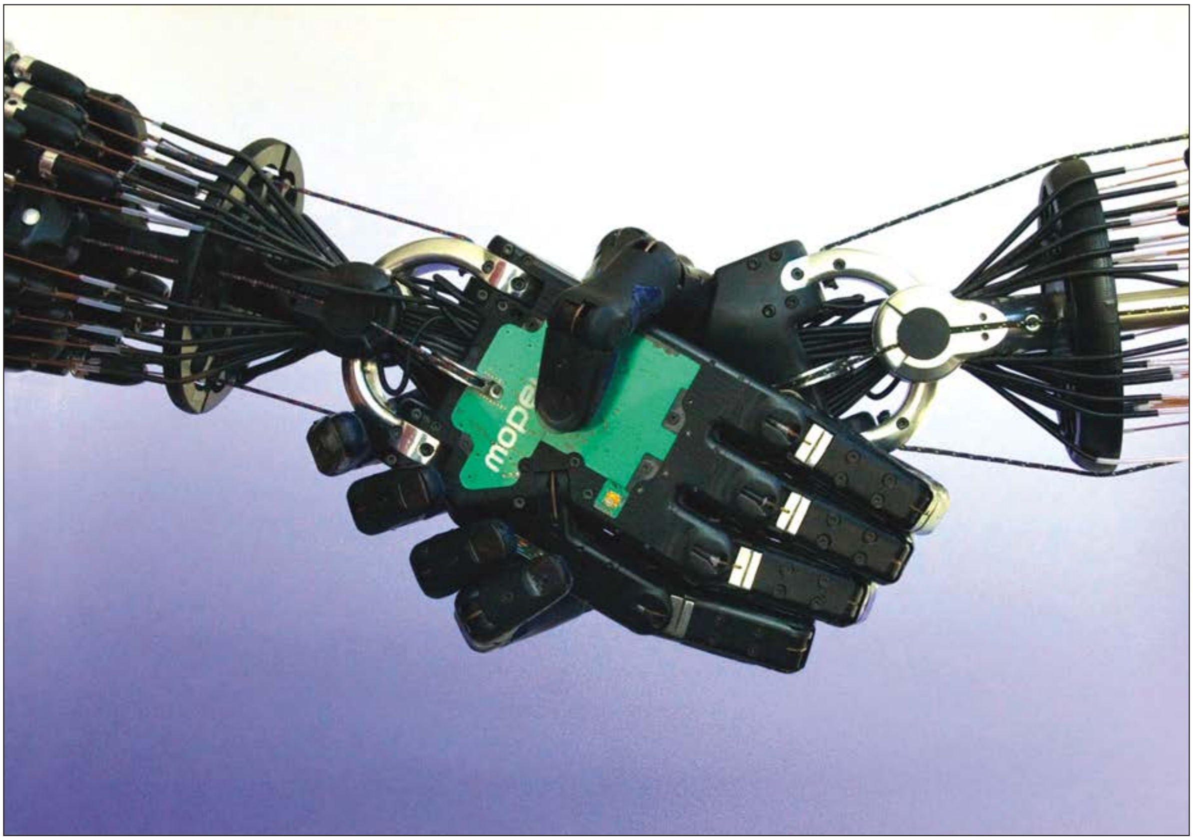

Journal of Artificial Intelligence

ISSN 1994-5450

science alert

ANSInet
an open access publisher
http://ansinet.com

Journal of Artificial Intelligence 3 (4): 220-238, 2010
ISSN 1994-5450

Agent Based Information Retrieval System Using Information Scent

P. Bedi and S. Chawla
Department of Computer Science, Room No. 108 New Academic Block,
Adjoining Art Faculty Building, University of Delhi, Delhi-110007, India

Abstract: Information on the web is increasing at an enormous speed. Search histories of the users, browsing pattern, query expansion using relevance feedback are some of the techniques used in the literature to personalize the web search. This study proposed the integrated approach of personalized web search using Agents and Information Scent. Agent based information retrieval system personalizes the web search by clustering the query sessions of users on the web using information scent, information scent is the measure of the sense of value of clicked web page in the query session with respect to the information need of the user. Interface agent after receiving the input query generates the query recommendations using the cluster which is closes to the information need of input query and is expanded using related keywords to disambiguate its context. The proposed framework effectively personalizes the web search through input query sense disambiguation and exposes the part of Surface Web through Hubs and Authorities recommendations using user profile modeled using Information Scent of clicked URLs and clusters of query sessions on the web. The effectiveness of the proposed system on the precision of search results is confirmed with the experiments conducted on the data set collected using proposed architecture.

Key words: Information retrieval, agent, information scent, information need, hubs, authorities

INTRODUCTION

Information on the Web is very huge in size and there is a need to use this big volume of information efficiently for effectively satisfying the information need of the user on the web. Search engines are the major breakthrough on the web for retrieving the information relevant to the user queries for a specific information need. In Information Retrieval System on the Web it is found that input query entered by the user contain few and sometimes ambiguous words which are not sufficient to infer the information need of the user and hence retrieve the documents, out of which very few documents are truly relevant to the user among the billion retrieved by the search engine (Jansen *et al.*, 1998; Gudivada *et al.*, 1997). There is the need to improve the precision of search results to satisfy the information need of the user effectively. Extensive work has been done in the direction of personalization of the web search. In Salton (1983) and Allan (1996) techniques such as relevance feedback and query expansion has been employed in the personalization domain where the system automatically expands the user query with certain words that bring relevant documents not literally matching the original query.

Corresponding Author: S. Chawla, Department of Computer Science,
Room No 108 New Academic Block, Adjoining Art Faculty Building,
University of Delhi, Delhi-110007, India Tel: 91-11-27667059,27667591

Raghavan and Sever (1995) used a database of past queries that is matched with the current user query. If a significant similarity with a past query is found, the past results associated with the query are proposed to the user. Speretta and Gauch (2005) developed the misearch system, which improves search accuracy by creating user profiles from their query histories and/or examined search results. These profiles are used to re-rank the results returned by an external search service by giving more importance to the documents related to topics contained in their user profile. Rhodes (2000) proposed a new approach for the search named Just-in-Time IR (JITIR) where the information system proactively suggests information based on a person's working context, automatically identifying their information needs and retrieving useful documents without requiring any action by the user. Google labs released an enhanced version of Personalized Search that builds the user profile by means of implicit feedback techniques, adapts the results according to needs of each user, assigning a higher score to the resources related to what the user has seen in the past (Zamir *et al*., 2004).

