Agent Based Personalized Semantc Web Information Retrieval System

Dr.M.Thangaraj, Associcate Prof.
Department of Computer Science
Madurai Kamaraj University ,Madurai , Tamilnadu

Mrs.Mchamundeeswari, Assoicate prof.
Department of Computer science
V.V.V College for Women (Affiliated to Madurai Kamaraj university)
Virudhunagar,Tamil Nadu

***Abstract*—Every user has an individual background and a precise goal in search of information. The goal of personalized search is to search results to a particular user based on the user's interests and preferences. Effective personalization of information access involves two important challenges: accurately identifying the user context and organizing the information to match with the particular context. In this paper, the system uses ontology as a knowledge base for the information retrieval process. It is one layer above any one of search engines retrieve by analyzing just the keywords. Here, the query is analyzed both syntactically and semantically. The developed system retrieves the web results more relevant to the users query. The level of accuracy will be enhanced since the query is analyzed semantically. The results are re-ranked and optimized for providing the relevant links. Based on the user's information access behavior, an ontological profile is created, which is also used for personalization. If the system is deployed for web information gathering, search performance can be improved and accurate results can be retrieved.**

Keywords*—Agent; Personalization; Semantic web; information retrieval; ranking algorithm

I. Introduction

The main purpose of this section is to justify the need for an integrating approach that combines both intelligent agents and personalized semantic web service technologies. The study concentrates on personalized semantic web services and then intelligent agents and multiagent systems which are enumerated and the most pressing problems of agent technology pointed out.

A. Personalization using Semantic web:

Semantic technologies promise a next generation of semantic search engines. General search engines don't take into consideration the semantic relationships between query terms and other concepts that might be significant to the user. Thus, semantic web vision and its core ontology's are used to overcome this defect. The order in which these results are ranked is also substantial. Moreover, user preferences and interests must be taken into consideration so as to provide the user a set of personalized results.

B. Query Expansion using ontology:

Ontology is to create a shareable and agreeable semantic resource over a wide range of agents. The important goal of building ontology is it may serve as an index into a repository of information to facilitate information search and retrieval and also used to identify the user context accurately, so that the search results can be personalized by reorganizing the results returned from a search engine for a given query. In this research, context is extracted from Domain Ontology in terms of concepts and used to extract the semantic patterns in queries which can represent actual users' requirement.

Through personalization, one can improve the navigation on a web site by, for example, highlighting content and links of interest, hiding those that are irrelevant, and even providing new links in the site to the users likely web destinations. While personalization can help to identify relevant new information, new information can create problems in re-finding when presented in a way that does not account for previous information and interactions. This study presents a model of what people remember about search results, and shows that it is possible to merge new information invisible into previously viewed search result lists where information has been forgotten. Personalizing repeat search results in this way enables people to effectively find both new and old information effectively using the same search result list.

C. Agent based personalization:

The main characteristic of agent-based technology is that the structure of the software is represented by a group of agents who collaborate in achieving the goal of the task in hand. The combination of information retrieval and Multi-agent technology has the following features: .Adaptability, initiative and collaborative. Among different types of agents, the personal assistant agents are particularly interesting to this research. This type of agents operates at the user interface level and actively assists users by offering information and advice to the users (Wasson *et al.*, 2001). These agents usually apply a kind of intelligent learning algorithm so that they can intercept the users input, examine it and take actions that are more specific to those particular users' needs at that moment. These agents are also called learning or adaptive agents. Agent can initiatively retrieve the corresponding information based on users' demand, and even can monitor the changes of information sources and agents also share the information with other Agents.

This paper introduces a personalized information retrieval system based on multi-agent, which can accomplish information retrieval according to user interest knowledge via multi-agent collaboration for providing personal service to the user. In the process of personal information retrieval, the precision and quality depend on the veracious degree that the system master user interest. Therefore, the paper solves problems how to construct user interest model based on vector space, and how to update user interest model in time when user's interest changes:

II. Literature Review

Web personalization is understood in various dimensions. One way of doing this is categorization of users based on demographic information provided by the users at the time of selecting the style for personalization. An example of this is Google Personal search through ***igoogle***. This approach requires that the user must exactly know what information is needed prior to searching. The research is also going on to modify the structure of the web documents and make it semantic so that the documents are then retrieved on the basis of the meaning of the query and not the terms present in the query [2]. This approach seems very promising but is a long term project, the acceptability and usability of which depends on the user community. Another way to personalize the search is to classify users on the basis of pre-calculated classes. The classes may be pre-calculated through users browsing history. A classification of the on-line users to one of the predefined classes is typically based on similarity calculation between each predefined pattern and the current session. The current session is assigned to the most similar cluster [6, 8]. Further this approach is modified to accommodate fuzzy classification so as to prevent some users to become outliers [7].

