Combining User Interested Topic and Document Topic for Personalized Information Retrieval

K. Veningston and R . Shanmugalakshmi

Department of Computer Science and Engineering,
Government College of Technology, Coimbatore, India
{veningstonk,cseit.gct}@gmail.com

Abstract. Personalization aims to improve user's searching experience by tailoring search results according to individual user's interests. Typically, search engines employ two-level ranking strategy. Firstly, initial list of documents is prepared using a low-quality ranking function that is less computationally expensive. Secondly, initial list is re-ranked by machine learning algorithms which involve expensive computation. The proposed approach explores the second level of ranking strategy which exploits user information. In this approach, queries and search-result clicks are used to model the user interest profiles probabilistically. The user's history provides the prior probability that a user searches for a topic which is independent of user query. The document topical features are combined with user specific information to determine whether a document satisfies user's information need or not. The probability of relevance of each retrieved document for a query is computed by integrating user topic model and document topic model. Thus, documents are re-ranked according to the personalized score computed for each document. The proposed approach has been implemented and evaluated using real dataset similar to AOL search log dataset for personalization. Empirical results along with the theoretical foundations of the model confirm that the proposed approach shows promising results.

Keywords: Information Retrieval, Personalization, Re-ranking, Probabilistic model, Topic model.

1 Introduction

Web Information Retrieval (IR) [4][27] process faces the problems of information mismatching and overcapacity. As the amount of information on the Web increases rapidly, it creates many new challenges for Web search. When the same query is submitted by different users, a typical search engine returns the same result, regardless of who submitted the query. However with the recent advent of click data [38], web search engine now personalize search results quite often. At the same time, user's current interest for the same query may be different at different times, different places. The current web search approach may not be suitable for users with different information needs. For example, upon the query "java", some users may be interested in

S. Srinivasa and S. Mehta (Eds.): BDA 2014, LNCS 8883, pp. 60–79, 2014.

documents dealing with “java” as “a programming language” while some other users may want documents relating to “an island of Indonesia”. This kind of queries is referred to as an ambiguous query which mean to more than one category of results. For this kind of an ambiguous query, different users may have different search goals when they submit it to a search engine. However, it should not be treated as an ambiguous query. If it is possible for the search engine to derive user interests upon the query, then the user's intention becomes obvious. Personalized IR has become a promising area for disambiguating the web search query and therefore improving retrieval effectiveness by modeling the user profile by using his/her interests and preferences. While many search engines take advantage of information about people in general, or about specific groups of people, personalized search [17] depends on a user profile that is unique to the individual. Often short queries are ambiguous which provides very little information to a search engine on which the most relevant Web pages among millions need to be selected. A user profile can be used to supplement information about the search that is currently being represented by the query itself. This information can be used to narrow down the number of topics/contexts to be considered while retrieving the results. This increases the likelihood of including the most interesting results from the user's perspective. Typically the commercial search engine such as Google uses cookies and location information in order to personalize advertisements and most likely search results as well. The main contribution of the proposed approach is to exploit the topics in addition to that of treating whole document and search history towards personalizing search.

In this work, user’s search history, which is kept in a log format recording which queries the user has made in the past and which results he/she has chosen to view is utilized. This could be an important form of search context for the following reasons. First, a user’s background and interests can be learned from the user’s search history (e.g., by looking at the topics covered by the past queries). For example, if there have been a lot of queries like “car racing” and “Porsche club”, the user is probably interested in sports cars and “jaguar” is likely to mean the car. Second, from the users past (implicit) indication of document relevance, his/her reaction to the current retrieved documents implicitly provides an indication of his interests. For example, if the user searched with the same query “jaguar” before and clicked on Jaguar US’s homepage link, with high confidence it can be predicted that the user would do it again this time, and it makes good sense to list that webpage in the top. Even when there is no exact occurrence of the current query in history, still some similar queries would be helpful.

2 Current Practice and Research

When search query is issued, most of the search engines return the same results irrespective of the users’ interest because it lacks the existence of a semantic structure and hence it requires understanding of the information provided by the user. The following reasons accumulate complexity of the search process. The process of identifying intention of the user becomes difficult due to information available about user are very limited. At the same time, users do not wish to express their interest explicitly.

They want information instantaneously on supplying search query. Most of the users supply inaccurate input keyword query which is imprecise. They often under specify their true information needs. Thus query becomes ambiguous which needs to be understood by the retrieval system. Hence, personalization [29] strategy needs to be adopted in order to solve these problems faced by the retrieval system.

2.1 Related Work

- **Short-term and Long-term personalization:** Short term personalization [20] describes a personalized search based on the current user session. This approach is shown to improve retrieval quality. Long term personalization [24] describes a personalized search based on the entire history of user search in order to learn about the long term user characteristics.
- **Session based personalization:** Most of the personalized retrieval strategies do not distinguish between short term and long term user interests and make use of the whole search history to improve the search accuracy. Thus session based personalization [23] learns user interests by aggregating concept-based short terms identified within related search sessions.
- **Query ambiguity prediction:** Given an ambiguous query, it is either preferable to adapt the search result to a specific aspect that may be of the user's interest or to predict multiple aspects in order to maximize the probability that some query aspect is relevant to the user.
- **Implicit user modeling:** Typical retrieval systems lack user modeling and are not adaptive to individual users. Thus it is essential to infer a user's interest from the user's search context and use the inferred implicit user model for personalized search. In [37], the previous query has been exploited to enrich the current query and provide more search context to help disambiguation if two consecutive queries are related. This approach also infers user's interest based on the summaries of the viewed documents. The computed new user model is then be used to rank the documents with a standard information retrieval model.
- **Collaborative personalization:** In order to increase the user satisfaction towards online information search, search engine developers try to predict user preferences based on other user behavior. Thus, recommendations provided by the search engines may support users at some extent. Collaborative personalization attempts to better understand whether groups of people can be used to benefit from personalized search. The approach proposed in [15], combines individual's data with that of other related people to enhance the performance of personalized search. The use of group information for personalization is termed as groupization.
- **Search interaction personalization:** The approach presented in [8] incorporates user behavior data in order to improve the ordering of top results in real web search setting. It examines the alternatives for incorporating feedback into the ranking process and explores the contributions of search user feedback. The approach presented in [13] uses click-through information for improving web search ranking and it captures only one aspect of the user interactions with web search engines.