In Liu *et al.* (2004) personalization is done where user profile are built by analyzing the search history, both queries and selected result documents, comparing them to the first 3 levels of the OPEN DIRECTORY PROJECT category hierarchy. For each query, the most appropriate categories are deduced and used along with the query as current query context. Koutrika and Ioannidis (2005) proposed an online approach where user needs are represented by a combination of terms connected through logical operators which are used to transform the queries in personalized versions to be submitted to the search engines. Quickstep system (Middleton *et al*., 2001) follows a quasi-online approach where proxy server monitors browsed research papers and a nearest neighbor classifier assigns OPD categories to them overnight. In CubeSVD algorithm is introduced which is based on click through data analysis reflecting user's interest (Sun *et al*., 2005; Joachims *et al*., 1997; Joachims, 2002). The proposed algorithm aims to model the users' information needs by exploiting such data. If Web is a user model based intelligent agent capable of supporting the user in Web navigation, retrieval and filtering of documents taking into account specific information needs expressed by the user with keywords, free-text descriptions and Web document examples (Asnicar and Tasso, 1997). Micarelli and Sciarrone (2004) described the Wifs (Web Information Filtering System) which evaluates and reorders page links returned by the search engine, taking into account the user model of the user who typed in the query. InfoWeb (Gentili *et al*., 2003), an interactive system developed for adaptive content-based retrieval of documents belonging to Web digital libraries. The distinctive characteristic of InfoWeb is its mechanism for the creation and management of a stereotype knowledge base and its use for user modeling. The EUREKSTER4 search engine includes a proprietary module named Search Party based on collaborative filtering to help users find the best pages related to a given query. A social adaptive navigation system called Knowledge Sea (Ahn *et al*., 2005) exploits both the traditional IR approach and social navigation based on past usage history and user annotations. Users can search socially, referring to other users' behavior and opinions, by examining the color lightness and exploring icons next to each result, which respectively provide users with information about the popularity of the page and allow the user to view any available annotations. Compass filter Kritikopoulos and Sideri (2005) follows a similar collaborative approach, but it is based on Web communities by analyzing the Web hyperlink structure, similarly to the HITS algorithm (Kleinberg, 1998). Claypool *et al.* (1999) explores a possible combination of collaborative and content-based approaches by basing the interest prediction of a document on a weighted average adapted to the individual user. The DirectHit search engine used a popularity-based search algorithm, ranking URLs in order of popularity, with the pages visited most by other users ranking highest in their search results (Direct Hit, 1995).

I-SPY which is a popularity-based search interface designed to be used in conjunction with specialised search (Freyne *et al*., 2004) engines, with communities of users having similar interests and information needs. The SnakeT meta-search engine Ferragina and Gulli (2005) includes an innovative hierarchical clustering algorithm with reduced time complexity. It allows the users to select a subset of the clusters that are more likely to satisfy their needs. Then, the system performs a query refinement, building and submitting a new query that incorporates keywords extracted by the system from the selected clusters.

In Scatter/Gather the user is able to select one or more clusters for further analysis (Cutting *et al*., 1992). The selected groups are further clustered into a small number of clusters which are again presented to the user. After a sequence of iterations, the clusters become small enough and the resources are shown to the user. Outride Inc., an information retrieval technology company acquired by Google (2001), introduced a contextual computing system for the personalization of search engine results (Pitkow *et al*., 2002). InfoFACTORY (Tasso and Omero, 2002) contains a large set of integrated Web tools and services that are able to evaluate and classify documents retrieved following a user profile. This system suggests new, potentially interesting contents as soon as it is published on the Web. Lieberman (1995), Lieberman and Selker (2000) a client-side agent that accompanies the user as they browse the web, recommending links from the current page being browsed when asked to do so. WebWatcher (Armstrong *et al*., 1995; Joachims *et al*., 1997) acts as a tour guide in a similar way to Letizia, but [4]http://www.eurekster.com) requires the user to explicitly specify a goal at the outset. In (Pazzani *et al*., 1996) the user must provide explicit feedback by rating pages visited on a three-point scale. The user profile consists of a set of topic-specific profiles and is used to suggest web pages that the user might wish to visit from a topic-specific index- page that the user must supply the URL for. WebMate (Chen and Sycara, 1998) stores a separate section of user profile for each user interest, which it learns automatically and is used to suggest documents that match the user's interests and compiles a personalized newspaper from various news sources on the web. The ICPF (Zhu *et al*., 2003) provides the basis for a client-side recommender system. The system is initially trained using manually annotated logs of session histories to identify information content (IC) pages i.e., pages that contain information relevant to the user based on browsing behavior (such as following a link or going back a page). Adaptive Information Server (AIS) Billsus and Pazzani (2000) is a client-server agent for adaptive news access. It uses a hybrid user model, with separate models for long-term and short-term interests. Two personalised news services currently available on the web are Findory and MSN43 Newsbot (beta). Findory's algorithm appears to be a hybrid, combining both content-based and collaborative filtering methods to suggest news stories. MSN Newsbot clusters stories into categories for presentation. Ant Recommender System (ARS) is developed which is based on collaborative behavior of ants for generating Top-N recommendations (Bedi *et al*., 2009).

Research has been done by Bedi and Chawla (2007) to improve the Information Retrieval precision using Information Scent. In this study Agent based architecture of the Information Retrieval System on the Web with Information Scent has been proposed which improves the Information retrieval precision of the search engine results by personalizing the web search of the user in the domain in which user is searching for information using the information need of the user. The proposed Architecture uses the Multi-Agent System to personalize the web search of the user on the search engine in order to satisfy the information need of the user at higher precision by augmenting it with domain specific search. The novelty of the proposed approach is that it provides the integrated approach for personalization of web search using query operations and Hubs and Authorities recommendation together with the use of Information Scent in inferring the Information need of the users query session. It takes into consideration the input query sense disambiguation and personalizing the web search with Hubs and Authorities recommendation using Information need inferred from the query sessions of the user. The information need of the query sessions of the past users is modeled using Information Scent and content of clicked pages. Information Scent has been used in Query Session mining in which each clicked URLs is associated with the quantitative value called Information Scent which is the measure of the relevancy of clicked URLs with respect to the Information need of the user associated with the query session in which it is present. High Scent Information Sources are more relevant to the information need of the user than the Low Scent Information Sources. All the High Scent Clicked URLs of each cluster are used as seed set to crawl the neighboring links which are processed using Information Scent to generate the Authoritative and Hub Web pages of the domain associated with each cluster. The user response to the augmented search results with High Scent Hubs and Authorities Recommendations is tracked during the search session of the user and when the user request for the next result page the partial session of the user which contain its clicked URLs is used to further infer the information need of the user and used for Hubs and Authoritative recommendation as his information need become more evident from his session. The recommendation process continues till the user information need is satisfied.

The effectiveness of the proposed approach of Information Retrieval System with Information Scent and Agents has been confirmed with the results of Experimental Study which shows the significant improvement in the Information Retrieval precision of Google search engine results confirming the effectiveness of proposed approach in satisfying the information need of the users.

BASIC CONCEPTS

Software Agent

Software Agent is a software entity which functions continuously and autonomously, often inhabited by other agents and processes (Shoham, 1997). The requirement for continuity and autonomy derives from our desire that an agent be able to carry out activities in a flexible and intelligent manner that is responsive to changes in the environment without requiring constant human guidance or intervention shown in Fig 1. Ideally, an agent that functions continuously in an environment over a long period of time would be able to learn

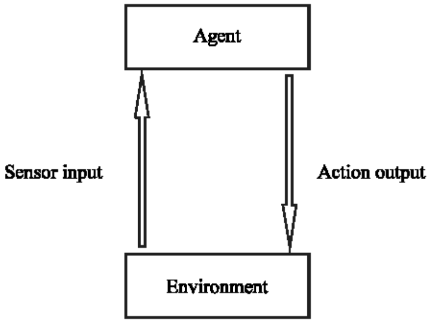

Fig. 1: Flexible and intelligent manner that is responsive to changes

from its experience. In addition, we expect an agent that inhabits an environment with other agents and processes to be able to communicate and cooperate with them and perhaps move from place to place in doing so.

