mSVM Clustering with IABC Approach for Query Based Recommendation System

Prof.G.Sivakumar[1], Dr.K.M. Subramanian[2]
[1,2]Associate Professor/CSE, Erode Sengunthar Engineering College, Thudupathi, Erode

Abstract---*Query recommendation is the mostpreferred method used for enhancing the usability of Web search engines. The proposed work aims to generate an effective Query based recommendation system by using datamining techniques along with a collection of similar queries triggered at any time and also by way of formulating future queries for multiple users. A modified Support Vector Machine (mSVM) clustering and swarm intelligence based profile management and recommendation system is proposed. Initially, the mSVM clustering generates cluster results of user's profile in accordance with the query logs. Subsequently, the Improved Artificial Bee Colony (IABC) algorithm is employed for nearest neighbour exploration to discover the recommendation queries with respect to the cluster results. Extraordinary mSVM clustering based query recommendation system is proposed with IABC for several search engine personalization functions like query suggestion at time of hitting, query recommendation to formulate the upcoming queries and offer the effective search result in accordance with the real intention of the user and moreover re-rank the listed queries and search result. Extensive experiments were carried out using mSVM-IABC approach on large-scale search logs obtained from a commercial search engine and then compared with existing approaches. Results indicated that the mSVM-IABC approaches considerably outperforms the same in terms of prediction accuracy and it is effective and has an inherent realistic approach with respect to Web query recommendation.*

Keywords: *Webmining, Profile management, Clustering, Swarm Intelligence, modified Support Vector Machine, Improved Artificial Bee Colony Optimization, query log*

1. Introduction

Search engine has avery important responsibility during the information retrieval process wherein it offers requested information in accordance with the query keywords in terms of web snippets. The result provided by the search engine is not appropriate every time, from time to time inappropriate URLs are also listed to the user as a result of smaller and unclear query keywords. In general, the search engine users are inexperienced and simply formulate their query keywords in front of the search engines. They possess less background knowledge in lieu of information requirements. In addition, the input queries are smaller and unclear (Wen et al., 2001). The smaller length queries do not generate correct results; hence the query recommendations are an indispensable method that offer suggestions to the user to formulate their queries in the future. Query recommendation assists in describing the user's information requirements more clearly. This is done so that search engines can list significant and proper answers. Contemporary researches establish that the investigation of query log and the use of users' behaviour information facilitate to get better query recommendation performance (Baeza-Yates et al., 2007) (Baeza-Yates et al.,2005). The real search target of the user is investigated and listed not only by using the query logs (Grimes et al.,2007)but also by using the clicked concepts from the web snippets ("Web-snippet" indicates the title, abstract, and URL of a Web page listed by the search engines) and the user's preferences in the search results.

NPD 2000 (Hsieh-Yee and Ingrid, 2001) states that an independent investigation of 40000 web users found that subsequent to a failed search, 76% of users attempt to rephrase their query in the same search engine. This is one state wherein the user can choose the listed queries. At some point in the search process, information regarding the user is accumulated in two different manners (Speretta and Gauch, 2005) (Wen et al., 2002). Implicit profile is automatically generated by means of the search behaviour of the user from the query log. Explicit profile is generated by the user by giving objective feedback. The foremost drawback of the query recommendation process with respect to the implicit (Zigoris and Zhang, 2006), (Sugiyama et al., 2004) user's feedback is that it is not feasible to perfectly discover the user's real search target. Google Personalized Search constructs a user profile by the way of implicit feedback in which the system adjusts the results in accordance with the search record of the user. The user will not offer the entire information openly and often. The search engine's query recommendation is personalized by means of the information regarding the users in terms of the concept depending on user profile. Information can be obtained from the users in two ways: either explicitly, like requesting opinion such as preferences or evaluations; or implicitly, like observing user behaviours for instance the time exhausted reading an on-line document, amount of times an URL is clicked and etc.

Explicit building of user profiles has certain disadvantages (Joachims et al., 2005). The users

possibly will offer conflicting or inaccurate data, the profile is static while the user's interests are more likely to change in due course, and the building of the profile can have possible implications on the user wherein the user might not desire to offer all the information recursively. In contrast, implicitly generated user profiles do not have any implications on the user as such. As a result, several research (Chen et al., 2002)(Claypool et al., 2001)generated user profiles implicitly and provided recommendations. At present, several commercial search engines and lots of research work concentrate on how to suggest queries based upon users' earlier query and click actions. The concept is to find most common queries which are comparable with the existing query either in content (Baeza-Yates et al., 2004)(Baeza-Yates and Tiberi, 2007) or in click context (Cucerzan and Ryen ,2007) (Fonseca et al., 2003). This manner of recommendation does not comprehensively provide an understanding of users' definite information requirements. It does not consider current users' search intention into consideration; rather, it employs collaborative recommendation that shares comparable interests with other users who recommend similar queries. However, this approach recommends the queries by means of content-based profiles and also collaborative profiles.

The two snippet click schemes, specifically, global scale snippet click scheme and a local scale snippet click scheme and subsequent recommendation algorithms are described in (Liu et al., 2011). Rather than discovering the comparable keywords from the query log, the real user's information requirements are analysed by retrieving the ideas from clicked snippets. However, the proposed user profile suggests queries and in addition re-ranks the recommended queries in accordance with the intention of the user. This approach produces the concept depending on the user profile based upon certain constraints like:

- User preferences specified explicitly in the log file
- Clicked snippets demonstrates the user's intention of the current query and
- Past queries and its click thru from the query log.