- **Ontology based personalization:** The approach presented in [2] attempts to personalize search results that involve building models of user context as ontological profiles by assigning implicitly derived interest scores to existing concepts in domain ontology and maintain the interest scores based on the user's ongoing behavior. This approach demonstrate that the semantic knowledge embedded in an ontology combined with long-term user profiles can be used to effectively tailor search results based on users' interests and preferences. However, changes in user profiles over time needs to be captured in order to ensure the incremental updates to the interest scores accurately reflect changes in user interests.

2.2 Problem Description

30 users including undergraduate and postgraduate students of Government College of Technology, Coimbatore, India performed the task of retrieving documents that satisfy their needs. They were given list of keyword queries to be searched using the typical search engine. Some of the keywords and its intention behind those keyword queries with respect to different users have been given in Table 1.

2.3 Proposed Approach Overview

As the key issue with the abundance of online information is to find relevant web documents, personalization of content is the key to address this issue. This paper presents user profile model which incorporates user intent by analyzing user's previous searches, issued queries and clicked information. It includes the modules as follows. (i) User profile information gathering, (ii) User topic modeling (iii) Matching of user topic model and document topic model to compute personalized relevance score for re-ranking the documents.

Table 1. Diverse interest of search users

Original query	Intention on relevant documents		
	User 1	User 2	User 3
World cup	Web pages mainly dealing with the foodball championship	Web pages mainly dealing with the ICC cricket world cup	Web pages mainly dealing with the T20 cricket world cup
India crisis	Web pages dealing with the economic crisis in India	Web pages dealing with the security crisis in India	Web pages dealing with the job crisis in India
Apple	Web pages on Apple store	Web pages on varieties of apple fruit	Web pages on Apple OS updates and downloads
The ring	Web pages about Ornament	Web pages about the horror movie	Web pages about circus ring show
Okapi	Pages related to animal giraffe	Pages related to okapi African luxury hand bags	Pages related to Information retrieval model BM25

3 Proposed User Topic Modeling for Personalized Search

Statistical language modeling for IR has emerged within the past several years as a new probabilistic framework for describing information retrieval processes. Language Modeling refers to the task of estimating a probability distribution that captures statistical regularities of natural language use. Applied to IR, language modeling refers to the problem of estimating the likelihood that a query and a document could have been generated by the same language model, given the language model of the document and with or without a language model of the query.

3.1 Probabilistic Approach for Personalization

In this work a probabilistic model [7][21][1] is used for predicting the relevance of a document to a specific user with respect to a query. The user representation corresponds to user-specific parameters for part of the model. The formalization assumes that there are only document-specific latent variables (i.e., document features), user specific latent variables (i.e., information need for the query), and combines them to determine whether a document's features satisfy the user's information need. The browsing history of users is obtained from the search logs and user profiles are generated from the browsing behavior.

3.2 User Profile Modeling

Personalization [25] aims to provide users with what they need without requiring them to ask for it explicitly. This means that a personalized IR system must somehow infer what the user requires based on either previous or current interactions with the user. θ_u is defined as a set of terms that the user has come across during the previous and current search and its probability of occurrence in user search session. The User Profile (*UP*) is built with the terms present in users search history. Typically user's search history comprises the queries and documents clicked. The system obtains information on the user this way and infers what are user's needs based on this information. In order to apply the personalization approach, the probability distribution $P(T_u|\theta_u,q)$ as the probability that when issuing a query q, a user u is seeking information on topic T_u. θ_u denotes the user-specific parameters i.e. user profile (*UP*) which possesses *terms* present in user search history. To obtain this conditional distribution, learn a user-independent language model $P(q|T)$ and a user specific prior probability of the topic, $P(T|\theta_u)$, and then apply Bayes' theorem as described in [7].

$$\theta_{\mathrm{u}} = UP_{w_i}$$

$$UP_{w_i \in History(D)} = P(w_i) = \frac{tf_{w_i,D}}{\sum\limits_{w_i \in D} tf_{w_i,D}} \tag{1}$$

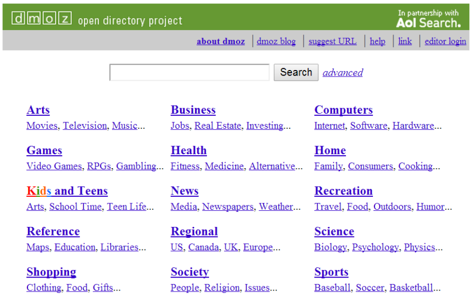
Table 2. Sample user profile representation

w_i	$P(w_i)$
Computer	0.012
Datastructure	0.02
Programming	0.0011
Instruction	0.001
Algorithm	0.032
Analysis	0.004

Topic modeling [6] can be a good choice for IR based problems as low dimensional topical representation can well represent the user search preferences optimistically. Topical modeling using Latent Dirichlet Allocation (LDA) in [35] and Probabilistic Latent Semantic Analysis (pLSA) in [32][3] has been successfully applied. The sample user profile θu shown in Table 2.

3.3 Training for User Interested Topic Identification

The user topic model is trained on the Open Directory Project (ODP) corpus [5] [http://www.dmoz.org/]. In this work, topical categories from the top most level of the ODP are used. ODP screen shot shown in Fig. 1 includes 15 broad categories such as arts, games, home, health, etc. and its sub-categories.


dmoz open directory project
In partnership with Aol Search.
about dmoz | dmoz blog | suggest URL | help | link | editor login
Search
advanced

Arts Movies, Television, Music...	**Business** Jobs, Real Estate, Investing...	**Computers** Internet, Software, Hardware...
Games Video Games, RPGs, Gambling...	**Health** Fitness, Medicine, Alternative...	**Home** Family, Consumers, Cooking...
Kids and Teens Arts, School Time, Teen Life...	**News** Media, Newspapers, Weather...	**Recreation** Travel, Food, Outdoors, Humor...
Reference Maps, Education, Libraries...	**Regional** US, Canada, UK, Europe...	**Science** Biology, Psychology, Physics...
Shopping Clothing, Food, Gifts...	**Society** People, Religion, Issues...	**Sports** Baseball, Soccer, Basketball...