Consistent with the requirements of a particular problem, each agent might possess to a greater or lesser degree attributes given as follows:

- **Reactivity:** The ability to selectively sense and act
- **Autonomy:** Goal-directedness, proactive and self-starting behavior
- **Collaborative behavior:** Can work in concert with other agents to achieve a common goal
- **Inferential capability:** Can act on abstract task specification using prior knowledge of general goals and preferred methods to achieve flexibility; goes beyond the information given and may have explicit models of self, user, situation and/or other agents.
- **Temporal continuity:** Persistence of identity and state over long periods of time
- **Personality:** The capability of manifesting the attributes of a believable character such as emotion
- **Adaptivity:** Being able to learn and improve with experience
- **Mobility:** Being able to migrate in a self-directed way from one host platform to another (Etzioni and Weld, 1995; Franklin and Graesser, 1996; Kim *et al*., 1997; Shoham, 1997).

When taken together, these attributes mark software agents as a fundamentally new paradigm-markedly different from related IT disciplines such as object-oriented systems, artificial intelligence and distributed computing.

Types of agents:

- Collaborative agents
- Interface agents
- Mobile agents
- Information/Internet agents
- Reactive agents

There are some applications which combine agents from two or more of these categories and we refer to these as heterogeneous agent systems.

Collaborative Agents

Collaborative agents emphasize autonomy and cooperation (with other agents) in order to perform tasks for their owners. General characteristics of these agents include autonomy, social ability, responsiveness and proactiveness.

Interface Agents

Interface agents emphasize autonomy and learning in order to perform tasks for their owners. In a key proponent of this class of agents, points out that the key metaphor underlying interface agents is that of a personal assistant who is collaborating with the user in the same work environment (Pattie, 1994). Note the subtle emphasis and distinction between collaborating with the user and collaborating with other agents as is the case with collaborative agents. Collaborating with a user may not require an explicit agent communication language as one required when collaborating with other agents.

Mobile Agents

Mobile agents are computational software processes capable of roaming Wide Area Networks (WANs) such as the WWW, interacting with foreign hosts, gathering information on behalf of its owner and coming back home having performed the duties set by its user. These duties may range from a flight reservation to managing a telecommunications network. However, mobility is neither a necessary nor sufficient condition for agenthood. Mobile agents are agents because they are autonomous and they cooperate, albeit differently to collaborative agents.

Information/Internet Agents

Information agents have come about because of the sheer demand for tools to help us manage the explosive growth of information we are experiencing currently and which we will continue to experience henceforth. Information agents perform the role of managing, manipulating or collating information from many distributed sources.

Reactive Software Agents

Reactive agents represent a special category of agents which do not possess internal, symbolic models of their environments; instead they act/respond in a stimulus-response manner to the present state of the environment in which they are embedded.

Agents has its applications in various domain like industry, Commercial Applications, Information Management, Electronic Commerce, Business Process Management, Medical Applications, Patient Monitoring etc. (Newell, 1982).

Multi-Agent Systems

A multi agent system can be thought of as a group of interacting agents working together to achieve a set of goals. The coordination mechanisms have been developed to help the agents interact when performing complex actions requiring teamwork. These mechanisms must ensure that the plans of individual agents do not conflict, while guiding the agents in pursuit of the goals of the system. In a pure Multi-Agent System, MAS, the agents are autonomous, potentially pre-existing and typically heterogeneous. An MAS does not require a restriction to a single task. The major concerns in these systems include coordinating intelligent behavior among a collection of autonomous intelligent agents and how they can integrate their knowledge, goals, skills and plans jointly to take action or to solve problems. Although, an agent here can also be a special task performer, it has an open interface, which is accessible to everyone. The Agents may not only be working toward a single goal, but also toward separate individual goals.

Communication among agents may vary from simple forms to sophisticated ones, as the one based on speech act theory. A simple form of communication is that restricted to simple signals, with fixed interpretations. A more elaborate form of communication is by means of a blackboard structure. A blackboard is a shared resource, usually divided into several areas, according to different types of knowledge or different levels of abstraction in problem solving, in which agents may read or write the corresponding relevant information for their actions. Another form of communication is by message passing between agents. Intelligent web-based system allows integration, easy access and sharing of domain specific Knowledge-Based Systems (KBSs) (El-Korany and El-Bahnasy, 2009). Coordination among agents is essential for achieving the goals and acting in a coherent manner. Coordination implies considering the actions of the other agents in the system when planning and executing one agent's actions. Coordination is also a means to achieve the coherent behavior of the entire system. Coordination may imply cooperation and in this case the agent society works towards common goals to be achieved, but may also imply competition, with agents having divergent or even antagonistic goals. Organization is defined as a coordination pattern of decision-making and communication among a set of agents who perform tasks to achieve goals in order to reach a global coherent state (Georgeff, 1984).

Information Scent

On the web, users search for information by navigating from page to page along the web links. Their actions are guided by their information need. Information scent is derived from Information Foraging theory (Olston and Chi, 2000; Pirolli and Card, 1999). The information scent cues play an important role in guiding users to the information they seek and they also play a role in providing users with an overall sense of the contents of collections. Information scent is the measure of sense of value and cost of accessing a page based on perceptual cues with respect to the information need of user. More the page is satisfying the information need of user, more will be the information scent associated to it. The interaction between user needs, user action and content of web can be used to infer information need from a pattern of surfing (Chi *et al*., 2001; Pirolli, 2004; Agichtein *et al*., 2006). Information scent is used in the proposed system to derive the quantitative measure of the sense of value of the clicked pages in query session with respect to the information need of the user associated with the query session. For a given sequence of clicked documents in particular query session more unique is the frequently clicked page to the session relative to the entire set of query sessions present in the data set, more likely it is close to the information need of the current query session and thus more is the information scent associated to it in determining the information need of the session. Another parameter that is taken in accessing the information scent of the clicked pages is the time spent on the clicked pages. The reason for considering the time factor is that the clicked page which consumes more user attention is more likely to satisfy his information need than the page which takes less time of user. Thus both the parameters decide the relevancy of the pages in determining the information need associated to query sessions using Information Scent. The concept of Information Scent takes into consideration the fact that every document clicked by the user in a particular query session are not equally relevant with respect to the information need of the user.