Some authors have constructed user profiles on the basis of modified collaborative filtering with detailed analysis of user's browsing history in one day [5]. User profiles are also constructed on the basis of ontology [1]. Some efforts have also been done to refine the search process by re-defining the queries and then submit it to the search engine. Refined queries are then clustered to form user's profiles [6]. In this approach also only visiting a page makes it interesting enough to update user profile. Another method to personalize is to discover association among various links accessed by the user through its sessions [3].

Another interesting effort has been done in actual personalization of users' interest in which they have considered that every user's behavior is different on same search results obtained through same search query [3]. They have used two properties of a document for modeling users i.e. attractiveness and perseverance. They have assumed that these properties depend on the popularity of the document among the similar user community and distance of that document from last selection. Normal user behavior suggests that after a certain no of unattractive documents the user stops navigating the search results. Efforts have also been done to construct user profiles using relevancy between the terms of the queries presented in current session and in earlier sessions [4].

Due to the intelligent agent technologies shortcomings, the inherent need for autonomous software entities in SWS environments, and the promising benefits of having both intelligent agents and (semantic) web services working cooperatively, numerous research projects have been carried out that try to put these two technologies together into integrated frameworks .The author Hendler (2001) proposes a method for describing the way the invocation of services should be done by agents by means of an ontology language. The Semantic Web FRED project (SWF) combines agent technology, ontologies, and SWS in order to develop a system for automated cooperation. The GODO (Goal-Oriented Discoveryfor SWS) system (Go´mez et al., 2006), which is based on a software agent that is located between different SWS execution environments (e.g. WSMX, METEOR-S,OWL-S Virtual Machine, etc.) and final users.

The authors Buhler and Vidal [10] highlight the passive behavior of web services and propose to wrap them in proactive agents. The problem of this approach is that semantically described web services are not considered at all. Another related solution is the one provided by the ''Agents and Web Services Interoperability Working Group (AWSI WG)"3, which is part of the IEEE FIPA Standards Committee which can handle the fundamental differences between agent technology and web services, that is, the use of different communication protocols (ACL vs. SOAP), service description languages (DF-Agent-Description vs. WSDL) and service registration mechanisms (DF vs. UDDI). With this approach, the so called Agent Web Gateway middleware (Shafiq et al., 2006) facilitates the required integration without changing existing specifications and implementations of both technologies. This category focuses on the overlapping features of the technologies under question. However, we believe that most of the functionality provided by Intelligent Agents and Web Services is complementary, so that each of these technologies must be situated at a different abstraction level.

The model proposed here is a frame work for building a user model in addition to explicit & implicit feedback from user and find the relevancy between the terms presented for query and the document using past sessions by user and the contents of the documents. Then the documents with the higher relevance ratio are presented to the user. The current user session data is used to update the user's profile for future reference. Two types of parameters are considered for constructing user model: static parameters and dynamic parameters. Static parameters are relevancy of documents with the specific category measured by the popularity of the document. Static parameter used in the model is: Term-document relevancy which is maintained in a 2-D matrix T. The relevancy calculation done here is based on the occurrence of terms in the document. We have tried to improve upon the modality of updating the matrix. The matrix is updated every with every user session with the browsing patterns of a user and for first 'n' sessions it keeps on constructing new columns with respect to the terms that relates to a document. Our system proposes a different ranking based on URL. The ranking is *query-dependent. The proposed* algorithm assigns a score that measures the quality and relevance of a selected set of pages depending on their URL to a given user query. The basic idea is to build a query-specific two dimensional vector table, called a related vector table, and perform URL analysis. The present paper proposes a slightly different ranking based on URL. In our research we use hybrid approach to find ranked webpage.

III. PROPOSED ARCHITECTURE:

This paper we proposes, an architecture for an Agent Based Personalized Semantic Web Information Retrieval (APSIR), which can help users to get the relevant web pages based on their selection from the domain list , so that users can obtain a set of related web pages from the system .