Fig. 1. ODP main page [`http://www.dmoz.org/`]

From the user's search history, it is assumed that a user's click on a document is equated with the observation rel(d,q) = 1 otherwise, rel(d, q) = 0 when there is lack of a click. It is assumed that the user's intended topic T_u is equal to the topic of the document that they click on. Un-clicked pages were assumed as irrelevant to the user in [**7**]. But, it is not fair that treating un-clicked pages as irrelevant because it could be either relevant or irrelevant to the user. The problem of finding user's negative preferences from un-clicked documents that are considered irrelevant to the user has been addressed by exploiting Spy Naïve Bayes (SNB) classifier [36]. The conventional Naïve Bayes requires both positive and negative examples as training data, while SNB require only positive examples.

Algorithm 1. *Relevance_group_user* (document d, query q)

```
V(q) = set of users who have previously searched for q
for each user v
if v has clicked on document d for q
    rel(d,q) = 1
    else
    rel(d,q) = 0
number of users, who find d as relevant for q, $N = \sum rel(d,q)$
```

The probability of relevance obtained from the *relevance function* is biased towards the population of users those who usually search on the query using the search engine. Suppose $V(q) = \{v\}$ be the set of users who have previously searched for query q and whose relevance feedback is used to train the *ranking function*. The probabilistic model explicitly takes these users' intended topics into consideration when interpreting the probability of relevance computed by the ranking function. $rel(d,q)$ is the expected relevance with respect to the distribution of users who typically search for query q across all possible query intents. In order to avoid biasing, the modified $rel(d,q)$ is estimated for each user as per the Algorithm 2.

Algorithm 2. *Relevance_individual_user* (query q, document d, user u)

```
D = set of documents retrieved for query q
for each document d in D
if document d is clicked by u for query q
    rel(d,q) = 1
    else
       rel(d,q) = 0
frequency of q in d
```

In order to learn topics of interest from the users search profile and document topics, the following Algorithms 3 & 4 have been implemented.

Algorithm 3. Training_user_topic(q,d, θ_u)

```
Input: query q in search log, doc d clicked for q by user u, user_profile $\theta_u$)
Output: Topics that are of interest for u
for each query q in user's search history
      for each document d clicked by the user
for each topic T in the ODP category
            compute $T_u$ by $P(T \mid \theta_u, q) = \delta(P(T \mid q)) + (1-\delta)(P(T \mid \theta_u)P(q \mid T)$
return ($T_u$)
```

Algorithm 4. Finding_document_topic(q,D)

```
Input: initial query q, retrieved documents D
Output: Topic of the documents retrieved
     for each document $d_i \in D$ retrieved for the query q
for each topic T in the ODP category
            compute $T_d$ by $P(T \mid d) = P(t_i \in d \mid T)P(T)$
return ($T_d$)
```

In order to compute T_d, choose a topic T according to a multinomial distribution conditioned on document d for each term t_i of document d in the training set. Then generate the word by drawing from a multinomial conditioned on topic T. In this way, documents can have multiple topics. Substituting the values of $rel(d,q)$, distribution of query topics, topic of each document, probability of relevance of a document with respect to a query for a specific user is estimated and thus personalized re-ranking is computed.

3.4 Exploiting User interest profile Model

An essential component of personalized search is learning user's interests. The search history for each user consists of the queries issued, the list of documents in the visible search results, and the list of documents clicked on by the user in response to each query. Since personalization in IR aims at enhancing user's knowledge by incorporating the user preferences and judgment into the retrieval models, the usage of implicit user interests and preferences has been identified so as to enhance current retrieval algorithms and anticipate limitations as World Wide Web (WWW) content keeps increasing, and user expectations keep growing and diversifying. Without requiring further efforts from users, personalization aims to compensate the limitations of user need representation formalisms such as the keyword-based or document-based representations. Thus, User profile is modeled using the initial results set for a given query, Titles of each documents initially presented, extracted terms from full text, extracted terms from snippets, whether URL is clicked previously or not and Dwell time i.e. time spent in each web pages that were clicked already. The goal of user modeling for personalization system is to gain the capability to adapt specific search context of their preferences to better suit their needs.

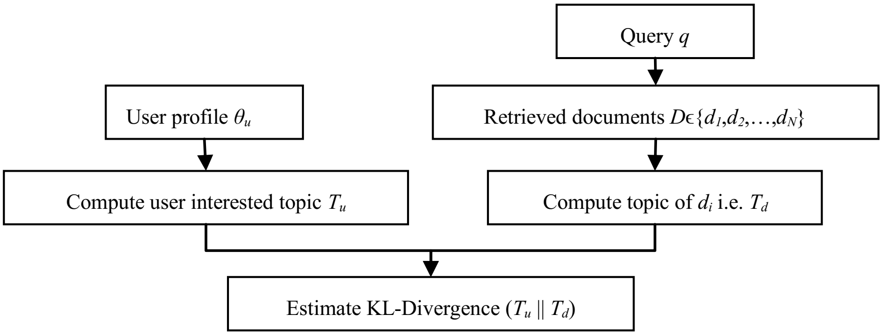

Query q
User profile θ_u
Retrieved documents $D\epsilon\{d_1,d_2,\ldots,d_N\}$
Compute user interested topic T_u
Compute topic of d_i i.e. T_d
Estimate KL-Divergence (T_u || T_d)

Fig. 2. User interested topic (T_u) vs. Document topic (T_d)

A model of information retrieval predicts and identifies what a user will find relevant given the user query. IR Models like Boolean model provides exact matching of documents and query. Vector space models consider the index representations as well as query as vectors embedded in a high dimensional Euclidean space, where each term is assigned a separate dimension. A Language Model (LM) refers to the task of estimating a probability distribution over all possible words in the document i.e. estimating the likelihood that a query and a document could have been generated by the same language model, given the language model of document and query. In this work, Probabilistic model proposed in [7] has been adopted to describe document's content by the topics. Also one more criterion is integrated into the basic retrieval model which is user profile represented by user interested topics. The discrete-valued variables T_d and T_u refer to the document's topic and the topic that the user is searching for, respectively. A document about topic T_d is assumed relevant to a user looking for topic T_u if the following points are met:

