which gradually reduces documents score as its rank increases but not too steeply for allowing user persistence in viewing further documents. The formula for DCG is given by,

$$DCG_r = REL_1 + \sum_{i=2}^{r} \frac{REL_i}{\log_2 i} \quad (3)$$

where, $REL_i$ -Relevance of the document at rank $i$.

$\frac{1}{\log_2 i}$ – Discount factor

In [11, 12] NDCG is used for performance evaluation. DCG is widely used for evaluating performance of systems with explicit feedback.

# 3. PROPOSED METRIC SUITE

We propose an evaluation framework which can automatically identify the type of relevance metric and algorithm to be applied based on various criterions that contribute to the rank score of the result.

## 3.1 PROFILE CLASSIFICATION

We classify user profile into three categories:

- Converged profile
- Semi-converged profile
- Non-converged profile

***Converged Profile (CP):***

A profile is said to be converged if the ratio of repeated queries are very higher than the ratio of unique queries.

***Semi-Converged Profile (SCP):***

A profile is said to be semi-converged if the ratio of repeated queries and unique queries are more or less equal.

***Non-Converged Profile (NCP):***

A profile is not converged if the ratio of unique queries is very higher than that of the repeated queries.

### *3.1.1 Profile Transformation:*

At certain point of time, there are chances that a semi-converged or non-converged profile can shift into a converged profile and vice versa, upon long time usage or repeated issue of queries by the users. So when the profile type changes, the type of metric to be applied will also be changing accordingly. The formula for predicting the point of transformation (PT) of a user's profile from one type to another is,

$$\mathrm{PT_i} = \frac{\text{Queries with RPq} > 10\%}{\#(\text{unique queries}_i)} * 100 \tag{4}$$

where, # (unique queries $_i$) -Total count of unique queries in user profile *i*.

where if,

$PT_i > 15$ - Converged profile;

$7 \le PT_i \le 15$ - Semi-converged profile;

$0 \le PT_i \le 6$ - Non converged profile.

### *3.1.2 Repetition Percentage of a Query (rpq):*

The number of times a query is repeated by a user is indicated by repetition percentage of a query.

$$RPq = \frac{NR_i}{\#(Queries_i)} * 100 \tag{5}$$

where, $NR_i$ - Number of re-occurrence of query *q* in profile *i*.

# ($\text{Queries}_i$) - Total number of queries in the profile *i*.

## 3.2 QUERY CLASSIFICATION

Queries are classified into following categories:

i. Type-1: Self-Repeated Query (SRQ)
ii. Type-2: Repeated Query (RQ)
iii. Type-3: SRQ-RQ

***Self-Repeated Query (SRQ):***

When a user issues a query which is previously issued only by that user and which is not issued by any other user then it is a Self-Repeated Query.

***Repeated Query (RQ):***

If a query issued by a user is not in that user's search history but in the search history of other users, then it is a repeated query.

***SRQ-RQ:***

If a query issued by a user in the search history of both the current user and other users, then it belongs to this type.

## 3.3 METRICS

The two metrics used for ranking search results are:

i. P-Click
ii. G-Click

***P-Click:***

The formula for calculating P-Click [7] score is:

$$P - Click_{Doc_n}(q, p, u) = \frac{|Clicks(q, p, u)|}{|Clicks(q, \blacksquare, u)| + \beta} \tag{6}$$

where, |*Clicks* (*q*, *p*, *u*)| - number of clicks on web page *p* for the query *q* by the user *u*

|*Clicks* (*q*, ■, *u*)| - total number of clicks for query *q* by *u*

$\beta$ - smoothing factor.

***G-Click:***

The formula for calculating G-Click is:

$$G - Score_{Doc_n} = \frac{\sum_{i=1}^{N} \left(P - Click_{Doc_n}\right)_i}{N} \tag{7}$$

where, $(P\text{-}Click_{Doc_n})_i$ - *P-Click* score of $Doc_n$ of user *i*

*N* – Total number of user profiles which contains $Doc_n$.

## 3.4 RELEVANCE METRIC PREDICTION ALGORITHM

When a user (for example A), issues a query (say, python), as a first step the algorithm checks the profile type of A and then the query category under which the issued query falls. If A's profile is identified to be converged, then the query is classified as per the rules specified. If,

- Type 1: P-Click value is sufficient enough to rank the retrieved result list.
- Type 2: G-Click value is used for ranking the results.
- Type 3: Top results are listed using P-Click score and remaining results based on G-Click score of the user.