APSIR is a crawler-based search engine that makes use of crawler to collect resources from both semantic as well as traditional web resources

This section explains the basic architecture of our system. In section III the working mechanism of the proposed system is describe. Section IV shows the performance evaluation results. Finally, section V sums up all the above said points.

The system APSIR (in Figure 1) consists of different components like User Agent, Semantic *Extraction Agent, and* Semantic *Searching Agent, Filtering Agent, Personalized Ranking Agent* and Knowledge Base. All agents are monitored entirely to fulfill proprietary system functions, including information retrieval and Knowledge Base update.

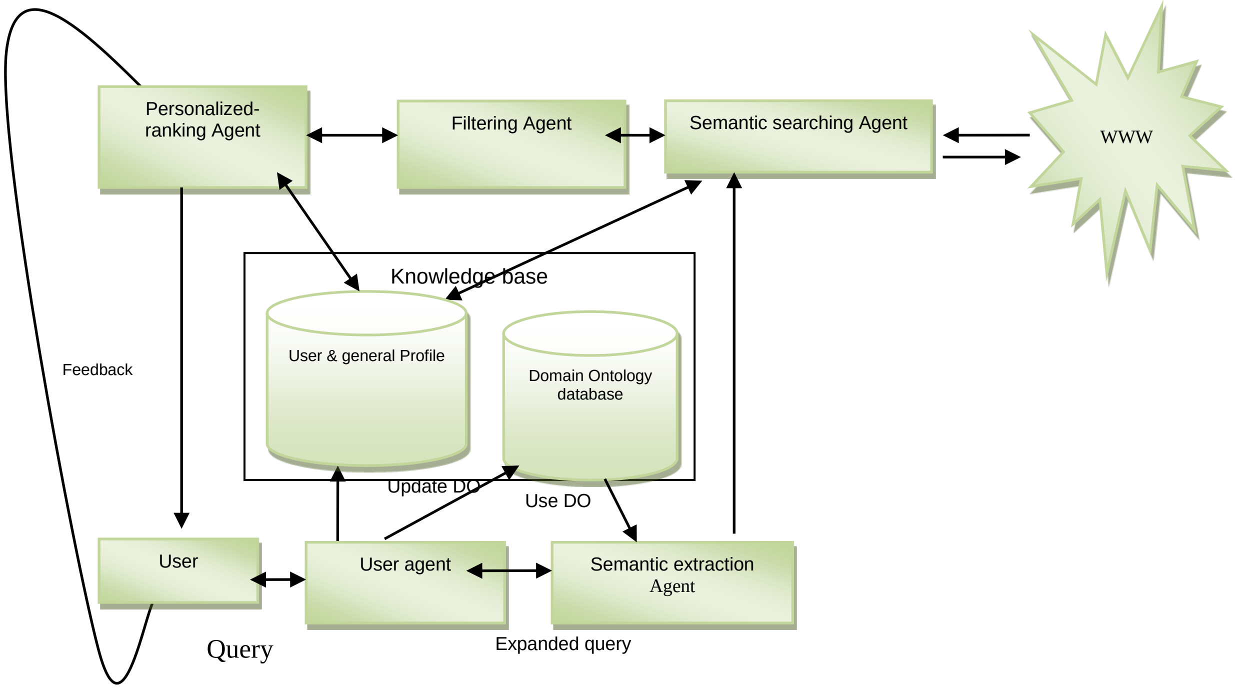

Fig. 1. Structure of Agent –based personalized Semantic web information retrieval system

A. *User Agent:* User Agent is the mutual interface between user and system, and provides a friendly platform to users. It can construct user interest model according to User's browsing history record and registration data. (System comes with an easy to use Google like search interface. After submitting his query, results are displayed.) User Agent incepts user's retrieval request (needed search) which is transformed to prescriptive format, and transmits the formatted user request to Semantic Extraction Agent to expand the query based on the respective domain and related terms based on ontology. User Agent also takes over all result from Personalized Ranking Agent, and presents personally these results to user. In addition, User Agent presides over creating a profile user for new user. User's browsing or evaluating behavior can also be stored as profile and it is learned by User Agent, so user interest model may be updated and improved in time. User Agent includes Environment view, Memory Base, Knowledge Base, Learning mechanism and Inference Engine.

- The view of the environment module in the User Agent is the user's input and output interface.
- Memory Base records the original information entered by the user.
- Knowledge Base defines the user's personal knowledge, classified information and the user model.
- Learning mechanism is used to summarize the behavior of users and formats the information.