In this paper, Hybrid User profile is produced in accordance with mSVM-IABC which is employed for several search engine personalization functions. These include query suggestion at time of hitting, query recommendation to formulate the upcoming queries and offering effective search results in accordance with the real intention of the user and re-ranking listed queries and search result. The input of the query recommendation development can be in the form of query log, user profile or an outside source like web pages, ontology, etc. The recommendation might be given ahead of querying, at the time of querying or subsequent to querying. The rest of the paper is organised as follows: Section 2 defines the Recommendation Technique based on the concept based user profile using mSVM-IABC. Here, mSVM is used to frame the user profiles and IABC is employed to produce the query recommendation; Section 3 discusses the experiments and results. Finally Section 4 forms the conclusion of the paper.

2. Query based Recommendation System

Query recommendation is a potential way for enhancing usability of Web search engines. In case if a user types a query in the search textbox and clicks submit, the query is being processed in which the subsequent tasks are performed, similarity checking and semantic meaning among the user queries is carried out to produce query suggestions. This scheme suggests a recommendation method that depends on the postulation that the system will offer improved results than the approaches that consider keywords similarity for suggesting related queries. The reason for the improved results is that it completely based on the investigation of query logs that attempt to expand the queries with the keywords associated with the user query and the cluster formed subsequently by means of query logs. Additionally, the results are optimized by completely taking care of the semantic behaviour. In this section, mSVM clustering is employed for the clustering process and the improved ABC approach is employed for forming recommendations to the user in accordance with the cluster results. The overall work is shown in Fig.1.

2.1. Profile Management using mSVM clustering

Web Usage Mining (WUM) turns out to be an indispensable tool for electronic promotion and it is extensively employed studybehavior of users. The queries can be obtained through sources like web server and proxy server. The navigation patterns of users are extremely significant for website holders. As this helps them to enhance the way of presenting information and additionally promote more users to the website. In order to find out the navigation pattern, the data mining approaches are extensively used. Especially, the clustering approach is employed to supervise the profiles based on user's query data. In this work, the Query Cluster is constructed using mSVM clustering and it holds a collection of similar queries triggered at any time which are employed for recommending and formulating future queries for several users.

This work integrates related queries that take place at any time period depending on the similarity of query keywords, clicked URLs and the perceptions based on top-k dominating queries by means of Iterative Top-k Dominating (ITD) algorithm. Consider there are a collection of groups

and each every one of them holds a collection of attributes, and each is related with a ranked listing of tuples, ID and score. The entire lists are ranked in decreasing order of the scores of tuples. Here, the foremost interest is to discover the best mixtures of attributes, each combination concerning one attribute from every group. Furthermore, specifically, it is desired to obtain the top-k combinations of attributes in accordance with the equivalent top-m tuples with matching IDs which is described in (Yiu and Mamoulis, 2007). Table 1 demonstrates the pattern for clicked documents with concept. The user examines the search result from the top to the bottom and makes a decision to click the documents. This is because search engines provide search information to the user as references in the form of query logs. Query log is a vital repository, which traces the user's search navigational behaviours.

The mining of these logs can considerably enhance the working of the search engines. With the aim of providing recommendations to formulate future queries, the search records in the query logs are investigated. The search records are categorized under the attributes:

<AnonID, Query, QueryTime, ItemRank, ClickURL, Concepts>

Table 2 gives an overview about the attributes and their descriptions employed in the data set. The entries in the query log are examined. Initially, the users and their sessions are recognized and the user's favourite query is produced (Umagandhi and Senthil, 2013). Smilar users are investigated and clustered by means of Agglomerative clustering algorithm (Beeferman and Berger, 2000). The query keywords from similar users are also provided as recommendations. On the other hand, in the case of agglomerative clustering there is no condition that may be deployed forrepositioning of objects that might have been 'incorrectly' clustered at an early phase. The result should be observed very closely to ensure that they are logical. As a result in this work, the mSVM clustering is employed to obtain the better results.

Concepts Extraction

The search engine reacts to the user's query Q in terms of web snippets. When a keyword appears a number of times in the top documents of the web snippets, subsequently an imperative concept regarding that query also appears. The major reappearance of a keyword in the top retrieved documents of the web snippets appends sufficient implication to the concept corresponding with the specific query. The user examines the retrieved web snippets from the top to the bottom (Joachims, 2002) and subsequently decides which one among the documents is appropriate and then clicks it. The significant ideas from the clicked documents are retrieved and accumulated in the concept log. The obtained concepts are pre-processed in the way of the following,

- The concepts are transformed into lowercase letters.
- Additional spaces are trimmed.
- The entire plurals are changed into singulars. (It is known as Lemmatization. Morpha is employed for the transformation. It can be downloaded from the following link www.informatics.sussex.ac.uk /research) Stemming and stop word removal. (Eliminate the words like cached, similar and etc. and also any symbols like @, ., ; and etc.

When a concept appears regularly on the Web-snippets for a specific query, it is regarded as an important concept related to that query. The support value is employed to discover the level of interest of a concept. In the process of concept extraction, initially find out the support value of the distinctive concepts of length one. When the support value meets the minimum support threshold, subsequently the concept with higher length is produced. The commonly occurred concepts in the clicked web snippets are recognized with the help of support formula (Leung and Dik, 2010) given in equation (1),

$$Support(C_i) = \left(\frac{sf(c_i)}{n}\right).|c_i| \quad (1)$$

Where $sf(c_i)$ represents snippet frequency, number of clicked web snippets includes the concept C_i, n indicates the total number of clicked web snippets. Support of a concept C_i is higher than the threshold s, subsequently C_i is an imperative concept associated with the query q. Here, the concept C_i is a significant one, when it appearsa minimum of 50% in the clicked documents. The support value is computed only for the concepts in the clicked snippets; when it meets the threshold s then it is considered as an important concept as a positive preference. Determining the maximum length of a concept is restricted to seven words, as it affects the computational time and in addition avoids extracting meaningless concepts. The instance of concept extraction is provided in Fig.2.