Modeling Information Need of Query Sessions Using Information Scent

The Inferring User Need by Information Scent (IUNIS) algorithm has been used for implicit user modeling method to infer the information need of the user (Chi *et al*., 2001; Heer and Chi, 2001; Pirolli, 1997). Page access PF.IPF weight and TIME are used to quantify the information scent associated with the clicked page in a query session in order to infer the information need of query sessions. In page access PF.IPF the PF is the access frequency of the clicked page in the given query session and the IPF is the ratio of total query sessions in the data set to the number of query sessions in which this page is clicked. This factor gives high weightage to those pages that are uniquely and frequently accessed in the query session and are relevant to the information need associated with the current query session. The second factor that is taken is Time spent on a page in a given query session. By including the time more weightage is given to those pages that consume more user attention. The information scent s_{id} is calculated for each page P_{id} in a given session Q_i as follows:

$$s_{id} = \text{PF.IPF } (P_{id}) * \text{Time } (P_{id}) \forall d \in 1..n \tag{1}$$

$$\text{PF.IPF } (P_{id}) = f_{P_{id}}/\max (f_{P_{id}}) * \log (M/m_{P_{id}}) \text{ where } d \in 1..na \tag{2}$$

where, n is the number of distinct clicked pages in the query session Q_i.
PF.IPF(P_{id}) and Time(P_{id}) are defined as follows:
PF.IPF (P_{id}) : PF corresponds to page P_{id} normalized frequency $f_{P_{id}}$ in a given query session Q_i and IPF correspond to the ratio of total number of query sessions M in the whole data set to the number of query sessions $m_{P_{id}}$ that contain the given page P_{id}.
Time (P_{id}): It is the ratio of time spent on the page P_{id} in a given session Q_i to the total duration of session Q_i.

The content vector of a page P_{id} is a keyword vector $(w_{1,id}, w_{2,id}, w_{3,id}, \ldots, w_{v,id})$ where v is the number of terms in the vocabulary set V. Vocabulary V is a set of distinct terms found in all distinct clicked pages in whole dataset relevant to a content feature. TF.IDF (term frequency * inverse document frequency) term weight is used to represent the content vector for a given page P_{id} (Baeza-Yates and Ribeiro-Neto, 1999; Salton, 1983).

Each query session is constructed as linear combination of vector of each page P_{id} scaled by the weight s_{id} which is the information scent associated with the page P_{id} in session Q_i. That is:

$$Qi = \sum_{d=1}^{n} Sid * P_{id} \tag{3}$$

In above formula n is the number of distinct clicked pages in the session Q_i and s_{id} (information scent) is calculated for each page P_{id} in a given session Q_i using Eq. 1 and 2. Each query session Q_i is obtained as weighted vector using formula (3). This vector models the information need associated with the query session Q_i.

Clustering Queries

Clustering is the process of grouping the data into classes or clusters so that objects within a cluster have high similarity in comparison to one another, but are very dissimilar to objects in other clusters. Dissimilarities are accessed based on attribute values describing the objects (Berkhin, 2002; Jain *et al*., 1999; Wen *et al*., 2002; Zhao and Karpis, 2002).

Leader-SubLeaders algorithm is the Hierarchical Algorithm which can be applied to both numerical and sequence data. The time and space complexity of the algorithm is O (ndh) and O ((L+SL)d) where n is the total No. of patterns, d is the dimensionality of the pattern, h is the number of levels h = 2, L is the number of Leaders and SL is the No. of subleaders (Vijaya, 2004).

Query sessions vector are clustered using Leader-SubLeaders algorithm because of its good performance and better classification accuracy for large data set. Leader-SubLeaders is an incremental algorithm which creates L clusters with L Leaders and SLi subleaders in the i th cluster. SubLeaders are the representative of the subclusters and help in classifying the given test pattern more accurately. Leaders-SubLeaders algorithm requires only two database scan to find the subclusters /clusters of the given dataset.

The Leader-SubLeader Algorithm has been applied to query sessions which are real valued vector. The similarity function chosen is cosine similarity which calculates the extent to which two query sessions are similar.

Query Sessions Vector Clustering Using Leader-Subleaders Algorithm

Leaders Computation Algorithm

Input: {Query sessions vector DataSet, Threshold}

Output: {Leader List associated with Clusters of Query sessions where each cluster is represented by the Leader}

Algorithm

```
Select any Query session vector as the initial Leader and add it to the Leader List and set Leader counter LC=1
For all Query sessions vector not yet processed
{
    Select the Query session vector Q.
    Calculate the similarity of Q with all the Leaders.
    Select the Leader which has maximum similarity represented by max.
    If (max>= threshold)
    {
        Assign it to the selected Leader.
        Mark the cluster number associated with the selected Leader for Q.
        Add it to the member List of this cluster.
        Increment the member count of this cluster.
    }
    else
    {
        Add Q to Leader list.
        Increment Leader counter LC=LC+1.
    }
}
```

SubLeaders Computation Algorithm

Input: { Leader List associated with Clusters of Query sessions where each cluster is represented by the Leader, SubThreshold: SubThreshold > Threshold value for similarity of Query sessions vector in SubLeaders }

Output: { Leader-SubLeader List is associated with Clusters of Query sessions where each cluster is represented by the Leader and SubLeaders of the i^{th} cluster are associated with the corresponding subclusters. }

Algorithm

```
For i= 1 to LC
{
    Initialize SubLeader List $SL_i$ with any Query session vector Q in $i^{th}$ cluster.
    set counter $SL_i$Count=1;
    for all j= 2 to $|Cluster_i|$
    {
        Calculate the similarity of Qij with all subleader in SLi
        Select the subLeader with maximum similarity represented by max1
        If (max1>= subthreshold)
        {
            Assign it to the selected SubLeader.
            Mark the subcluster number associated with the selected SubLeader for Qij.
            Add it to the member List of this subcluster.
            Increment the member count of this subcluster.
        }
        else
        {
            Add Qij to the SubLeader List $SL_i$.
            Set $SL_i$ Count= $SL_i$ Count +1.
        }
    }
}
```

Online Processing of Input Queries Session Vector

```
For each Input Query Session Vector q
{
    Calculate the similarity of q with all Leader vector associated with their clusters.
    Select the Leader with the maximum similarity represented as max. Calculate the similarity of q with all SubLeaders of the selected Leader.
    Select the SubLeader with the maximum similarity represented as max1.
    If(max> max1)
    {
        Use the cluster associated with selected Leader.
    }
    Else
    {
        Use the subcluster associated with selected SubLeader.
    }
}
```

ARCHITECTURE OF AGENT BASED INFORMATION RETRIEVAL SYSTEM USING INFORMATION SCENT

The proposed architecture shown in Fig. 2 is based on blackboard approach where all the communication occurs through a global shared memory structure query sessions database called the blackboard. Agents use available information without knowing its origin.