If A's profile is classified into Semi-converged profile, then for,

- Type 1: If the query is repeated twice at the minimum by the user, then P-Click value is used. Else normal ranking is used.
- Type 2: G-Click value is used for ranking the results.
- Type 3: If the query is repeated no less than three times by the user, then P-Click value is sufficient enough to rank the retrieved result list. Else G-Click score is used.

If A's profile is non-converged, then for,

- Type 1: P-Click for top results and normal search for remaining results.
- Type 2: G-Click value is used.

- Type 3: If the query is repeated no less than three times by the user, then P-Click value is sufficient enough to rank the retrieved result list. Else G-Click score is used.

**Input:** Query, user profile
**Pre-Condition:** User must be logged in
**Output:** Relevant metrics
Input query Q
Check User Profile (UP) type and query (Q) type
**If** ((UP Є CP) && (Q Є Type 1)) **then** P-Click
**Else-if** (Q Є Type 2) **then** G-Click
**Else** P-Click and G-Click
**If** ((UP Є SCP) && (Q Є Type 1) && (*RPq* >10%))
**then** P-Click
**Else-if** ((Q Є Type 3) && (*RPq* > 10%))
**then** P-Click
**Else** G-Click
**If** ((UP Є NCP) && (Q Є Type 1)) **then** P-Click
**Else-if** (Q Є Type 3) && (*RPq* >10%)
**then** P-Click
**Else** G-Click
**End**

Fig.1. Prediction Algorithm

The next step is to predict the appropriate relevance evaluation metric to evaluate the performance of proposed metrics. For converged profiles, vast details of click-through data and relevant documents will be available i.e. implicit content about users' usage of result list will be available. In such cases, it is feasible to use precision for evaluating the performance of metrics used for converged profile. For semi-converged and non-converged profiles, appropriate implicit data will not be available. In such cases, obtaining explicit feedback from the user about result list will be much suitable for evaluating the performance of relevance metrics. Hence for type 2 and type 3 profiles DCG can be used to evaluate the performance of the metrics.

## 4. EXPERIMENTAL RESULTS

### 4.1 DATA SET

For a period of 30 days, 30 users were made to use the proposed search system for searching various queries. The click-through data, queries issued by the users, P-Click and G-Click scores and repetition percentages of each query and profile convergence level of each user was maintained. Table.3, Table.4 and Table.5 contains a sample for Converged, Semi-Converged and Non-Converged user profiles respectively. Table.6 shows the sample values for G-Click scores of few links.

Table.2. Sample Data Set

| Item | Nos. |
|---|---|
| #days | 30 |
| #users | 30 |
| #sessions | 900 |
| #converged profile | 10 |
| #semi-converged profile | 10 |
| #non-converged profile | 10 |
| #queries | 100 |

### 4.2 PERFORMANCE EVALUATION

The Fig.2 shows the performance of proposed system for all three types of queries based on profile convergence level of the users. The system performance is comparatively high for Type-1 and Type-3 queries rather than Type-2 queries. Fig.3 shows the overall performance of the system irrespective of query type. The experimental results show that the system gives better performance as the convergence level of user profile increases.