The proposed approach also takes in to consideration concepts with a maximum length of seven into account. Maximum number of concept's mixture to be produced for the query Q is determined as follows in equation (2):

$$Max(C_i) = \sum_{i=1}^{n} 2^{m_i} - 1 \quad (2)$$

Where m_i indicates the number of concepts in the i[th] document and n represents the number of documents.

Amount of combinations among the concept is nC_r where n indicates the length of the document, specifically, number of concepts in the document and$r$ indicates the number of words to be merged. For instance, number of concepts to be produced with the length of four in the document $D1$ is $\left(5C_4 = \frac{5!}{5!(5-4)!} = 1\right)$ it is 1.

2.2. *Architecture for Generation of Query Cluster and profile formation*

Here, the modified Support Vector Machine (mSVM) clustering is employed to generate query cluster. The user provides the query through middleware which goes to the search engine. The user's demand and their navigational behaviours are stored in the query log file. The user inspects the search result from top to bottom and makes a decision whether or not the retrieved results are appropriate as perquery demand. Occasionally, the user inspects the search results and is pleased about the information accessible in the abstract of the web snippets itself. For these scenarios, the user will not click any URL, the message "NoClick" will be allocated to the attribute ClickURL. The log entries are cleared; the users and their sessions are recognized by using the approaches given in(Umagandhi andSenthilkumar, 2012). Different pre-processing tools are existing to pre-process the log entries (Marquardt et al., 2004). Once the log entries are cleared completely, the unique queries submitted by the several users and their clicked unique URLs are recognized and accumulated in the data files using mSVM clustering.

mSVM clustering for profile management

An SVM-based clustering approach is commenced wherein clusters data have no prior information of input classes. Once this initialization phase is absolute, the SVM confidence parameters for classification with regards to each of the training requests can be accessed. The least confidence data (e.g., the most horrible of the mislabelled data) then has its' labels switched to the other class label. SVM is subsequently re-run on the data set (with moderately re-labelled data) and assuredly converges in this scenario since it converged earlier, and at this point it includes smaller amount of data points to bear mislabelling penalties. This scheme apparently limits exposure to the local minima traps that can take place with other schemes. As a result, the algorithm then enhances its imperceptibly convergent result by deploying SVM re-training following each re-labelling on each of the worst of the misclassified vectors, that is, those feature vectors with confidence factor values beyond some threshold. The recurrence of the above process enhances overall accuracy, and at this point a measure of separability occurs, until there are no misclassifications inherent. Dissimilarities regarding this type of clustering process have been shown.

Concept-based user profiles are utilized in the clustering procedure for achieving a personalization effect. The data is linearly separable, linear mSVMdetermines maximum margin linear classifier. The SVM clustering is an extension that permits efficient clustering of related text from article. The mSVM approach takes care of multi labelled concept with 'm' classes and it decomposes complexity into 'm' binary problems. There are current decomposition schemes that seem to be more dominant. On the other hand, for ease and for dissimilarity with related results, simple decomposition for SVM clustering is preferred.

Initially, a query-concept hyperplane is built by the clustering approach with one set of points related to the set of users' queries, and the other related to the sets of extracted concepts. The boundary hyperplanes on the two classes of data are parted by a distance $2/w$, known as the "margin" in which $w^2 = w_\beta w_\beta$. By raising the margin among the separated data to the extent that possible the SVM's optimal separating hyperplane is acquired. In the standard SVM process, the objective to maximize w^{-1} is restated as the objective to minimize w^2. The augmented Lagrangian formulation then chooses an optimum defined at a saddle point of,

$$L_A(x, y, w, b, \alpha) = \left(\frac{w_\beta}{2}\right) - \alpha_\gamma y_{\gamma\beta^-}\left(w_\beta x_{\gamma\beta^-} \times b\right) - \alpha \quad (3)$$

Theoretically, the positive parameter w_β in the augmented Lagrangian function, regarded as the penalty parameter, can also be modified from iteration to iteration, where $= \sum_\gamma \alpha_\gamma\, \alpha_\gamma > 0\ (1 \le \gamma \le M)$. The saddle point is acquired by lessening in accordance with $\{w_1, \ldots, w_N, b\}$ and increasing with regard to$\{a_1, \ldots, a_M\}$. Consider $\{x_i\}$ be an obtained concepts of Npoints in a space. Resembling the nonlinear SVM process, using a non-linear transformation ϕ, $x$is transformed to a high-dimensional space – *Kernel space* – and try to find the smallest amount of enclosing sphere of radius R. The Mahalanobis distance formula is employed for similarity matching and is given in equation (4)

$$\left\| \sqrt{\phi(x_j)^2 - a} \right\| \le R^2 \text{for all} j = 1, \ldots, N \quad (4)$$

where$a$ represents the center of the sphere. Starting from the center point, the clusters are generated in accordance with the $\|.\|$ Mahalanobis distance. At this moment, the cluster assignment can be determined as follows. Consider a segment of points y, the clustering rule can be characterized as the adjacency matrix is given in equation (5)

$$A_{ij} = \begin{cases} \forall y & on\ the\ line\ segmer \\ 0 & oth \end{cases} \quad (5)$$

The entire data points are verified to allocate a particular cluster. Besides that, outliers are uncategorized as their feature space lies outside the enclosing sphere. The personalized mSVM clustering approach iteratively unites the most comparable pair of query points, and subsequently the most similar pair of concept pints, and next combines the most similar pair of query points, and so on. In case of agglomerative clustering algorithm, which indicates the same queries submitted from several users by one querypoints, it is necessary to consider the same queries submitted by several users independently to accomplish the desired personalization effect. This means, when two particular queries, whether or not they are matching, represent different things to two different users, hence these should not be combined as they represent two different sets of concepts for the two users. Consequently, each individual query submitted by every user is treated as an individual vertex in the bipartite graph by labelling every query with a user identifier. Following the personalized bipartite graph is generated, the initial experiments disclose that when the algorithm is implemented directly on the bipartite graph, the query clusters produced will rapidly merge queries from different users together, as a result losing the personalization effect. It is found that similar queries, though submitted by different users and having diverse meanings, tend to have certain generic concept points like 'information' in common. Algorithm 1 provides the details of the personalized clustering algorithm, a query-concept is generated as input for the clustering algorithm. In order to implement this, clustering is partitioned into two phases. In the initial clustering phase, mSVM is executed to cluster the entire queries, however it would not combine identical queries from several users. Subsequent, the community merging step is executed to merge query clusters containing identical queries from different users.