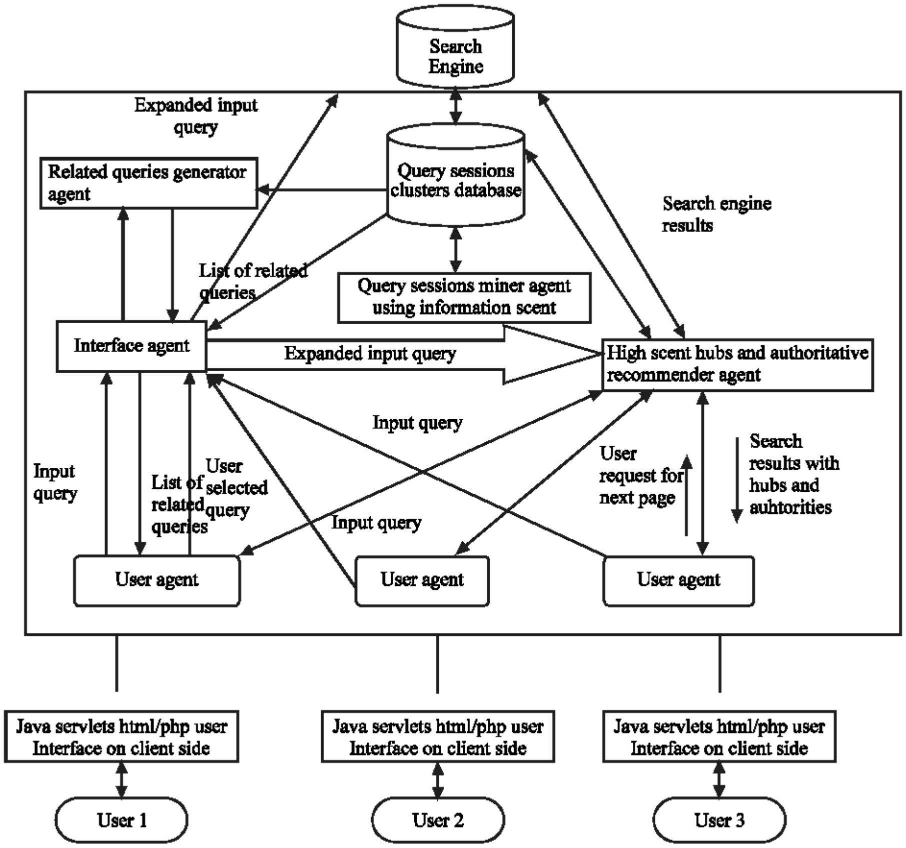
Search Engine
Expanded input query
Related queries generator agent
Query sessions clusters database
Search engine results
List of related queries
Query sessions miner agent using information scent
Interface agent
Expanded input query
High scent hubs and authoritative recommender agent
Input query
Input query
List of related queries
User selected query
Input query
User request for next page
Search results with hubs and auhtorities
User agent
User agent
User agent
Java servlets html/php user Interface on client side
Java servlets html/php user Interface on client side
Java servlets html/php user Interface on client side
User 1
User 2
User 3

Fig. 2: The proposed architecture

The blackboard is usually employed in tightly-coupled multi-agent systems. This shared-memory communication mechanism offers low overhead, predictability and high reliability, which are desirable features for use in real-time applications.

The above architecture is the proposed approach of improving the precision of search engine results using the High Scent Hub and Authoritative web page recommendations obtained from web query sessions mining using Information Scent.

The Web search of the user is personalized to effectively satisfy the information need of user inferred from the input query sense disambiguation and current query session of the user using the following Agents:

- User Agent
- Interface Agent
- Related Queries Generator Agent
- Query Sessions Miner Agent using Information Scent
- High Scent Hubs and Authoritative Recommender Agent

User Agent

Is an Agent representing the user who has logon to the proposed System for searching the Web for the specific information need of the user. User Agent keep track of clicked URLs of the query session of the user to infer the information need of the user from his clicked URLs in Suruchi and Punam (2007) and send both clicked URLs and partial query session vector inferring his information need to the High Scent Hubs and Authoritative Recommender Agent for mining in Query sessions database and Hubs and Authorities Recommendations.

During the online search it sends the input query to the Interface agent for input query sense disambiguation and finally sends the user selected query to Interface agent for information retrieval.

Interface Agent

Disambiguates the context of the input query of the user by presenting the recommended related queries in the domain similar to that of input query. The user selected query is used to find the cluster which has most similar information need as that of user selected input query and expanded using those keywords of the selected cluster which correlates best with input query (Suruchi and Punam, 2008a-c). The user selected expanded query is finally issued by Agent to the search engine for retrieving the search engine results.

Related Queries Generator Agent

Generates the related Queries recommendation using the clusters of query sessions. Each cluster represent the set of similar information need Query sessions. Each Query session vector in the cluster is associated with input Query and clicked URLs. The keywords of the input query sent by the Interface Agent is used by this Agent to find the cluster which best represent the information need similar to that of input query. The selected cluster is used to generate the recommendation of related queries associated with the Query sessions of the cluster (Suruchi and Punam, 2008a-c).

Query Sessions Miner Agent using Information Scent

Provides the bedrock on which other agents work. This agent mines the query sessions database which is collected from the user Agents. Each query session is identified by the query session id, query keywords and the clicked URLs associated with the input query. Information Scent is calculated for each clicked page of the query sessions to model the information need associated with each query session using Eq. 3.

The query session miner agent clusters the query sessions vector using clustering algorithm in order to group the similar query sessions in clusters/subclusters. Each cluster represents the unique information need identified in query session mining. Each cluster/subcluster is associated with input queries, High Information Scent clicked URLs.

Each subcluster is further preprocessed to find the High Scent Hubs and Authorities using clicked URLs associated with the subclusters of the cluster. HITS Algorithm is modified to use the Information Scent of the clicked URL to generate the Hubs and Authorities score for each web page in the Web Graph associated with the processed subcluster. Web pages having High Authority score relative to Hub Score is considered as Good Authority and Vice Versa (Suruchi and Punam, 2008b).

Thus each cluster/subcluster is associated with the High Scent Hubs and Authoritative pages for the information need associated with the cluster/subcluster. This Agent does its processing periodically offline.

High Scent Hubs and Authoritative Recommender Agent

Uses the expanded input query sent by the Interface Agent to find the cluster/subcluster in the Query session Database which best represent the information need associated with the input query. Selected cluster/sub-cluster is used to generate the High Scent Hub and Authoritative web pages for the input query that are recommended with the search engine results from Google and send it to the user.

During the search session of the user, User Agent send the captured clicked URLs and partial query session vector of the user to the Hubs and Authoritative Recommender Agent when the request of next result page is made to it. The clicked URLs associated with the query sessions of the user is stored in the Query session data base by Hubs and Authoritative Recommender Agent for mining in future. The partial query session vector is used to find the cluster/subcluster which closely represent the information need of the user and recommend the High Scent Hubs and Authorities associated with the selected cluster/subcluster with the next search engine result page to the user (Suruchi and Punam, 2008c).