(1) Topic T_d satisfies users with information need T_u. Use Kullback-Leibler Divergence (KL-D) [28] between these two contextual models i.e. T_d and T_u to measure the similarity between two contexts. It is unlike exact matching instead matching based on the topicality of the information known already. KL-D metric quantifies the information gain between two probability distributions. It also measures the divergence of probability distribution of a topic in the document (T_d) to its distribution in the user interested topics (T_u). The lesser the divergence from T_u, the more informative the topic is for document T_d. The KL-D value of a topic t in document and user interest is given as shown in Eq. (2).

$$KL - D(T_d \parallel T_u) = \sum_{t \in D \cap U} P(T_d(t)) \log \frac{P(T_d(t))}{P(T_u(t))} \tag{2}$$

where $P(T_d(t))$ is the probability of topic t in document topic model T_d and $P(T_u(t))$ is the probability of topic t in user topic model T_u.

(2) Given that the document's topic matches that of the user's search intent i.e. user's topic, the document is relevant to the query based on Eq. (3).

$$P(Q \mid T_u, T_d) = \prod_{q_i \in Q} (\alpha P(q_i \mid T_u) + (1-\alpha) P(q_i \mid T_d)) \tag{3}$$

where, α is a weighting parameter that lies between 0 and 1, $P(q_i|T_u)$ is the probability of the word in the user interested topic model i.e. the user independent query topic model learned, $P(q_i|T_d)$ is the probability of the word in the document topic model.

***Proof**: KL Divergence (KL-D) of two documents*

Let $P(T_d(t))$ and $P(T_u(t))$ be two probability distributions of a discrete random variable. The KL-D is only defined if $P(T_d(t))$ and $P(T_u(t))$ both sum to 1 and if $P(T_u(t)) > 0$ for any t such that $P(T_d(t)) > 0$.

In order to compute KL-D, a document d is observed as discrete distribution of $|d|$ random variables, where $|d|$ is the number of words in the document. Let d_1 and d_2 be two documents for which we want to calculate their KL-divergence. It is run into two problems:

- Compute the KL-divergence twice due to asymmetry: KL-D($T_d \| T_u$) and KL-D($T_u \| T_d$).
- Due to the constraint for defining KL-D, calculations must only consider words occurring in both d_1 and d_2.

3.5 Personalized Re-ranking Process

Typically, there are two variants of user context information [37] to model user's search experience. Firstly, the short-term context which emphasizes that the most recent search is most directly close to the user's current information need. Successive searches in a session usually have the same context. However, detecting a session is a difficult task. Secondly, the long-term context which assumes that user will have their interests over a long time. It means that the past search may have some impact on current search. Re-ranking of the results reflects the most relevant results for the user. It is a process of re-ordering the retrieved results based on combination of short-term and long-term user preferences. Re-ranking computation performs the following two processes. They are,

- Calculating personalized score for document
- Generating personalized result set

3.5.1 Personalized Score

The personalized relevance score for each document *d* for a query *q* is computed for a user *u* who issued query *q* as follows:

- Compute the topic T_u of user interest
- Retrieve all the documents $d_{1,}$ $d_{2,}$......., d_N for query *q* from a traditional search engine
- Compute the topic T_d of each document

The conditional distribution $P(T|d)$ specifies the topic of each document. This distribution could be estimated using techniques described in [36][32][3] in order to predict the ODP category for each web document. Typically, for the given query *q* and the user profile θ_u, in order to find the relevant documents, the documents are re-ranked by $P(D|Q,\theta_u)$ using Bayes' theorem as shown in Eq. (4).

$$P(D \mid Q,\theta_u) = \frac{P(D \mid \theta_u)P(Q \mid D,\theta_u)}{P(Q \mid \theta_u)} \tag{4}$$

The personalized score is computed for each user incorporating his/her interest and preferences. Let *D* be the set of documents returned by the search engine. The rank of each document *D* returned for a query *Q* for user *u* is computed by integrating topic model and user model as shown in Eq. (5).

$$P(Q \mid D,\theta_u) = P(Q \mid T_d, T_u) + \prod_{q_i \in Q} (\beta P(q_i \mid \theta_u) + (1-\beta) P(q_i \mid D)) \tag{5}$$

where, β is a weighting parameter that lies between 0 and 1, $P(q_i|\theta_u)$ is the probability of the word from the user interest profile model, $P(q_i|D)$ is the probability of the word from the documents retrieved i.e. document model.

3.5.2 Personalized Result

Search engines always return millions of search results and thus it is essential to re-order results to facilitate users to find documents what they want. Re-ordering web search results is assumed as an application of user interest modeling. The initial documents retrieved for the query by the search engine can be re-ranked according to the personalized score computed. The documents are then scored based on the probability $P(Q|D,\theta_u)$ and arranged based on descending order of the personalized score.

4 Experimental Setup

4.1 Dataset Description

The dataset used in this work is similar to AOL search log [12] which possesses implicit feedback in the form of click-through data collected by a search engine and it is released for research purpose. The AOL Search Data is a collection of real query log data that is based on real users. The data set consists of 20M web queries collected from 650k users over the period of three months.

Table 3. Statistics about AOL search log dataset

Number of lines of data	36,389,567
Number of instances of new queries (with or without click-through data)	21,011,340
Number of requests for “next page” of results	7,887,022
Number of user click-through data	19,442,629
Number of queries without user click-through data	16,946,938
Number of unique queries	10,154,742
Number of users log	657,426

4.2 Baseline Approaches

The following approaches have been considered for comparing the proposed personalization model to assess the performance improvements.