Algorithm 1: mSVM Clustering Algorithm for Query Cluster and profile management process

Input: A Query-Concept dataset $\{\boldsymbol{x_i}\}$
Output: Clustering of $\{\boldsymbol{x_i}\}$, A Personalized Clustered Query-Concept
// Initial Clustering
Initialize query concepts $\{\boldsymbol{x_i}\}$
Apply mSVM for clustering $\boldsymbol{Cl}(\boldsymbol{A}, \boldsymbol{B})$
$\boldsymbol{Cl_A}$:= {non-bounded support vectors of A},
If ($\boldsymbol{Cl_A}$ contains more elements than R)
then R := $\boldsymbol{Cl_A}$.
$\boldsymbol{SV\_A}$:= {the support vectors of class A},
$\boldsymbol{A}$:= A minus $\boldsymbol{SV\_A}$,
$\boldsymbol{SV\_B}$:= $\boldsymbol{SV\_A}$,
//Build clusters portions
Let $\boldsymbol{R} = \{\boldsymbol{r_1}, \ldots, \boldsymbol{r_k}\}$ (obtained from Step 2).
$\boldsymbol{Cl} := \{\boldsymbol{Cl_1}, \ldots, \boldsymbol{Cl_k}\}$
With$\boldsymbol{Cl_i}\{\boldsymbol{x\ in\ X\ closer\ to\ r_i than\ to\ any\ other\ r_j}\}$.
// Join clusters portions
Replicate the following statement until $\boldsymbol{Cl}$ does not vary.
for each $\boldsymbol{Cl_i} \in \boldsymbol{Cl}$:
$\boldsymbol{c_i}$:= Adjacency matrix of $\boldsymbol{Cl_i}$,
Find $\boldsymbol{Cl_j}$ containing a point nearest to $\boldsymbol{c_i}$ using Mahalanobis distance metric
$\boldsymbol{Cl}\{\boldsymbol{x_i}\} := \left(\boldsymbol{Cl} - \{\boldsymbol{Cl_i}, \boldsymbol{Cl_j}\}\right) \cup \{\boldsymbol{Cl_i}, \boldsymbol{Cl_j}\}$,
If (score $\boldsymbol{Cl}\{\boldsymbol{x_i}\}$< score(Cl))
thenCl:= $\boldsymbol{Cl}\{\boldsymbol{x_i}\}$.
// Community Merging
Step 6. Acquire the similarity scores $\boldsymbol{Cl}\{\boldsymbol{x_i}\}$ for the entire possible pairs of queries using distance correlation.
Step 7. Combine the pair of most related queries $(\boldsymbol{q_i}, \boldsymbol{q_j})$ that includes the same queries from different users.
Step 8. Unless termination is accomplished, replicate steps 6 and 7.

2.3. Query based recommendation using Improved Artificial Bee Colony

Swarm Intelligence is an efficient approach that manages both natural and artificial systems. It offers an efficient approach for discovering optimal solutions. In the last few decades gone by researchers have attempted to employ these approaches to solve several complications in different fields. Recommender system is one of the most significant applicationsin e-commerce and it has a significant role in understanding the user's behaviour or interest by which it enlarges the revenue of sales or usage of services of website. This section explains a swarm intelligence optimization for web mining to discover the best possible solutions and on the basis of the same theremaining process is then carried out. In case of information filtering technology, companies makes use of ecommerce websites that employ collaborative filtering to present records on items and products that are expected to be of significance to the end user and reader. In providing the recommendations, the recommender system deploys the features of the registered user's profile and attitudes and patterns of their whole community of users and then evaluates the information on the basis of reference characteristics to provide recommendations. Recommendation contributes to similar group category of users on the whole. It also provides suggestions to users in accordance with their information provided previously or based on the currently browsed web page. The nearest neighbourhood selection process is employed to find the best possible solution, as well as the users that are closest to current user.

Consider Uthat indicates the current working user, the users with comparable likes are treated as the best possible solution for user U. The profiles are chosen from the database. In accordance with the threshold value, profiles are then combined together. The prominent usage profiles and those that are similar among are preferred. Web personalization implicitly or explicitly gathers data from the user. The above figure represents the structural aspects of how recommendation is obtained. Subsequently the profile is then produced and is then geared up for recommendation. The subsequent step following the profile generation and management is the nearest neighbourhood selection, which can be acquired by using by Improved Artificial Bee Colony Optimization Swarm Intelligence Techniques. After that the relative position of user and time distance function in accordance with the threshold value are computed. This relative position is taken as the weight among the edges. The recommendation process is then finally carried out. With the aim of improving the relevance of the recommendations, recommendations are ranked with the help of the relative position of user u from whom the recommendations have been derived, in addition to the edgeweight $w[u,v]$. Furthermore, page topic is employed to enhance or boost scores of page whose topics match the interests of user u. When a page is recommended by several early active users, the final recommendation from u to v is basically the sum of contributions of the early active users for v.