B. *Semantic Extraction Agent:* Semantic Extraction Agent aims to find the semantic features in the users' queries. It will make use of agent technologies and ontology technologies to analyze the association relation in the users' queries and document to extract semantic features. This module contains the following components:

- Query preprocessing: Meaningless words like neuter pronouns, articles, and symbols in the content will be removed from query.
- Semantic Analyzing: This component identifies semantics elements like Subject, Property, and Object in the Query content and analyzes their semantic relations.
- Semantic matching : In the personalized information retrieval system, Semantic matching agent takes charge of receiving formatted user request from User Agent, and the user request is expanded (based on ontology)according to user interest. Afterwards, the perfected user request is transmitted to Semantic Searching Agent. It analyses the returned data from Searching Agent, filtrating useless information, and the processed results are send to user.

Alg. For QE using domain ontology

Input: Original query terms set (Q_{or}) where Q_{or} ={ t_1 ,t_2…t_n}

Output: Query terms set (Q_{set}) where Q_{or} Ù Q_{ex} Q_{ex} is the expanded query terms

```
 queryexpand($Q_{or}$)
{
$Q_{set}$={empty}
// expand query based on ontolgoy
Get $Q_{or}$ and add it to $Q_{set}$.  // split each word in the query and stored as $Q_{set}$
   for all term $t_i$ in $Q_{set}$
{
     If ( $t_i$ in $D_{ontology}$)
       {
         If( $t_i$ in $P_{ontology}$) and ( $t_i$ in $R_{ontology}$) // find its possible and related terms
           $Q_{ex}$=$P_{ontology}$+$R_{ontology}$
         }
 ElseIf  ( $t_i$ in $p_{ontology}$)
   {
     If  ( $t_i$ in $D_{ontology}$) and  ( $t_i$ in $R_{ontology}$ )
           $Q_{ex}$=$D_{ontology}$+$R_{ontology}$
   }
ElseIf  ( $t_i$ in $R_{ontology}$)
   {
     If  ( $t_i$ in $D_{ontology}$  ) and ( $t_i$ in $P_{ontology}$ )
           $Q_{ex}$=$D_{ontology}$+$P_{ontology}$
 }
}   // for loop
} // end
```

C. *Semantic Searching Agent:*

This component is responsible for searching and retrieving relevant results. Semantic Searching Agent mainly includes Search Strategy, Search Optimization and Crawler.

- Search Strategy includes depth-first search strategy
- Search Optimization includes the way accessing to the page that should be subjected to management of websites and the frequency of visiting, collecting important web pages which have high page weight and have changed, ensuring pages that will not be repeated crawled.
- Crawler is a program that crawl pages based on the hyperlinks between webs to collect information

```
Semantic search($Q_{or}$)
{
 Queryexpand($Q_{or}$ )      // expand user query based on ontolgoy
If the term is found
{
Retrieve the relevant web page from  in knowledge base/web
Store the query and web pages to db
}
}
Else   // new ontology term
{   Update ontolgoy with new terms
}
}d
```

D. *Filtering Agent:*

In this process, matching algorithms are presented to enable fast matching and searching for content. Components included in this module are: Knowledge Base and Semantic Matching.

Knowledge base: It includes the user's personalized information transmitted by the User Agent. When matching, Semantic Matching will make use of users personalize information (users' interest behavior and search history) to match and search more accurate and useful information for users.

Semantic Matching: According to the Users behavior, this component will match semantics in users' queries and semantics in the documents, and in accordance with the relevant, the results will be submitted to the User Agent.

E. *Personalized Ranking Agent:*

In the personalized information retrieval system, Personalized Ranking Agent is the decision-making center of personalized information retrieval system based on multi-agent. Using Re-ranking alg. find the new score based on users interest.

Algorithm of calculation of relevance score(re-ranking) for the Web pages.

Input : web page (P) Query term {t1,t1,t3…tn} from the expanded query

Output relevance score (ranking) for all the web page (WP)

```
Urlset={$url_1$,$url_2$…$url_n$ } for the given query
Re-rankign(urlset ,Qex)
{
For all url in the urlset
{
read webpage(p);
Query set {}←---  Separate each term in Qex
N=total number of term
I=1
If ti in WP
{
  TF =TF+∑D(Ti)
}
    For all terms $t_i$ in  Query set
   {
Wcount($t_i$)=wcount($t_i$)+$\sum_{ti..tn}\sum_{ti}wp_{i=1}$(D($t_i$))
    }
termcount=wcount/N
Count=termcount+TF
}
Fs=Get feedback score{0,1,-1}
ts=viewtime(WP)
Save count,f,c, ts in DB table
}
```

F. *Knowledge Base:*

Knowledge Base is used for storing every user interest model, user- record, and rules or parameters that serve for ensuring system well-balanced circulation.