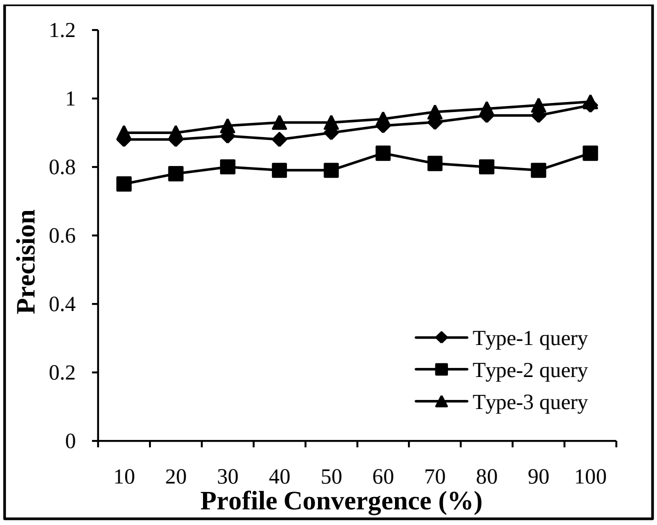


Fig.2. Precision for different query types

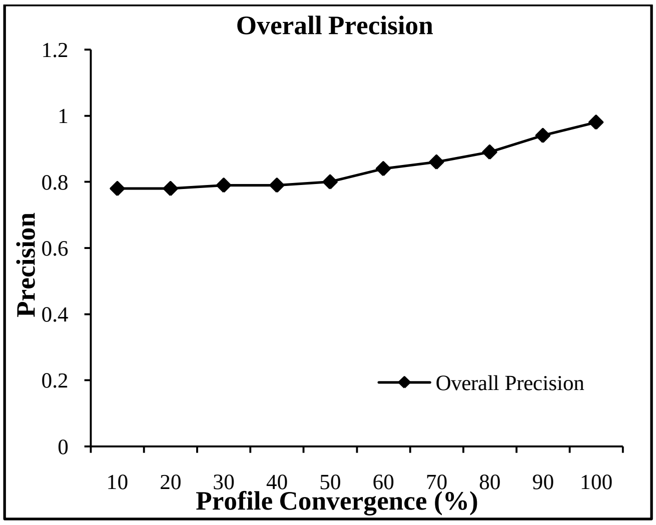


Fig.3. Overall precision of the system for different levels of profile convergence

## 5. CONCLUSION

In this paper, we have proposed an evaluation framework for prediction of metrics and algorithms to be applied for retrieving relevant web search results for individual users. We further proposed techniques and strategies for classifying user profiles and queries. Metrics for updating user profile category has also been proposed. This approach would be useful to improve search accuracy and for retrieving relevant results for each individual user depending on their preference. Performance of the proposed system is also evaluated using relevance evaluation metrics.

## APPENDIX

Table.3. Sample for Converged user profile (80%)

| Query | Repetition Percentage | Doc | P-Click |
|---|---|---|---|
| Computer science | 23 | D1 | 0.424242 |
| | | D2 | 0.363636 |
| | | D3 | 0.181818 |
| Operating system | 27 | D1 | 0.4186 |
| | | D2 | 0.32558 |
| | | D3 | 0.23256 |
| Software | 34 | D1 | 0.44444 |
| | | D2 | 0.4 |
| | | D3 | 0.13333 |
| Algorithms | 56 | D1 | 0.51429 |
| | | D2 | 0.34286 |
| | | D3 | 0.11429 |
| AI | 78 | D1 | 0.412026 |
| | | D2 | 0.35897 |
| | | D3 | 0.20513 |
| Hardware | 65 | D1 | 0.38596 |
| | | D2 | 0.31579 |
| | | D3 | 0.28070 |
| Graphics | 32 | D1 | 0.41026 |
| | | D2 | 0.30769 |
| | | D3 | 0.25641 |
| Programming | 45 | D1 | 0.51852 |
| | | D2 | 0.29629 |
| | | D3 | 0.14185 |
| HCI | 65 | D1 | 0.45714 |
| | | D2 | 0.28571 |
| | | D3 | 0.22857 |
| Data Communication | 43 | D1 | 0.40678 |
| | | D2 | 0.33898 |
| | | D3 | 0.23729 |
| Mobile computing | 23 | D1 | 0.46154 |
| | | D2 | 0.35897 |
| | | D3 | 0.15385 |
| Security | 76 | D1 | 0.4 |
| | | D2 | 0.35556 |
| | | D3 | 0.22222 |
| IDS | 56 | D1 | 0.46154 |
| | | D2 | 0.30769 |
| | | D3 | 0.20513 |
| Internet | 55 | D1 | 0.51429 |
| | | D2 | 0.28571 |
| | | D3 | 0.17143 |
| Malicious software | 34 | D1 | 0.43243 |
| | | D2 | 0.32432 |
| | | D3 | 0.21622 |
| VPN | 27 | D1 | 0.4 |
| | | D2 | 0.25455 |
| | | D3 | 0.32727 |
| Cryptography | 67 | D1 | 0.41026 |
| | | D2 | 0.30769 |
| | | D3 | 0.25641 |
| Biometrics | 89 | D1 | 0.37288 |
| | | D2 | 0.33898 |
| | | D3 | 0.27119 |
| Hacking | 55 | D1 | 0.39024 |
| | | D2 | 0.34146 |
| | | D3 | 0.24390 |
| Firewall | 41 | D1 | 0.42105 |
| | | D2 | 0.31579 |
| | | D3 | 0.21053 |
| Authentication | 31 | D1 | 0.46154 |
| | | D2 | 0.30769 |
| | | D3 | 0.15385 |
| Honey pots | 71 | D1 | 0.44444 |
| | | D2 | 0.22222 |
| | | D3 | 0.22222 |