The process of Improved Artificial Bee Colony algorithm can be described as follows:

1. Create the initial population of user profiles.
2. Compute the relative position of user and Time Distance Function: (u_i, t_j) $RF = \sqrt{(u_{(n-1)i} - u_{ni})^2 + (t_{(n-1)j} - t_{nj})^2} + (R_f - R_i)$ where RF indicates the relative position distance function, β indicates the penalty of collision with the static obstacle and i signifies the number of profiles.
3. Compute fitness value of each of neighbourhood search generated as using equation: $fitnessi = \frac{1}{F_j}$where j indicates the neighbouring profiles whose fitness value is to be computed.
4. Transmit employed bees to enhance neighbour search arbitrarily.
5. When the fitness value of a specific neighbour profile does not enhance after certain limit of experiments, it is then discarded.
6. If not the best neighbour value is updated and its trial value is transformed.
7. Compute the Relative position of user and Time Distance Function of neighbourhood profiles by considering the collision with dynamic obstacles as using equation: $RF_d = F + \gamma * j$ (7) where γ indicates the penalty of collision with dynamic profiles and j indicates the number of dynamic profiles colliding.
8. Organize profiles including dynamic profiles in descending order of fitness value.
9. The profile holds high fitness value is expected to be chosen for further improvement by onlooker bees.
10. When a particular profile is not enhanced after certain experiments, it is discarded.
11. The employed bee of discarded profile turns into a scout bee and begins a random search for a better profile match.
12. The above process is continued to a fixed number of cycles.
13. Triangle inequality method is implemented to further optimize the best profile match result.

The steps provided in the algorithm show the complete process of the proposed approach. Subsequent to recognizing the k recommended items from several log files for the user u and for the query Q, k is shown in the search engine interface (Stefanidis et al., 2012). It is observed that the accomplishment of recommendations is based upon the reasoning behind them. This is the motivation feature for offering an explanation along with each suggested item, i.e., for explaining why this particular recommendation comes into view in the top-k list. The recommendations and their explanations are represented using a simple template mechanism or tool tip text. The recommended items are the user's preferred queries. Ranking of past queries provided by the user is done byanalyzing thereal intention from the query log, concepts with high support as well as query terms from the similar users.

3. Experimental Results and Discussions

In this section, in order to assess the performance of the user profiling strategies, 200 test queries were used, which were deliberately intended to have indefinite meanings (e.g. the query 'cricket'). This was done with the aim of providing a standard cluster for every query the mSVM is employed. The clusters obtained from the algorithms are evaluated against standard clusters to verify their correctness. 100 users are requested to make use of the search engine for exploring answers for the 200 test queries (accessible at http://www.dmoz.org/). With the intention of avoiding any bias, the test queries are arbitrarily chosen from 10 different categories. The user profiles are deployedusing themSVM clustering approach to group comparable queries collectively in accordance with users' requirements. The personalized clustering approach is a two phase approach which includes initial clustering phase in order to cluster queries inside the scope of each user, and subsequently the community merging phase in order to group queries for the community. The results of the comparison drawn on the proposed work and the existing query recommendation system

with the assistance of graph have been shown. The query recommendation of Google search engine result for 'cricket' is shown in Fig.3.

Moreover, the proposed work recommendation results have been shown in Fig.4 for 'cricket'. For the purpose of comparison, two metricshave been employed for assessing performance, i.e., Precision Factor and Average Relevance Factor.

3.1. Precision Factor Evaluation

Precision is characterized as a metric to make sure that the query returns all associated suggestions. This implies that, precision is the fraction of number of appropriate suggestions to the total number suggestions returned by the system. The formula is provided in equation (6)

$$\%\ Precision = \frac{Number\ of\ relevance\ recommendation}{Total\ number\ of\ recommendation} \times 100 \quad (6)$$

3.2. Average Relevance Factor Evaluation

The Average Relevance Factor describes the average precision of the formulated suggestion system in relation to a set of queries provided by the user. The formula is given in equation (7):

$$\%\ Average\ Relevancy = \frac{\sum_{i \in N} no.of\ users's\ relevance\ suggestion\ fo}{|Q|} \times 100 \quad (7)$$

where$N$ indicates the total number of queries and Q indicates the set of queries.

The amount of queries exhibiting precision rate in the particular range of threshold values is explicitly shown in Fig.5. Based on the figure it is evident that the proposed work generates comparatively better precision results than existing ones. The reason of better results is the fact that computational complexity of the SVM with IABC is high whichleads to increase in the precision rate. When the number of queries increases the precision rate of the proposed system subsequently increases.

The number of queries exhibiting similarity in the specified range of threshold values is specifically brought out in Fig.6. Based on the figure it is evident that the proposed work provides better results than the existing work. The reason for the same is that the convergence speed of the IABC with mSVM is high which leads to increase in relevancy factor. When the number of queries increases the average relevancy of the proposed system also increases.

3.3. Recommendation Evaluation

The proposed recommendation is assessed using an evaluation form. The users are requested to search in one query category. On the evaluation form, the users are also requested to provide relevancy score for listed queries. For each recommended query, the users are requested to label it with a relevancy score $\{0, 1, 2\}$ where 0: irrelevant, 1: partially relevant, and 2: relevant. The amount of recommended queries varies and is in accordance with the aim of the user. In Table 3, $\{R1, R2 \ldots . R7\}$ indicates the recommended queries. Here $R1$ is constantly the favourite query of the user. Though it may perhaps be inappropriate at several other times.

Fig.7 shows that most of the users scored the recommended queries as either relevant or most relevant. From the figure it may be concluded that the proposed work here secures high relevancy scores than the exiting methods of agglomerative clustering and time independent user recommendation system.