Working of the Proposed Architecture

Step 1: The Java Servlets provides the interface which enables the user to get the services of the proposed architecture

Step 2: The user who is required to search the web through the proposed architecture is required to log on

Step 3: For each user who is log on user agent will be created which will keep track of the user search behavior during search sessions

Step 4: When the user issue the input query it is transferred to the Interface agent which will disambiguate the context of the input query with the help of Related Queries Recommender Agent which will retrieve the related queries for a given input query

Step 5: The user will select the query which will more closely represent his information need

Step 6: The user selected query will be expanded with related terms and sent by the Interface agent to the Search engine to retrieve the search results

Step 7: High Scent Hubs and Authoritative Recommender Agent will use the expanded input query if first time request for page is made /current user query session vector of the user if the request for next page request is made to retrieve the High Scent Hubs and Authorities associated with the selected cluster/subcluster and recommend it with the search results to the user agent which will display it to the user interface

Step 8: When the user request the next result web page of the current web search the user agent will send the request to High Scent Hubs and Authoritative Recommender search agent for the next result web page

Step 6: Goto step 7 until the user terminates the current search session

Step 10: If the user issue the new input query goto step 4

The proposed framework has four fold advantages for personalized web search. First is the use of Information Scent to infer the information need of the user from his clicked pattern of URLs in query session and used it to find the query sessions with similar information need and group them in clusters where each cluster represent the similar information need query sessions in a specific domain. Second it exposes the part of surface Web during web search in the form of Hubs and Authorities which are rich source of information in the particular domain but is hidden to the user due to absence of keywords used in user input query and Third it crawls the web in domains of each cluster using High Scent Clicked URLs to identify the High quality domain specific Hubs and Authorities. It is evident (Davison, 2000) that topical locality of pages mirror spatial locality in the Web i.e., WWW pages are typically linked to other pages with similar content. It is found that pages are significantly more likely be related topically to pages to which they are linked. The quality of Hubs and Authorities generation has been improved as it has been generated from those clicked URLs which satisfy the information need of the users in the particular domain identified using their Information Scent and Content of clicked URLs. Fourth one is that web search of the user is personalized using context disambiguation through query recommendation and expansion which overcome the problem of input query sense ambiguation as the keywords of input query are too few and sometimes ambiguous to infer the initial information need of the input query and tracks the behavior of user during web search to further infer his information need. The search is better personalized by tracking the user search browsing patterns to search results in order to obtain the partial query session vector which will further infer the information need of the user to better personalize his search with High Scent Hubs and Authorities Recommendation.

EXPERIMENTS

Experiment was conducted on the data set collected using the proposed Architecture for Information Retrieval System on the Web. The data set is generated by capturing the clicks of the anonymous users issuing input queries in various domains mainly Academics, Entertainment and Sport on the Proposed Architecture. The Proposed System architecture was tested using Google search engine results as Google Search Engine is rated as the best search engine among all available and most widely used. Improvement in the precision of Google search engine results was the Landmark for evaluating the Effectiveness of proposed architecture of Information Retrieval with Information Scent and Agents.

The users issue the queries in each of the selected domain to retrieve from the Google search engine results with no recommendation from system initially. The users click to the URLs of the result page was captured by the user agents associated with users and is stored as the query session of the user in the database.

Each query session was associated with the Query session id, input query and clicked URLs. The Query Session Miner Agent was invoked after collecting the sufficient query sessions from the users. The Query session Miner Agent compute the Information Scent associated with each clicked URLs in the Query sessions. The query sessions vectors were generated for each query session using Information Scent and content of clicked URLs.

The number of distinct URLs in data set was found to be 2995. The data set was preprocessed to get 595 query sessions. The query sessions vectors were then clustered using Leader-SubLeaders Algorithm. In this experiment similarity of any two query sessions was calculated using cosine measure. The Threshold value for Leaders computation was set to 0.5 and the SubThreshold value for SubLeaders computation was set to 0.75.

Every cluster/subcluster was identified by cluster id and mean keyword vector associated with it and was associated with input queries and the clicked URLs of the query sessions in it.

The HITS Algorithm was implemented on each subcluster of clusters to identify the High Scent Hubs and Authorities for specific Information need associated with the clusters using the Information Scent of the clicked URLs of the cluster. The web pages in the base set of clusters were given their initial hub and authority score which was the Information Scent of the web page. The web pages having their high final hub score relative to authority score were identified as High Scent Hub Web pages and vice versa.

The Hubs and Authorities associated with each subcluster were stored in the database along with their Information Scent value for retrieval during online processing. The above stated operations were performed by Query Session Miner Agent using Information Scent periodically at regular interval of time during offline processing.

The experiment was performed on randomly selected test queries shown in Table 1 which were categorized into trained queries set and untrained queries set. The trained queries were those input queries which had sessions associated with them in data set and untrained queries were those input queries which did not have sessions associated with them in data set.

The experiment was performed on Pentium IV PC with 2 GB RAM under Windows XP using JSP, JADE and Oracle database. WebSphinx crawler was used to fetch the clicked documents of query sessions in the data set. WVTool provides the classes to obtain the content vector of clicked documents and an interface to the Web Sphinx Crawler. JADE is the Java Agent Development Environment implements the Agent as classes and implement their action autonomously through their behavior list. Each query session was transformed into the vector representation using Information Scent and content of clicked URLs. The query sessions were clustered to identify the various information needs associated with each cluster/subcluster on the Web.

In online processing the input query issued to the proposed system generated the Google search engine result page with the High Scent Hubs and Authorities Recommendation for satisfying the information need of the users effectively and to improve the precision of search engine results.

Table 1: Sample of queries taken in each of the category

Category	Queries
Untrained set	Java online tutorial, MovieSong, Spacefood, novels, magazine, movies, familyplay games, movie pictures, India Football team, free software download
Trained set	Homeloan, distanceeducation online, free pics, OOPS tutorial,how to play. vcd files, mpeg movies, dragonball, intranet

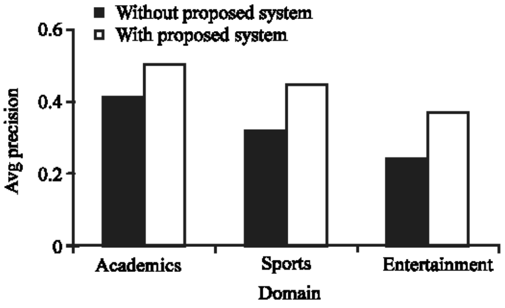

Fig. 3: Average precision of search results without proposed and proposed system on untrained queries

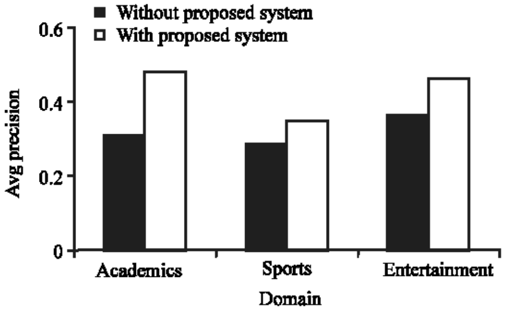

Fig. 4: Average precision of search results without proposed and proposed system on trained queries

The effectiveness of the proposed approach was determined by the average precision of the trained and untrained set of queries in each of the selected domain using the Google Search engine with proposed system and without proposed system.