| | | | |
|---|---|---|---|
| MAC OS | 72 | D1 | 0.53333 |
| | | D2 | 0.26667 |
| | | D3 | 0.13333 |
| Java | 62 | D1 | 0.54545 |
| | | D2 | 0.18182 |
| | | D3 | 0.18182 |
| MIS | 4 | D1 | 0.44444 |
| | | D2 | 0.22222 |
| | | D3 | 0.22222 |
| Chats | 5 | D1 | 0.36364 |
| | | D2 | 0.36364 |
| | | D3 | 0.18182 |
| Forums | 7 | D1 | 0.44444 |
| | | D2 | 0.22222 |
| | | D3 | 0.22222 |
| Open source | 8 | D1 | 0.44444 |
| | | D2 | 0.22222 |
| | | D3 | 0.22222 |
| Dot net | 6 | D1 | 0.44444 |
| | | D2 | 0.22222 |
| | | D3 | 0.22222 |
| Asp.net | 9 | D1 | 0.4 |
| | | D2 | 0.35556 |
| | | D3 | 0.22222 |

Table.4. Sample for Semi-Converged User profile (60%)

| Query | Repetition Percentage | Doc | P-Click |
|---|---|---|---|
| VPN | 25 | D1 | 0.36364 |
| | | D2 | 0.36364 |
| | | D3 | 0.18182 |
| Hacking | 45 | D1 | 0.44444 |
| | | D2 | 0.22222 |
| | | D3 | 0.22222 |
| Web service | 65 | D1 | 0.44444 |
| | | D2 | 0.22222 |
| | | D3 | 0.22222 |
| C # | 43 | D1 | 0.44444 |
| | | D2 | 0.22222 |
| | | D3 | 0.22222 |
| Open source | 21 | D1 | 0.4 |
| | | D2 | 0.35556 |
| | | D3 | 0.22222 |
| Soap | 36 | D1 | 0.43243 |
| | | D2 | 0.32432 |
| | | D3 | 0.21622 |
| Cryptography | 78 | D1 | 0.4 |
| | | D2 | 0.25455 |
| | | D3 | 0.32727 |
| Multimedia | 98 | D1 | 0.41026 |
| | | D2 | 0.30769 |
| | | D3 | 0.25641 |
| Asp.net | 73 | D1 | 0.37288 |
| | | D2 | 0.33898 |
| | | D3 | 0.27119 |
| Firewall | 42 | D1 | 0.39024 |
| | | D2 | 0.34146 |
| | | D3 | 0.23729 |
| Biometrics | 72 | D1 | 0.46154 |
| | | D2 | 0.35897 |
| | | D3 | 0.15385 |
| Authentication | 63 | D1 | 0.4 |
| | | D2 | 0.35556 |
| | | D3 | 0.22222 |
| Email | 31 | D1 | 0.46154 |
| | | D2 | 0.30769 |
| | | D3 | o.20513 |
| Animation | 91 | D1 | 0.51429 |
| | | D2 | 0.28571 |
| | | D3 | 0.17143 |
| Framework | 53 | D1 | 0.46154 |
| | | D2 | 0.30769 |
| | | D3 | 0.15385 |
| Forums | 71 | D1 | 0.44444 |
| | | D2 | 0.22222 |
| | | D3 | 0.22222 |
| Dot net | 24 | D1 | 0.53333 |
| | | D2 | 0.26667 |
| | | D3 | 0.13333 |
| Laptops | 34 | D1 | 0.54545 |
| | | D2 | 0.18182 |
| | | D3 | 0.18182 |
| Dell inspiron | 11 | D1 | 0.44444 |
| | | D2 | 0.22222 |
| | | D3 | 0.24390 |
| Ipad | 9 | D1 | 0.42105 |
| | | D2 | 0.31579 |
| | | D3 | 0.21053 |