Best Matching 25 (BM25). BM25 (Best Matching) [31] is a bag-of-words retrieval function that ranks a set of documents based on the query terms appearing in each document, regardless of the inter-relationship between the query terms within a document. Given a query Q, containing keywords q_1, q_2,…,q_n, the BM25 probabilistic ranking function of a document D is computed as shown in Eq. (6).

$$score(D,Q) = \sum_{i=1}^{n} IDF(q_i).\frac{tf_i.(k_1+1)}{tf_i + k_1.\left(1-b+b.\frac{|D|}{avgdl}\right)} \quad (6)$$

where, tf_i is q_i’s term frequency in document D, $|D|$ is the length of the document D in terms of number of words, $avgdl$ is the average document length in the dataset. k_1 and b are tuning parameters. k_1 has little effect on retrieval performance, b is a document length normalization parameter to be ranging within 0-1.

$$IDF(q_i) = \log \frac{N - df_i + 0.5}{df_i + 0.5} \tag{7}$$

where, *N* is the total number of documents in the dataset and df_i is the number of documents containing the query q_i.

Rocchio Algorithm. This approach uses relevance feedback [14] to improve retrieval performance. Rocchio algorithm incrementally modifies the query by adding terms from the explicit relevance feedback. The typical Rocchio approach has been modified for comparison for work in personalized search. In order to construct the user profile, past queries Q_n entered by the user and its associated clicks as relevance feedback RF_n has been used. Thus, terms from $RF_{1...n}$ is extracted to compute their frequencies, which in turn represented as $\{(w_1, tf_{w_1}), (w_2, tf_{w_2}), \ldots, (w_i, tf_{w_i}), \ldots, (w_n, tf_{w_n})\}$ where w_i is a word in RF_n and tf_{w_i} is the frequency of occurrence of w_i. Then, the set of documents *D* returned by the search engine for a query *Q* is re-ranked by the personalized score computed as shown in Eq. (8).

$$Score(D, Q) = \sum_{w \in Q} \left(\frac{tf_{w,Q}}{|Q|} + \frac{tf_{w,UP}}{|UP|} \right) . \frac{tf_{w,D}}{|D|} \tag{8}$$

where $tf_{w_i,Q}$ is the term frequency of word w_i in query Q, $tf_{w_i,UP}$ is the term frequency of w_i in user profile *UP*, $tf_{w_i,D}$ is the term frequency of w_i in document *D*, and |*Q*|, |*UP*|,|*D*| are the length of the query i.e. number of words in *Q*, length of the user profile i.e. number of words in *UP*, length of the document i.e. number of words in *D* respectively. Thus, the documents retrieved for the initial query are then re-ordered based on the score computed.

Document Language Model (LM) Approach. In exploiting LM [37][19], the user profile is learned by collecting words and their probabilities from the implicit relevance feedback for all the training queries. In order to re-rank the retrieved documents, retrieve top few results from the traditional search engine and then re-order them based on the score computed as given in Eq. (9).

$$P(Q \mid D, UP) = \prod_{q_i \in Q} (\alpha P(q_i \mid UP) + (1 - \alpha) P(q_i \mid D)) \tag{9}$$

where $P(q_i|UP)$ is the probability of the word q_i in user profile, $P(q_i|D)$ is the probability of the word in the document and α is a weight tuning parameter which takes values between 0 and 1.

Query Language Model Approach. In this approach, the probability of the word in user profile is smoothed with a general LM estimated from a large number of queries from the query log as given in Eq. (10).

$$P(q_i \mid UP) = \beta P(q_i \mid UP) + (1 - \beta) P(q_i \mid QueryLog) \tag{10}$$

where $P(q_i\ |QueryLog)$ is the probability of the word q_i in the search query log and β is a weight tuning parameter which takes values between 0 and 1.

4.3 Evaluation Metrics

The re-ranking algorithms proposed in this work have been evaluated using a variety of accepted IR metrics [4][27].

- **Precision:** This measures the accuracy of the retrieved results. Precision defines the fraction of retrieved documents that are labeled as relevant i.e. documents ranked in the top n results that are found to be relevant. If the documents within the top *k* are irrelevant, then this measures the user satisfaction with the top *k* results.

$$P@k = \frac{\#of\_relevant\_doc\_retrieved\_among\_k}{k} \tag{11}$$

- **Recall:** This measures the coverage of the relevant documents in the retrieved results. Recall defines the fraction of relevant documents that are retrieved.

$$R@k = \frac{\#of\_relevant\_doc\_retrieved\_among\_k}{total\#of\_relevant\_documents} \tag{12}$$

- **Interpolated precision:** The interpolated precision ($P_{Interpolated}$) at a certain recall level *r* is defined as the highest precision found for any recall level $r' \geq r$.

$$P_{Interpolated}(r) = \max_{r' \geq r} P(r') \tag{13}$$

- **Mean Reciprocal Rank (MRR):** The mean reciprocal rank is the average of the reciprocal ranks of results for a sample of queries *Q*. Consider the rank position *k* of first relevant document, Reciprocal Rank score =1/*k*. MRR is the mean of reciprocal rank across multiple queries given by Eq. (14).

$$MRR = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{rank_i} \tag{14}$$

- **Normalized Discounted Cumulative Gain (NDCG):** Precision and Mean Average Precision can only handle cases with binary judgment i.e. relevant or irrelevant. To measure the ranking quality accurately, Discounted Cumulative Gain (DCG) [18] has been used. DCG is a measure that gives more weight to higher ranked documents by discounting the gain values for lower ranked documents. This measure the usefulness of a retrieved document based on its position in the result list. The ranked results are examined from top ranked results to lower for a given query. The highly relevant documents appearing lower in search result list will be penalized by reducing relevance value r_i logarithmically proportional to the position of the result. The DCG accumulated at a particular rank position *N* is defined as given in Eq. (15).

$$DCG_K = \sum_{i=1}^{K} \frac{2^{r_i} - 1}{\log_2(i+1)} \tag{15}$$

where r_i is an integer relevance label (0= “Bad” and 5= “Perfect”) of result returned at position *i*. “Bad” documents do not contribute to the sum, thus will reduce DCG for the query pushing down the relevant labeled documents, reducing their contributions. Since search result lists vary in length depending on the query issued to the search engine, the results of one query cannot be consistently compared with the results returned to another query using DCG measure. In order to normalize DCG, sort documents of a result list by the order of relevance to produce the maximum possible DCG till the position *K* i.e. termed as ideal DCG (IDCG). The normalized DCG is computed using Eq. (16).

$$NDCG_K = \frac{DCG_K}{IDCG_K} \tag{16}$$

5 Result Analysis

5.1 Experimental Design

The data for each user consists of queries and their corresponding clicked URLs. The training data is used for learning user profile and testing data is used for evaluating the approaches.