Figure 3 and 4 show the improvement in the average precision of the search results with Proposed System as compared to the use of Google search engine results without proposed system which confirms the effectiveness of the proposed system in satisfying the information need of the user at the High precision of the Google search engine results. The improvement in average precision of search results shows that precision of search results can be improved to great extent if the Information Retrieval process on the web is supplemented with the effective identification of the information need of the users on the Web through Query Recommendation and expansion using Information Scent and by augmenting the search results with the High Quality Web Pages in the form of Hubs and Authorities which are sometimes hidden and could not be retrieved due to absence of keywords of the input query of the user. Furthermore Quality of Hubs and Authorities have been improved as they are identified using High Scent Clicked URLs which satisfy the similar information need of query session in the specific domain associated with the clusters. Due to this precision of search results has been improved by increasing the number of relevant documents in proportion to the number of retrieved documents.

CONCLUSION

The architecture of the Information Retrieval System with Information Scent and Agents presented in this study is the step in the direction of satisfying the information need of the user effectively and improving the precision of the search engine results. The proposed architecture provides the integrated approach to personalize the web search effectively using input query sense disambiguation with Query Recommendation and Expansion. Search is further personalized using High Scent Hubs and Authorities Recommendation. Interface agent after receiving the input query generates the input query recommendations to disambiguate the context of the input query. This is accomplished by clustering the Query sessions of clicked pages using Information scent. The clusters which closely approximate the information need of input query are used to recommend queries for a given input query. The disambiguated input query selected by the user is further expanded using related keywords to generate the High Scent Hubs and Authorities recommendations along with the Google search engine results. The user's responses to the result page are tracked to infer his information need through user profile and further personalize the search with Web Page Recommendations of High Scent Hubs and Authorities with search results. Experiments were conducted on the data set generated by capturing the clicked URLs of the users input queries in various domains through the proposed system implemented using JADE, JSP and Oracle. The improvement in the average precision of search results shows that proposed system proves to be effective in satisfying the information need of the users associated with the input queries by augmenting the search results with domain specific High Scent Hubs and Authorities Recommendation along with the input query context disambiguation.

REFERENCES

Agichtein, E., E. Brill and S. Dumais, 2006. Improving web search ranking by incorporating user behaviour information. Proceedings of the 29th Annual International ACM Conference on Research and Development on Information Retrieval, Aug. 06-11, Seattle, Washington, USA., pp: 19-26.

Ahn, J.W., P. Brusilovsky and R. Farzan, 2005. Investigating users' needs and behavior for social search. Proceedings of Workshop on New Technologies for Personalized Information Access at 10th International User Modeling Conference (UM'05), July 24-30, Edinburgh, UK., pp: 1-12.

Allan, J., 1996. Incremental relevance feedback for information filtering. Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Aug. 18-22, Zurich, Switzerland, pp: 270-278.

Armstrong, R., D. Freitag, T. Joachims and T. Mitchell, 1995. WebWatcher: A learning apprentice for the World Wide Web. Proceedings of the AAAI Spring Symposium on Information Gathering, from Heterogeneous, March 1995, Stanford, CA, pp: 1-7.

Asnicar, F.A. and C. Tasso, 1997. ifWeb: A prototype of user model-based intelligent agent for document filtering and navigation in the world wide web. Proceedings of Workshop Adaptive Systems and User Modeling on the World Wide Web, June 2-5, Chia Laguna, Sardinia, Italy, pp: 3-12.

Baeza-Yates, R. and B. Ribeiro-Neto, 1999. Modern Information Retrieval. Addison Wesley, London.

Bedi, P. and S. Chawla, 2007. Improving information retrieval precision using query log mining and information scent. Inform. Technol. J., 6: 584-588.

Bedi, P., R. Sharma and H. Kaur, 2009. Recommender system based on collaborative behavior of ants. Int. Artif. Intell., 2: 40-55.

Berkhin, P., 2002. Survey of Clustering Data Mining Techniques. Accure Software Inc., San Jose, CA., USA.

Billsus, D. and M. Pazzani, 2000. User modeling for adaptive news access. User Model. User-Adapted Interact., 10: 147-180.

Chen, L. and K. Sycara, 1998. WebMate: A personal agent for browsing and searching. Proceedings of the 2nd International Conference on Autonomous Agents, May 10-13, ACM Press, New York, pp: 132-139.

Chi, E.H., P. Pirolli, K. Chen and J. Pitkow, 2001. Using information scent to model user information needs and actions on the web. Proceedings of ACMCHI Conference on Human Factors in Computing Systems, (HFCS'01), New York, USA., pp: 490-497.

Claypool, M., A. Gokhale, T. Miranda, P. Murnikov, D. Netes and M. Sartin, 1999. Combining content-based and collaborative filters in an online newspaper. Proceedings of the ACM SIGIR Workshop on Recommender Systems-implementation and Evaluation, Aug. 19-19, ACM Press, USA., pp: 1-8.

Cutting, D., J. Carger, J. Pedersen and J. Tukey, 1992. Scatter/gather: A cluster-based approach to browsing large document collections. Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, June 21-24, Copenhagen, Denmark, pp: 318-329.

Davison, B.D., 2000. Topical locality in the Web. Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval Athens, July 24-28, Athens, Greece, pp: 272-279.

Direct Hit, 1995. Popularity-based search. http://www.directhit.com/.

El-Korany, A. and K. EL-Bahnasy, 2009. A multi-agent framework to facilitate knowledge sharing. J. Artificial Intel., 2: 17-28.

Etzioni, O. and D.S. Weld, 1995. Intelligent agents on the Internet: Fact, fiction and forecast. IEEE Expert, 10: 44-49.

Ferragina, P. and A. Gulli, 2005. A personalized search engine based on web-snippet hierarchical clustering. Proceedings of the Special interest tracks and posters of the 14th international conference on World Wide Web, May 10-14, ACM Press, New York, USA., pp: 801-810.

Franklin, S. and A. Graesser, 1996. Is it an agent or just a program? A taxonomy for autonomous agents. Proceedings of the 3rd International Workshop on Agent Theories, Architectures and Languages, (ATAL'96), Springer-Verlag, New York, pp: 1-10.

Freyne J., B. Smyth, M. Coyle, E. Balfe and P. Briggs, 2004. Further experiments on collaborative ranking in community-based web search. Artif. Intell. Rev., 21: 229-252.

Gentili, G., A. Micarelli and F. Sciarrone, 2003. InfoWeb: An adaptive information filtering system for the cultural heritage domain. Applied Artif. Intell., 17: 715-744.

Georgeff, M.P., 1984. A theory of action for multi-agent planning. Proceedings of the 4th National Conference on Artificial Intelligence, Aug. 6–10, Austin, TX, pp: 125-129.

Gudivada, V.N., V.V. Raghavan, W. Grosky and K.R. Gottu, 1997. Information retrieval on world wide web. IEEE Internet Comput., 1: 58-68.

Heer, I. and E. Chi, 2001. Identification of web user traffic composition using multi-modal clustering and information scent. Proceedings of the 1st SIAM ICDM Workshop on Web Mining, 2001, Chicago, IL., pp: 51-58.

Jain, A.K., M.N. Murty and P.J. Flynn, 1999. Data clustering: A review. ACM Comput. Surveys, 31: 264-323.

Jansen M., A. Spink, J. Bateman and T. Saracevic, 1998. Real life information retrieval: A study of user queries on the Web. ACM SIGIR Forum, 32: 5-17.