| | | | |
|---|---|---|---|
| Samsung | 4 | D1 | 0.46154 |
| | | D2 | 0.15385 |
| | | D3 | 0.30769 |
| Cadburys | 7 | D1 | 0.22222 |
| | | D2 | 0.44444 |
| | | D3 | 0.22222 |
| Breaking dawn | 3 | D1 | 0.44444 |
| | | D2 | 0.22222 |
| | | D3 | 0.22222 |
| DELL | 7 | D1 | 0.54545 |
| | | D2 | 0.18182 |
| | | D3 | 0.18182 |
| Sony vaio | 8 | D1 | 0.42105 |
| | | D2 | 0.31579 |
| | | D3 | 0.22222 |
| Apple laptops | 14 | D1 | 0.38596 |
| | | D2 | 0.31579 |
| | | D3 | 0.28070 |
| Twilight | 17 | D1 | 0.41026 |
| | | D2 | 0.30769 |
| | | D3 | 0.25641 |
| HP laps | 8 | D1 | 0.51852 |
| | | D2 | 0.29629 |
| | | D3 | 0.14185 |
| Diary milk | 13 | D1 | 0.45714 |
| | | D2 | 0.28571 |
| | | D3 | 0.22857 |
| Lenovo | 18 | D1 | 0.40678 |
| | | D2 | 0.33898 |
| | | D3 | 0.38596 |

Table.5. Sample for Non-Converged profile (30%)

| **Query** | **Repetition Percentage** | **Doc** | **P-Click** |
|---|---|---|---|
| +2 results | 45 | D1 | 0.28571 |
| | | D2 | 0.17143 |
| | | D3 | 0.46154 |
| English serials | 87 | D1 | 0.30769 |
| | | D2 | 0.15385 |
| | | D3 | 0.44444 |
| Kai po che | 48 | D1 | 0.22222 |
| | | D2 | 0.22222 |
| | | D3 | 0.53333 |
| Tamil Nadu govt logo | 55 | D1 | 0.26667 |
| | | D2 | 0.13333 |
| | | D3 | 0.22222 |
| Twitter | 64 | D1 | 0.44444 |
| | | D2 | 0.22222 |
| | | D3 | 0.30769 |
| Mathematics | 4 | D1 | o.20513 |
| | | D2 | 0.51429 |
| | | D3 | 0.56432 |
| TMB | 8 | D1 | 0.36364 |
| | | D2 | 0.36364 |
| | | D3 | 0.18182 |
| Chart tools | 1 | D1 | 0.44444 |
| | | D2 | 0.22222 |
| | | D3 | 0.22222 |
| Samsung | 9 | D1 | 0.44444 |
| | | D2 | 0.22222 |
| | | D3 | 0.22222 |
| MIS | 14 | D1 | 0.44444 |
| | | D2 | 0.22222 |
| | | D3 | 0.22222 |
| Java | 34 | D1 | 0.4 |
| | | D2 | 0.35556 |
| | | D3 | 0.22222 |
| Sony | 56 | D1 | 0.43243 |
| | | D2 | 0.32432 |
| | | D3 | 0.21622 |
| Smart draw | 19 | D1 | 0.54545 |
| | | D2 | 0.18182 |
| | | D3 | 0.18182 |
| Nokia | 2 | D1 | 0.44444 |
| | | D2 | 0.22222 |
| | | D3 | 0.24390 |
| SBI | 7 | D1 | 0.42105 |
| | | D2 | 0.31579 |
| | | D3 | 0.21053 |
| KVB | 8 | D1 | 0.46154 |
| | | D2 | 0.15385 |
| | | D3 | 0.30769 |
| LVB | 59 | D1 | 0.44444 |
| | | D2 | 0.22222 |
| | | D3 | 0.22222 |

| | | | |
|---|---|---|---|
| IOB | 78 | D1 | 0.54545 |
| | | D2 | 0.18182 |
| | | D3 | 0.18182 |
| Linked In | 11 | D1 | 0.42105 |
| | | D2 | 0.31579 |
| | | D3 | 0.22222 |
| HCL | 13 | D1 | 0.38596 |
| | | D2 | 0.31579 |
| | | D3 | 0.28070 |
| tngov.in | 12 | D1 | 0.41026 |
| | | D2 | 0.30769 |
| | | D3 | 0.25641 |
| Face book | 17 | D1 | 0.51852 |
| | | D2 | 0.29629 |
| | | D3 | 0.14185 |
| Flowers | 16 | D1 | 0.45714 |
| | | D2 | 0.28571 |
| | | D3 | 0.22857 |
| Images for white flowers | 2 | D1 | 0.40678 |
| | | D2 | 0.33898 |
| | | D3 | 0.38596 |
| Top 10 Hindi movies | 4 | D1 | 0.28571 |
| | | D2 | 0.28571 |
| | | D3 | 0.28571 |
| Melody songs | 7 | D1 | 0.36364 |
| | | D2 | 0.18182 |
| | | D3 | 0.36364 |
| Board examinations | 9 | D1 | 0.28571 |
| | | D2 | 0.36364 |
| | | D3 | 0.36364 |
| TV serials | 5 | D1 | 0.28571 |
| | | D2 | 0.28571 |
| | | D3 | 0.36364 |
| Software | 4 | D1 | 0.22857 |
| | | D2 | 0.40678 |
| | | D3 | 0.33898 |
| AI | 3 | D1 | 0.38596 |
| | | D2 | 0.22857 |
| | | D3 | 0.40678 |