Training: The search results returned by the traditional search engine for the past queries are collected. The corresponding top documents/snippets from the search engine are extracted using a plug-in and then used for learning user profile.

Testing: The top *N* results returned by the traditional search engine are collected. These results are then passed to the personalized re-ranking process. This process then re-scores the results and returns the re-ranked results. The top *N* results from the re-ranked results are compared with the relevance judgment. Thus, the performance of the re-ranking approach is evaluated.

The AOL dataset contains only the URL of clicked documents. It does not contain the actual document content. Hence, the real dataset has been generated and used for the experimental evaluation. The experiments were carried out on a dataset consisting of 50 users. Each user was asked to submit number of queries to traditional search engine. For each query, they were further asked to examine the top 20 results in order to identify the set of relevant documents.

Table 4 gives the sample statistics of the real dataset of 10 users. The document collection consists of top 20 documents for each query from the search engine which is typically assumed to be assessed by the user while retrieving the relevance of the documents. The total size of the document collection is 14265 documents excluding ‘.ppt’, ‘.pdf’ and ‘.doc’ formats. The commercial search engine is used in order to retrieve initial set of results matching the query. The proposed implementation is simulated using Lucene based IR system (http://lucene.apache.org/).

In order to construct the user profile, divide the search history of each user into two groups in such a way that each of which possesses equal number of search queries. For example, User 1 searched for 43 different queries, divide his/her search log into 2 sets with 21 queries each approximately. Then, user profile is learned for first set. The second set is used for testing purpose.

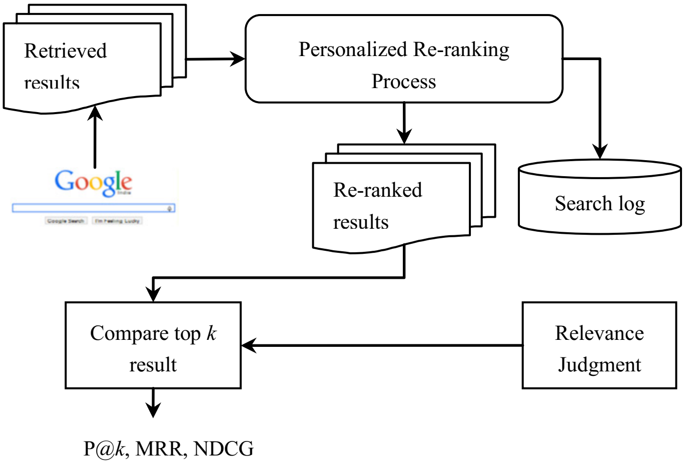


Fig. 3. Experimental setup for evaluation

In order to learn the user interest profile, the past search queries issued and its corresponding relevant documents are assumed. Performance is calculated during testing. In testing, user was asked to enter the query. Subsequently, re-ranked set of documents is generated using the proposed approach using user profile learned from first set of queries. Accordingly, precision at top *k*, MRR and NDCG are measured to show the performance improvement over baseline systems.

Table 4. Sample real dataset statistics (10 Users)

User	# of Queries	Total # of relevant documents	Average # of relevant documents
User 1	43	225	5.23
User 2	39	125	3.21
User 3	63	295	4.68
User 4	62	188	3.03
User 5	37	190	5.14
User 6	28	91	3.25
User 7	45	173	3.84
User 8	31	96	3.10
User 9	51	240	4.71
User 10	39	128	3.28

5.2 Parameter Tuning

The performance of the proposed method is sensitive to the choice of α and β parameters. These parameters needs to be tuned as there are two ranking combination schemes shown in Eq. (3) and Eq. (5) to be used. The parameter α determines the weight of the user interested topic model (T_u) and document topic model (T_d) while β parameter determines the weight of the user interest profile model (θ_u) and document model (D). Taking the top 10 search results as an instance, we give a range of values for α and β and compare the relative improvement in Precision, MRR, and NDCG. We compare the two ranking combination schemes and the results are shown in Fig. 4 and Fig. 5 respectively. With regard to the proposed scheme, as long as α and β value is big enough, the improvement in IR measures stay around the maximum value without much diminishing change. Although the optimum value of α and β is hard to formulate, the empirical results show that if we simply re-rank totally by user interest profile model (θ_u) and user interested topic model (T_u), the improvement in Precision, MRR, and NDCG is very close to the maximum value that can be achieved.

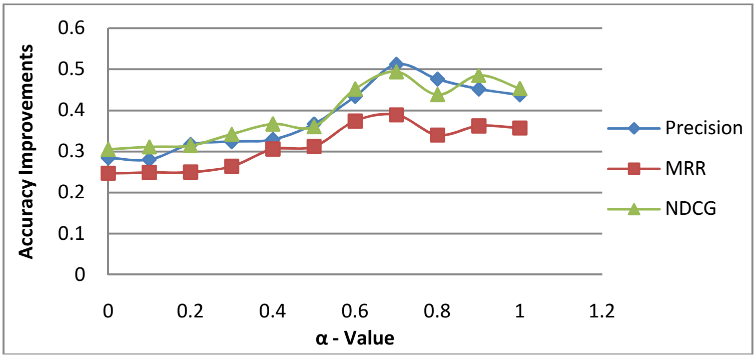


Fig. 4. α Parameter tuning on Eq. (3) for 5 queries at top 10 search results

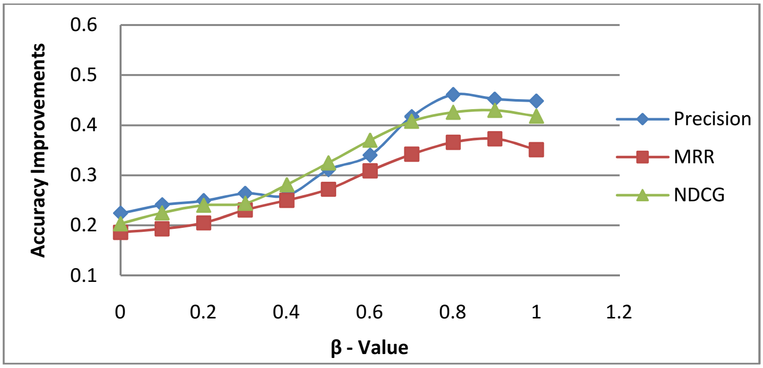


Fig. 5. β Parameter tuning on Eq. (5) for 5 queries at top 10 search results

From the Fig. 4 and Fig. 5, it is noted that the results of the combination of different models is getting better when the values of α and β is sufficiently large. Initially β-value was set 0.5 while tuning the α-value and then α-value was set 0.7 while tuning the β-value. Thus, it is implied that α and β values 0.7 and 0.85 respectively yields better accuracy on an average.