4. Conclusion

In this work, a novel design of query based recommendation system using mSVM with IABC has been proposed for realizing effective web search. The most vital aspect is that this approach completely depends on users' behavior, which in turn decides the relevance among concept and user query words. Furthermore, the results are also optimized by ensuring its nearest neighbour search for query recommendation using the IABC approach. In this manner, the time user spendson the required information from search result list can be considerably trimmed down and more appropriate keywords can thus be presented. The results obtained from practical evaluation are relatively promising in terms of enhancing the effectiveness of interactive web search engines. The experimental results have shown clearly that the approach deployed here outperforms two baselines in both spheres; coverage and quality. It covers the semantic relation with the query and user behavior from search log. Also it enhances quality by using several ranking techniques on the results.

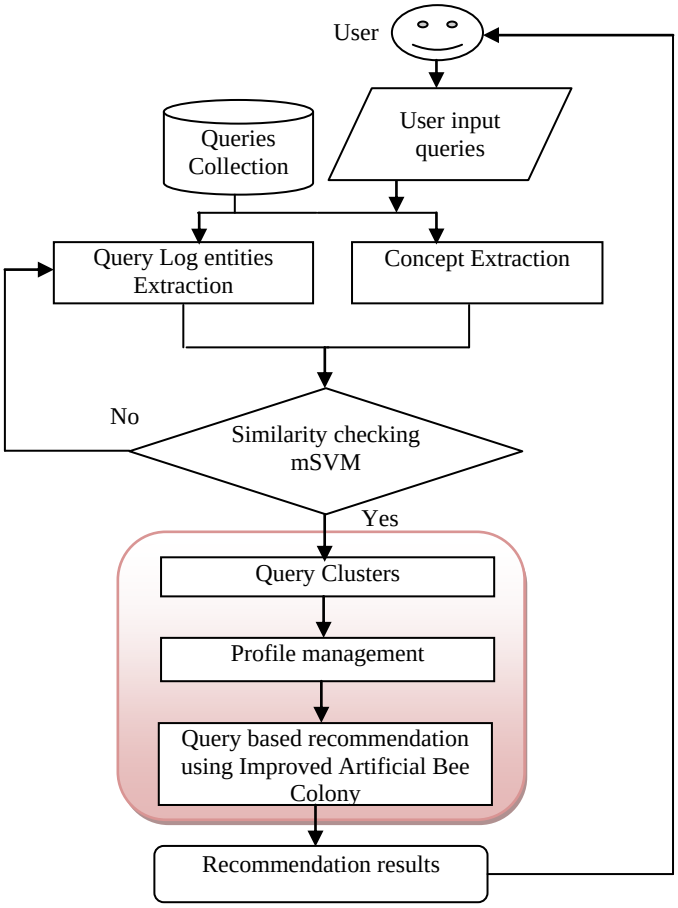

Fig.1. The Overall Architecture Diagram

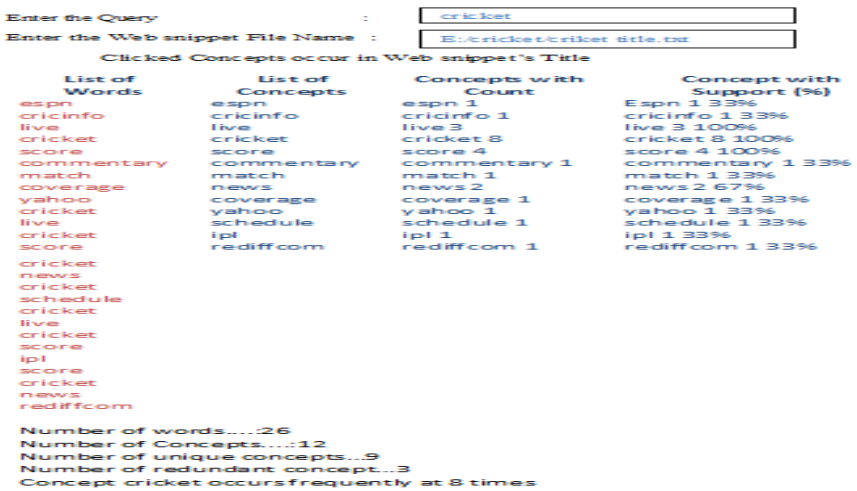

Fig.2. Concept Extraction with support value calculation

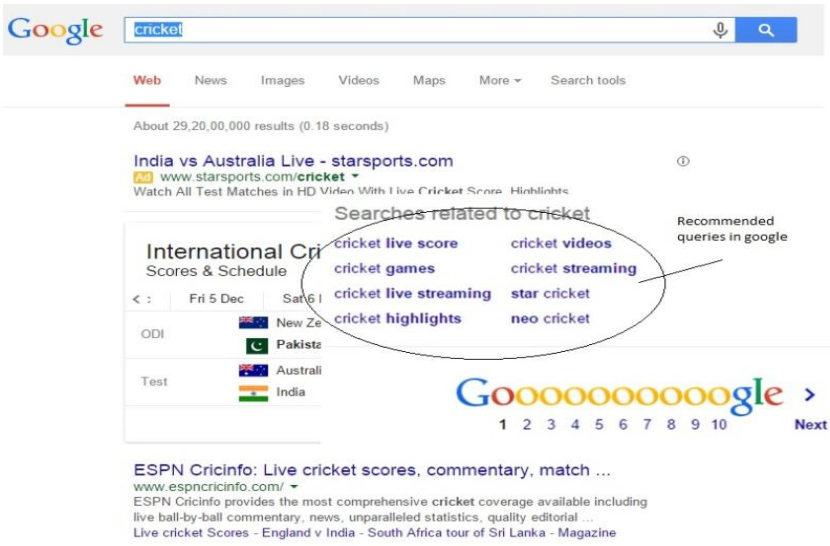

Fig.3. Recommendation from Google for 'cricket'