G. *User interest profile:*

Two general methods are used to discover user interest (i) apparent feedback and(ii) connotative feedback.

In apparent feedback, user can input the data of personal interest or evaluation to current work.

1) Apparent feedback: When information retrieval, user gives a weigh value Wv, which represents user's satisfactory degree to the provided document D, formalized expression is described as follows.

$$\text{Satis\_De}(D) = f(Wv) \quad 0 \leq \text{Satis\_De}(D) \leq 1 \tag{1}$$

In system implement, user can select whether evaluation page appear or   evaluation page may appear constrainedly. The satisfactory degree setting may be an option bar , so user can adjust to set *Wv*.

2) Connotative feedback: The system may obtain user interest information via tracking user behavior and operation.

The under-mentioned factors may be used to discover impliedly user's interest. a) History record - User is interested in the pages, which are browsed before time, the more accessing times the higher interest degree. b) User behavior. Some operations (e.g. saving, printing or copying) indicate user interest when user browsing page. In addition, browse time are also related with user interest. So the mine above-mentioned data is to discover user's interest.

H. Construction of user interest Profile:

First of all, the following process may be used to classify browsing history record documents.

Step1: if QUERY match exactly (BH(browsing history)) which is standardized vector set of browsing history record.

Step2: For *i*=1; *j*=2, 3, …, |BH(*D*)|, calculate the relativity of document *Di* with document *Dj* in BH(*D*) set.

$$Sim(D_i,D_j)=W_i*W_j/|\, W_i|*|W_j| \qquad \textbf{(2)}$$

All documents *Dj* with *Sim*(*Di*,*Dj*)≥*thersold value* (*thersold value* is no. of web page to be displayed) . So the classified document vector set *S*1(*D*), *S*2(*D*), … ,*Sn*(*D*) are gained. All specific terms in document vector set *Si*(*D*) is sorted according to the weight descending.In this way, we can get a user interest vector User*Ini* = ((*ti1*, *wi*1), (*ti*2, *wi*2), … , (*tik*, *wik*), … , (*ti*,*DT_Limit*, *wi*,*DT_Limit*)) *wik* is standardized weight of specific term *tik*. Then, the user interest model is constructed as:

$$\textbf{User Interest Model}\ (UIM) = UserIn_1 \cup UserIn_2 \cup UserIn_3 \ldots.. \cup UserIn_n \qquad \textbf{(3)}$$

There, *n is* the classified set number of history record documents.

I. User behavior factor:

When collecting user interest information, should be paid attention to other factors. For example, user attitude to browsing page is very important factor of user interest information. Some pages are saved, some pages are copied, or some pages are printed. By all appearances, user is interested much more in those copied, saved, or printed pages related to merely browsed pages. So the users domain interest degree is introduced whereas before-mentioned reason. *User_Interest*(User*Ini*) denotes the degree of interest of interest domain User*Ini* that document belongs to. Therefore, Knowledge Base also stores user behavior data besides user interest model.

1) FreqInDi, which is the citing frequency of user interest domain UserIni that user browsed document belongs to, and

2) SaveInDi, which is the saving frequency of user interest domain UserIni that user saved document belongs to, and

3) SpeedInDi, which is the viewing timing of UserIni documents, and

So we can construct a numerical function of domain interest degree, which may reflect interest information of user behavior.

Fi (*FreqInDi*, *SaveInDi*, *SpeedInDi*) All interest domains may be resorted at any moment according to the function *Fi* of domain interest degree. In this way, changes of user interest along with time can be reflected in user interest model

J. Result storing and viewing :

Step1: Convert retrieval request query string *Q* to vector.

V(*Q*) = ((*qt*1, *qw*1), (*qt*2, *qw*2) , … , (*qtr*, *qwr*))

Step2: Construct document vector for any returned document *FDi*.(from browsing history)

V(*FDi*) = ((*fti*1, *fwi*1), (*fti*2, *fwi*2), … , (*fti*,*NT_Limit*, *fwi*,*NT_Limit*))

Step3: Calculate the comparability between document vector *V*(*FDi*) and query vector *V*(*Q*) .

$$Wsim(FD_i,Q)= nfW_i*nqWj/|\, nfW_i|*|nqW_j| \qquad \textbf{(4)}$$

There, *n* is total number of specific terms in query *Q or in document*.