5.3 Evaluation on Real Dataset

Experiments have shown that the proposed personalization approach (T_U) achieve better results over baseline methods. The rich representation of user interest model served as a fine user search context model which capture user search goal accurately. Thus, bring the relevant documents in the first few results.

Table 5. MRR and Precision at top-*k* results for 10 queries

Method	MRR@5	P@5	P@10
Best Matching (BM25)	0.239	0.3607	0.2914
Rocchio algorithm	0.305	0.4322	0.3783
Document Language Model approach (LM_D)	0.332	0.473	0.4145
Query Language Model approach (LM_Q)	0.371	0.5118	0.447
Proposed integrated topic model and user model approach (T_U)	0.428	0.5605	0.4926

Table 5 shows the MRR obtained at top 5 results and Precision obtained at k ϵ 5, 10 for 10 different queries. It is observed that the performance of the proposed T_U is found to be better compared to that of the baseline systems. Fig. 6 shows the precision - recall curve obtained for 10 different queries. The performance improvement has been observed in terms of accuracy and coverage of retrieved results.

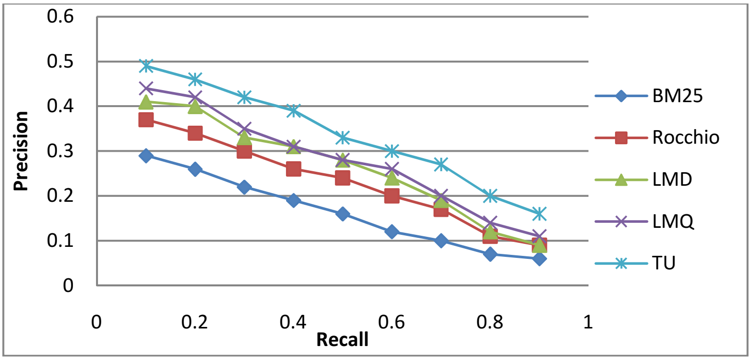


Fig. 6. Precision Vs. Recall obtained for 10 queries

Fig. 7 shows NDCG obtained at k ϵ 1, 2, 3,..., 10 for 10 different queries. It is observed that the performance of the proposed TU is found to be consistently better compared to that of the baseline systems. It is observed that the proposed method shows performance improvement over baseline methods on real dataset by bringing the highly relevant documents in first few results. Thus, increased NDCG values at first 10 results means that the proposed method ranks documents appropriately incorporating user search context in order to satisfy users with relevant documents.

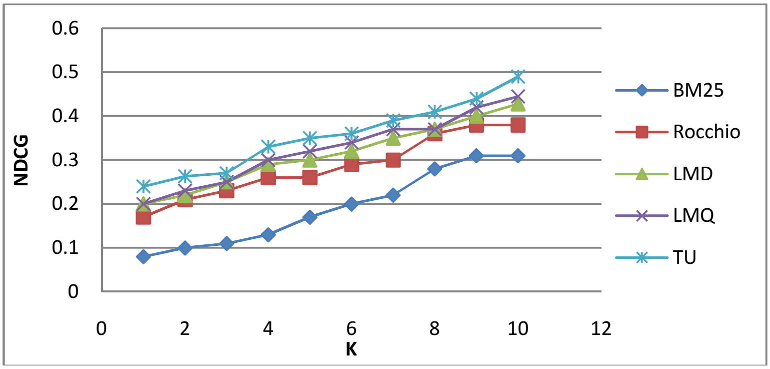


Fig. 7. NDCG at K obtained for 10 queries

6 Conclusion

Personalization has been performed at client side by re-ranking the results returned by the traditional search engine. The distribution of user topics and probability distribution of document topics have been estimated to calculate the personalized score in order to observe the relevance of documents. The documents are then re-ranked according to the score obtained. There is merit in locally generating a topic model, and then locally filtering and re-ranking search results, as this approach can work even when cookies are not accepted or deleted by the browser to maintain privacy. However, it is inferred that still there is a gap existing in the process of user profile information capture and representations.

Acknowledgements. The work presented in this paper is supported and funded by the Department of Science and Technology (DST), Ministry of Science and Technology, Government of India under INSPIRE scheme. Authors wish to extend their thanks to DST.

References

1. Berger, A., Lafferty, J.: Information retrieval as statistical translation. In Proc. SIGIR, pp. 222–229. ACM (1999)
2. Sieg, A., Mobasher, B., Burke, R.: Web search personalization with ontological user profiles. In: Proc. CIKM, pp. 525–534. ACM (2007)