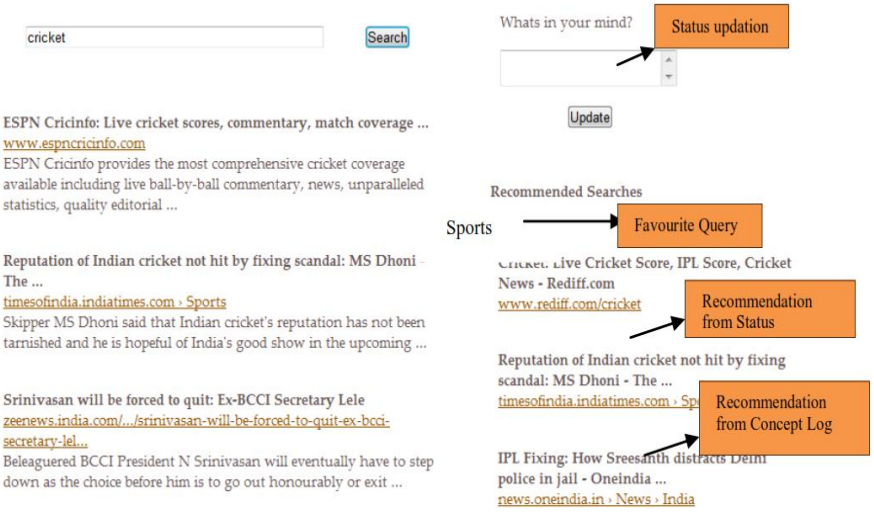

Fig.4. Recommendations from mSVM-ABC recommendation system for 'cricket'

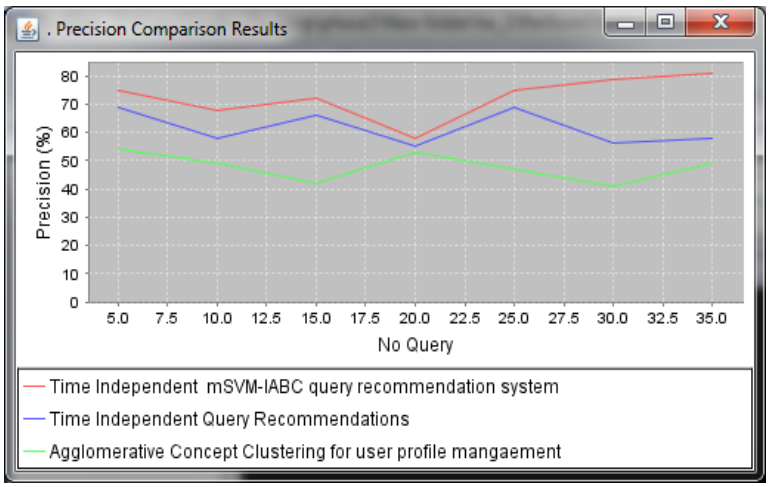

Fig.5. Precision Comparison Results

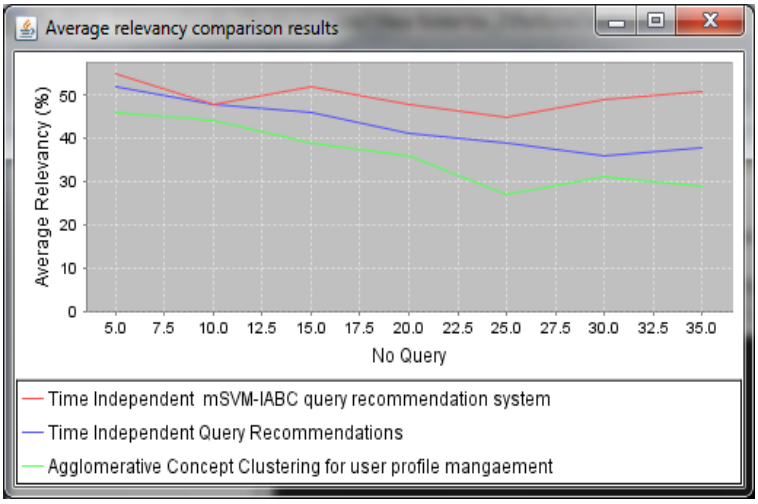

Fig.6. Average relevancy comparison results

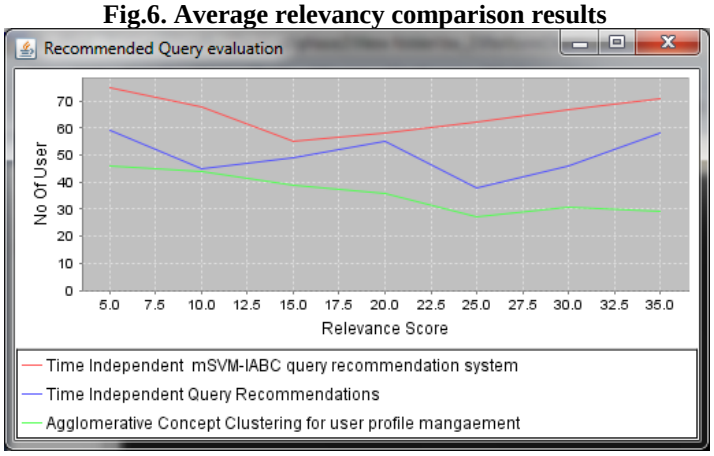

Fig.7. Recommended Query evaluation

Table 1: Example of Clicked Documents with Concept

URL	Concepts
www.espncricinfo.com/ci/engine/match/scores/live.html	cricket score
www.starsports.com/cricket/index.html	cricket live streaming
www.espncricinfo.com/ci/content/current/match/fixtures/	cricket match schedule
www.cricbuzz.com/cricket-news	cricket news
http://www.icc-cricket.com/cricket-world-cup	cricket world cup 2015
en.wikipedia.org/wiki/List_of_One_Day_International_cricket_records	cricket player with most international records
india.cricketworld4u.com/profile/	cricket players profile