Joachims, T., D. Freitag and T. Mitchell, 1997. WebWatcher: A tour guide for the world wide web. Proc. Int. Joint Conf. Artif. Intel., 1: 770-775 (In Japanese).

Joachims, T., 2002. Optimizing search engines using clickthrough data. Proceedings of 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, July 23-26, Edmonton, Alberta, Canada, pp: 133-142.

Kim, J.H., H.S. Shim, H.S. Kim, M.J. Jung, I.H. Choi and J.O. Kim, 1997. A cooperative multi-agent system and its real time application to robot soccer. Proceedings of the IEEE International Conference on Robotics and Automation, April 1997, Albuquerque, New Mexico, pp: 638-643.

Kleinberg, J.M., 1998. Authoritative sources in a hyperlinked environment. Proceedings of the 9th Annual ACM-SIAM Symposium on Discrete Algorithms, Jan. 25-27, San Francisco, California, United States, Society for Industrial and Applied Mathematics, pp: 668-677.

Koutrika, G. and Y. Ioannidis, 2005. A unified user profile framework for query disambiguation and personalization. Proceedings of the Workshop on New Technologies for Personalized Information Access, July 24-25, Edinburgh, Scotland, UK., pp: 44-53.

Kritikopoulos, A. and M. Sideri, 2005. The Compass filter: Search engine result personalization using web communities. Lecture Notes Comput. Sci., 3169: 229-240.

Lieberman, H., 1995. Letizia: An agent that assists web browsing. Proceedings of the 14th International Joint Conference on Artificial Intelligence, Aug. 1995, Montreal, Canada, pp: 924-929.

Lieberman, H. and T. Selker, 2000. Out of context: Computer systems that adapt to and learn from, context. IBM Syst. J., 39: 617-632.

Liu, F., C. Yu and W. Meng, 2004. Personalized web search for improving retrieval effectiveness. IEEE Trans. Knowl. Data Eng., 16: 28-40.

Micarelli, A. and F. Sciarrone, 2004. Anatomy and Empirical evaluation of an adaptive web-based information filtering system. User Model. User-Adapted Interact., 14: 159-200.

Middleton, S.E., D.C.D. Roure and N.R. Shadbolt, 2001. Capturing knowledge of user preferences: Ontologies in recommender systems. Proceedings of the 1st International Conference on Knowledge Capture, Oct. 2 2-23, Victoria, British Columbia, Canada, pp: 100-107.

Newell, A., 1982. The knowledge level. Artif. Intell., 18: 87-127.

Olston, C. and E.H. Chi, 2000. ScentTrails: Integrating browsing and searching on the World Wide Web. ACM Trans. Hum. Comput. Interact., 10: 177-197.

Pattie, M., 1994. Social Interface Agents: Acquiring Competence by Learning from users and other Agents. In: Software Agents, Etzioni, O. (Ed.). AAAI Press, Stanford, CA., pp: 71-78.

Pazzani, M., J. Muramatsu and D. Billsus, 1996. Syskill and webert: Identifying interesting web sites. Proceedings of the 13th National Conference Artificial Intelligence, (NCAI'96), China, pp: 54-59.

Pirolli, P., 1997. Computational models of information scent-following in a very large browsable text collection. Proceedings of the ACMCHI Conference on Human Factors in Computing Systems, (HFCS'97), New York, USA., pp: 3-10.

Pirolli, P. and S.K. Card, 1999. Information foraging. Psychol. Rev., 106: 643-675.

Pirolli, P., 2004. The use of proximal information scent to forage for distal content on the world wide web. In: Working with Technology in Mind: Brunswikian Resources for Cognitive Science and Engineering, Kirlik, A. (Ed.). Oxford University Press, Oxford.

Pitkow, J.E., H. Schtze, T.A. Cass, R. Cooley and D. Turnbull *et al*., 2002. Personalized search. Commun. ACM, 45: 50-55.

Raghavan, V.V. and H. Sever, 1995. On the reuse of past optimal queries. Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval, July 09-13, Seattle, Washington, United States, pp: 344-350.

Rhodes, B.J., 2000. Just-in-time information retrieval. Ph.D. Thesis, MIT Media Laboratory, Cambridge, MA.

Salton, G., 1983. An Introduction to Modern Information Retrieval. Mc-Graw-Hill, New York.

Shoham, Y., 1997. Agent Oriented Programming: A Survey. In: Software Agents, Bradshaw, J.M. (Ed.). MIT Press, Menlo Park, CA.

Speretta, M. and S. Gauch, 2005. Personalized search based on user search histories. Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence, Sept. 19-22, IEEE Computer Society Washington, DC, USA., pp: 622-628.

Sun, J.T., H.J. Zeng, H. Liu, Y. Lu and Z. Chen, 2005. CubeSVD: A novel approach to personalized web search. Proceedings of the 14th International Conference on World Wide Web, May 10-14, ACM Press, New York, USA., pp: 382-390.

Suruchi, C. and B. Punam, 2007. Personalized web search using information scent. Proceedings of the International Joint Conferences on Computer, Information and Systems Sciences and Engineering, Dec. 3-12, Bridgeport University, USA., pp: 61-67.

Suruchi, C. and B. Punam, 2008a. Finding Hubs and authorities using Information scent to improve the information retrieval precision. Proceedings of the 2008 International Conference on Artificial Intelligence, July 14-17 Las Vegas, NV, USA., pp: 185-191.

Suruchi, C. and B. Punam, 2008b. Improving information retrieval precision by finding related queries with similar information need using information scent. Proceedings of the 1st International Conference on Emerging Trends in Engineering and Technology, July 16-18, IEEE Computer Society Press, New York, pp: 486-491.

Suruchi, C. and B. Punam, 2008c. Query expansion using information scent. Proceedings of the International Symposium on Information Technology, Aug. 26-29, IEEE, Kuala Lampur, Malyasia, pp: 339-344.

Tasso, C. and P. Omero, 2002. La Personalizzazione Dei Contenuti Web: E-commerce, I-access, Egovernment. Franco Angeli, Milano, Italy.

Vijaya, P., 2004. Leaders-subleaders: An efficient hierarchical clustering algorithm for large data sets. Pattern Recognit. Lett., 25: 505-513.

Wen, R.J., Y.J. Nie and J.H. Zhang, 2002. Query clustering using user logs. ACM Trans. Inform. Syst., 20: 59-81.

Zamir, O.E., J.L. Korn, A.B. Fikes and S.R. Lawrence, 2004. Personalization of placed content ordering in search results. United States Patent Application 20050240580. http://www.freepatentsonline.com/y2005/0240580.html.

Zhao, Y. and G. Karypis, 2002. Comparison of agglomerative and partitional document clustering algorithms. The SIAM Workshop on Clustering High-Dimensional Data and Its Applications, Washington, DC, April 2002.

Zhu, T., R. Greiner and G. Häaubl, 2003. An effective complete-web recommender system. Proceedings of the 12th International World Wide Web Conference, May 20-24, Budapest, Hungary, pp: 1-9.