3. Lin, C., Xue, G.-R., Zeng, H.-J., Yu, Y.: Using Probabilistic Latent Semantic Analysis for Personalized Web Search. In: Zhang, Y., Tanaka, K., Yu, J.X., Wang, S., Li, M. (eds.) APWeb 2005. LNCS, vol. 3399, pp. 707–717. Springer, Heidelberg (2005)
4. Manning, C.D., Raghavan, P., Schutze, H.: Introduction to Information Retrieval. Book published by Cambridge University Press (2008)
5. Carpineto, C., Romano, G.: ODP239 dataset (2009), http://credo.fub.it/odp239/
6. Blei, D.M., Lafferty, J.D.: Topic Models. Technical Report, Princeton University (2009)
7. Sontag, D., Collins-Thompson, K., Bennett, P.N., White, R.W., Dumais, S., Billerbeck, B.: Probabilistic Models for Personalizing Web Search. In: Proc. WSDM, pp. 433–442 (2012)
8. Agichtein, E., Brill, E., Dumais, S.: Improving Web Search Ranking by ncorporating user behavior information. In: Proc. SIGIR, pp. 19–26. ACM (2006)
9. Qiu, F., Cho, J.: Automatic identification of user interest for personalized search. In: Proc. 15th Intl. Conf. on World Wide Web, pp. 727–736. ACM (2006)
10. Radlinski, F., Dumais, S.: Improving Personalized Web Search using result diversification. In: Proc. SIGIR, pp. 691–692. ACM (2006)
11. Bordogna, G., Campi, A., Psaila, G., Ronchi, S.: Disambiguated Query Suggestions and Personalized Content-Similarity and Novelty Ranking of Clustered Results to Optimize Web Searches. Elsevier - Information Processing and Management (48), 1067–1077 (2012)
12. Pass, G., Chowdhury, A., Torgeson, C.: A Picture of Search. In: Proc. 1st Intl. Conf. on Scalable Information Systems (2006)
13. Xue, G.-R., Zeng, H.-J., Chen, Z., Yu, Y., Ma, W.-Y., Xi, W., Fan, W.: Optimizing web search using web click-through data. In: Proc. CIKM, pp. 118–126. ACM (2004)
14. Rocchio, J.J.: Relevance feedback in information retrieval. In: Proc. The smart retrieval system - Experiments in Automatic Document Processing, pp. 313–323 (1971)
15. Teevan, J., Morris, M.R., Bush, S.: Discovering and using groups to improve personalized search. In: Proc. WSDM, pp. 15–24. ACM (2009)
16. Teevan, J., Dumais, S.T., Liebling, D.J.: To Personalize or Not to Personalize: Modeling Queries with Variation in User Intent. In: Proc. SIGIR, pp. 163–170. ACM (2008)
17. Teevan, J., Dumais, S.T., Horvitz, E.: Beyond the Commons: Investigating the Value of Personalizing Web Search. In: Proc. Workshop New Technologies for Personalized Information Access (PIA), pp. 84–92 (2005)
18. Kalervo, J., Jaana, K.: Cumulated Gain-Based Evaluation of IR Techniques. ACM Transactions on Information Systems (2002)
19. Ponte, J.M., Croft, W.B.: A language modeling approach to information retrieval. In: Proc. SIGIR, pp. 275–281. ACM (1998)
20. Hu, J., Chan, P.K.: Personalized Web Search by Using Learned User profiles in re-ranking. In: Proc. SIGKDD, pp. 1–14. ACM (2008)
21. Gao, J., He, X., Nie, J.-Y.: Clickthrough-Based Translation Models for Web Search: from Word Models to Phrase Models. In: Proc. CIKM, pp. 1139–1148. ACM (2010)
22. Sugiyama, K., Hatano, K., Yoshikawa, M.: Adaptive Web Search Based on User Profile Constructed without any Effort from Users. In: Proc. 13th Intl. Conf. on World Wide Web, pp. 675–684. ACM (2004)
23. Daoud, M., Tamine-Lechani, L., Boughanem, M.: Learning user interests for a session-based personalized Search. In: Proc. 2nd Intl. Symposium on Information Interaction in Context, pp. 57–64. ACM (2008)
24. Matthijs, N., Radlinski, F.: Personalizing Web Search using Long Term Browsing History. In: Proc. WSDM, pp. 25–34. ACM (2011)

25. Agrawal, R., Gollapudi, S.: Diversifying Search Results. In: Proc. WSDM, pp. 5–14. ACM (2009)
26. Krestel, R., Fankhauser, P.: Reranking web search results for diversity. Springer Information Retrieval (15), 458–477 (2012)
27. Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval. Addison Wesley (1999)
28. White, R.W., Chu, W., Hassan, A., He, X., Song, Y., Wang, H.: Enhancing personalized search by mining and modeling task behavior. In: Proc. WWW, pp. 1411–1420. ACM (2013)
29. Anand, S.S., Mobasher, B.: Intelligent techniques for web personalization. In: Mobasher, B., Anand, S.S. (eds.) ITWP 2003. LNCS (LNAI), vol. 3169, pp. 1–36. Springer, Heidelberg (2005)
30. Stamou, S., Ntoulas, A.: Search Personalization through Query and Page Topical Analysis. Proc. User Model User-adapt Interact 19(1-2), 5–33 (2009)
31. Robertson, S.E., Walker, S., Hancock-Beaulieu, M., Gull, A., Lau, M.: Okapi at TREC. In: Proc. Text Retrieval Conference, pp. 21–30 (1992)
32. Hofmann, T.: Probabilistic Latent Semantic Indexing. In: Proc. SIGIR, pp. 50–57. ACM (1999)
33. V.: K, M. Simon. Collaborative Filtering for Sharing the Concept Based User Profiles. In: Proc. of 3rd IEEE International Conference on Electronics Computer Technology (ICECT), vol. 4, pp. 187–191 (2011)
34. Veningston, K., Shanmugalakshmi, R.: Enhancing personalized web search re-ranking algorithm by incorporating user profile. In: Proc. of 3rd IEEE International Conference on Computing Communication and Networking Technologies (ICCCNT), pp. 1–6 (2012)
35. Wei, X., Bruce, C.W.: LDA-based document models for ad-hoc retrieval. In: Proc. SIGIR, pp. 178–185. ACM (2006)
36. Ng, W., Deng, L., Lee, D.L.: Mining User Preference Using Spy Voting for Search Engine Personalization. ACM Trans. Internet Technology 7(4), article 19 (2007)
37. Shen, X., Tan, B., Zhai, C.: Implicit User Modeling for Personalized Search. In: Proc. CIKM, pp. 824–831. ACM (2005)
38. Dou, Z., Song, R., Wen, J.-R., Yuan, X.: Evaluating the Effectiveness of Personalized Web Search. IEEE Trans. Knowledge and Data Engineering 21(8), 1178–1190 (2009)