Table 2: Attribute along with its description

Attribute	Description
AnonID	Anonymous ID assigned for every user
Query	The query supplied by the user
QueryTime	The date and time on which the query triggered by the user
ItemRank	Rank assigned to each clicked URL
ClickURL	The URL address clicked by the user when the query was supplied
Concepts	Important concepts retrieved from the clicked URL

Table.3. User relevancy score

Query: Cricket	R1	R2	R3	R4	R5	R6	R7
User 1	0	2	2	1	0	2	0
User 2	1	2	2	2	1	1	1
User 3	0	1	1	0	1	0	0
User 4	0	2	1	1	0	0	1
User 5	2	2	2	0	0	0	1
User 6	2	2	2	0	1	1	2
User 7	0	2	1	1	1	0	2
User 8	0	1	2	2	2	1	2
User 9	1	2	0	2	2	2	0
User 10	1	2	1	0	2	2	0

References

[1] Wen, J. R., Nie, J. Y., Zhang, H. J.: Clustering user queries of a search engine. In Proceedings of the 10th international conference on World Wide Web, ACM, 162-168 (2001).

[2] Baeza-Yates, R., Hurtado, C., Mendoza, M.: Improving search engines by query clustering. Journal of the American Society for Information Science and Technology. 58,1793-1804 (2007).

[3] Baeza-Yates, R., Hurtado, C., Mendoza M., Dupret, G.: Modeling user search behaviour, in Web Congress, IEEE. 1-10 (2005).

[4] Grimes, C., Tang, D., Russell, D. M.:Query logs alone are not enough, In Workshop on query log analysis at WWW, (2007).

[5] Hsieh-Yee, I.: Research on Web search behaviour. Library & Information Science Research. 23, 167-185 (2001).

[6] Speretta, M., Gauch, S. : Personalized search based on user search histories. In Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence, 622-628 (2005).

[7] Wen, J. R., Nie, J. Y., Zhang, H. J. : Query clustering using user logs. ACM Transactions on Information System. 20, 59-81. (2002).

[8] Zigoris, P., Zhang, Y.: Bayesian adaptive user profiling with explicit & implicit feedback. In Proceedings of the 15th ACM international conference on Information and knowledge management, ACM, 397-404 (2006).

[9] Sugiyama, K., Hatano, K., Yoshikawa, M.: Adaptive web search based on user profile constructed without any effort from users. In Proceedings of the 13th international conference on World Wide Web, ACM. 675-684 (2004).

[10] Joachims, T., Granka, L., Pan, B., Hembrooke, H., Gay, G.: Accurately interpreting click through data as implicit feedback. In Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval. 154-161 (2005).

[11] Chen, C.C., Chen, M. C., Sun, Y. : PVA: A self adaptive personal view agent. Journal of Intelligent Information Systems, 18,173-194 (2002).

[12] Claypool, M., Le, P., Wased, M., Brown, D.: Implicit interest indicators. In Proceedings of the 6th International Conference on Intelligent User Interfaces, ACM, 33-40 (2001).

[13] Baeza-Yates, R., Hurtado, C., Mendoza, M.: Query recommendation using query logs in search engines. In Current Trends in Database Technology EDBT Workshops, Springer Berlin Heidelberg, 588-596 (2004).

[14] Baeza-Yates, R., Tiberi, A.: Extracting semantic relations from query logs. In Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 76-85 (2007).

[15] Cucerzan, S., White, R. W.: Query suggestion based on user landing pages, In Proceedings of the 30th annual international conference on Research and development in information retrieval, ACM SIGIR, 875-877 (2007).

[16] Fonseca, B. M., Golgher, P. B., de Moura, E. S., Ziviani N.: Using association rules to discover search engines related queries. In Web Congress, IEEE, 66-71 (2003).

[17] Liu, Y., Miao, J., Zhang, M., Ma, S., Ru, L.: How do users describe their information need.Query recommendation based on snippet click model. Expert Systems with Applications. 38, 13847-13856 (2011).

[18] Yiu, M. L., Mamoulis, N.: Efficient processing of top-k dominating queries on multi-dimensional data. In Proceedings of the 33rd international conference on Very large data bases VLDB Endowment, 483-494 (2007).
[19] Umagandhi, R., Senthil Kumar, A.V.:Time Heuristics Ranking Approach for Recommended Queries using Search Engine Query Logs, Kuwait journal of Science and Engineering, communicated. (2013)
[20] Beeferman, D., Berger, A.:Agglomerative clustering of a search engine query log. in Proceedings of the sixth international conference on Knowledge discovery and data mining, ACM SIGKDD, 407-416 (2000).
[21] Joachims, T.: Optimizing search engines using click through data. In Proceedings of the eighth international conference on Knowledge discovery and data mining, ACM SIGKDD 133-142 (2002).
[22] Leung K.W.T., Lee, D. L. : Deriving concept based user profiles from search engine logs. IEEE Transactions on Knowledge and Data Engineering, 22, 969-982 (2010).
[23] Umagandhi, R., Senthilkumar, A.V.: An Efficient Method to Identify Users and Sessions from Web Logs. IJARCS, 3, 50-54 (2012).
[24] Marquardt, C., Becker, K., Ruiz, D.: A Pre-Processing Tool for Web Usage Mining in the Distance Education Domain.In Proceedings of the International Database Engineering and Applications, 78-87 (2004).
[25] Stefanidis, K., Ntoutsi, I., Norvag, K., Kriegel, H. P.: A framework for time-aware recommendations. In Database and Expert Systems Applications, Springer Berlin Heidelberg, 329-344 (2012).