Table.6. Sample for G-Click scores

| Query | Link | G-Click |
|---|---|---|
| java | http://www.Cjava2snet.edu.aspx | 0.537349 |
| java | http://www.javas2sserver.com | 0.42035 |
| java | http://www.javaenters.net.aspx | 0.319693 |
| Mail server | http://www.mailservers.edn.aspx | 0.4492064 |
| Mail server | http://www.mailservers.net | 0.4492064 |
| Mail server | http://www.mailservers.ip.aspx | 0.4644689 |
| Mail server | http://www.mailservers.ips.aspx | 0.4078144 |
| soap | http://www.soapipsserver.com | 0.615392 |
| soap | http://www.webservices.edu.aspx | 0.6512987 |
| framework | http://www.framework.edu.aspx | 0.4644689 |
| framework | http://www.framework45.edu.aspx | 0.6507985 |
| java tutorial | http://www.Cjava2snet.edu.aspx | 0.7555556 |
| C sharp | http://www.csharp45.edu.aspx | 0.7111111 |

## REFERENCES


[1] J. Jayanthi, M. Ezhilmathi and S. Rathi, “Evaluating the Effectiveness of Web Search Metrics”, *International Journal of Applied Information Systems*, Vol. 4, No. 10, pp. 28-35, 2012.

[2] Fang Liu, Clement Yu and Weiyi Meng, “Personalized web search for improving retrieval effectiveness”, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 16, No. 1, pp. 28-40, 2004.

[3] Ahu Sieg, Bamshad Mobasher and Robin Burke, “Web search personalization with ontological user profiles”, *Proceedings of the sixteenth ACM conference on Conference on Information and Knowledge Management*, pp. 525-534, 2007.

[4] Yuanhua Lv and Cheng Xiang Zhai, “Adaptive relevance feedback in information retrieval”, *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, pp. 255-264, 2009.

[5] Michal Kajaba, Pavol Navrat and Daniela Chuda, “A simple personalization layer improving relevancy of web search”, *Journal on Computing and Information Systems*, Vol. 13, No. 3, pp. 29-35, 2009.

[6] Z. Dou, R. Song, J.-R. Wen and X. Yuan, “Evaluating the Effectiveness of Personalized Web Search”, *IEEE Transaction on Knowledge and Data Engineering*, Vol. 21, No. 8, pp. 1178-1190, 2009.

[7] Y. Li, Z. Zheng and H. K. Dai, “KDD CUP-2005 Report: Facing a Great Challenge”, *ACM SIGKDD Explorations Newsletter*, Vol. 7, No. 2, pp. 91-99, 2005.

[8] Li Lin, "Enhancing Web Search by Personalized Re-ranking and Related Keyword Suggestion", PhD dissertation, University of Tokyo, 2009.

[9] D. Billus and M. J. Pazzani, "A hybrid user model for news story classification", *Proceedings of 7th International Conference on User Modeling*, pp. 99-108, 1999.

[10] W. Lam, S. Mukhopadhyay, J. Mostafa and M. Palakal, "Detection of shifts in user interests for personalized information filtering", *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 317-325, 1996.

[11] Paul Alexandru Chirita, W. Nejdl, R. Paiu and C. Kohlschutter, "Using ODP metadata to personalized search", *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 178-185, 2005.

[12] H. R. Kim and P. K. Chan, "Learning Implicit user interest hierarchy for context in personalization", *Proceedings of the International Conference on Intelligent User Interfaces*, pp. 101-108, 2003.

[13] Jaime Teevan, Susan T. Dumais and Daniel J. Liebling, "To personalize or not to personalize: modeling queries with variation in user intent", *Proceedings of the 31st annual international ACM SIGIR conference on Research and Development in Information Retrieval*, pp. 163-170, 2008.

[14] Jia Hu and Philip K. Chan, "Personalized web search by using learned user profiles in re-ranking", PhD dissertation, Florida Institute of Technology, 2008.