Step4: Calculate the comparability between document vector *V*(*FDi*) and user interest vector *V*(User*Inj*) .

$$Psim(FD_i,Userin_j)= nfW_i*nWj\ /\ |\, nfW_i|*|nW_j| \qquad \textbf{(5)}$$

n is total number of specific terms in document *FDi* or user interest model

Then, the comparability of document *FDi*, query *Q* and user interest model. *User_Model* is represented as follow.

$$Sim(FDi,\ Q,\ User\_Model) = Wsim(FDi,\ Q) + Psim\ (FDi,\ Userin). \qquad \textbf{(6)}$$

If all returned documents are processed then go to Step5, otherwise go to Step2.

Step5: Output sorted searching results according to *Sim*(*FD*,*Q*, *User_Model*) value descending for the returned documents.

K. User interest model update

When user browses output documents, the system memorizes user's behavior (browsing, saving etc.) to Knowledge Base in real time. The system may give an evaluation page for asking user to do satisfactory degree .Evaluating all documents, which satisfactory degree *Satis_De*(*D*) is more than a default minimal value,(threshold value) is extracted for constructing new user interest domain vector.

New user interest domain vector is used to replace old user interest domain vector, which is cited seldom. The storage capacity of user interest model is commonly limited to finite space capability, for example: N_Class_MAXTIME. When the number of user interest domain vector exceeds the capability limit(>=particular time interval); some user interest domain vectors, which are cited seldom (scaling by domain interest degree function Fi (FreqInDoi, CopyInDoi, PrintInDoi, SaveInDoi, SpeedInDoi)), may be deleted from user interest model and moved to dump table. So the number of user interest domain vectors is limited to definite scope, and the system can track user interest in time.

IV. Implementation and Experimentation

In this section experiments carried out to evaluate the performance of proposed system will be discussed from a quantitative point of view by running some experiments to evaluate the precision of the results. The basic idea of the experiment is to compare the search result from keyword based search engine with proposed one on the same category and the same keywords. The criteria of our experiments include suitability (the ratio of the amount of useful information to the total amount of information) age (the period of the document post) and semantic matching (the accuracy of matching).After several time of similar information search, our system will get better results than the current search engine expectedly by updating user profile based on the user feedback autonomously. A test set collection is which consists of set of documents, queries and a list of relevance documents are used to evaluate the proposed system. These are used to compare the results of proposed system by performing relevance based evaluation method.

The proposed system is implemented in C#.Net as Web-based system using Visual Studio 2008, .NET Framework 3.5, and SQL Server 2005. The number of stored documents is more than 3 lakhs documents. These Web documents are about computer science domain. The improvement is measured by performing different experiments using the relevance based evaluation method . It uses the metrics: precision, recall, f-measure, average precession (AP) and mean average precision (MAP), to measure the performance of proposed system.

A set of queries has been manually for comparative performance measurement. The set of sample queries is given in Table 1. It Show the different levels of performance for different queries, the proposed semantic information retrieval method that improves the document ranking.

TABLE I. AP AND F-MEASURE USING PERSONALIZED AND UN-PERSONALIZED RANKING FOR SINGLE USER AND MULTIPLE USER

TABLE 1A: SINGLE USER

Keyword	AP		F-measure	
	existing Using keyword query	**Current Using semantic query**	**existing Using keyword query**	**Current Using semantic query**
Java	.76	0.56	.61	0.41
Constructor	1.00	0.81	.69	0.52
Polymorphism	1.17	1.00	.89	0.67
memory compaction	1.1	0.79	.58	0.53
Encapsulation	1.3	0.79	.69	0.53
disk space management	1.2	0.79	.76	0.53
abstract classes	.98	0.84	.89	0.56

TABLE 1B: MULTIPLE USER

User	AP		F-measure	
	existing Using keyword query	**Current Using semantic query**	**existing Using keyword query**	**Current Using semantic query**
User 1	.65	0.55	.55	0.41
User 2	.688	0.55	.99	1.08
User 3	.459	0.375	1.258	1.125
User 4	.65	0.375	1.356	1.125

Fig 2 and Fig. 3 shows comparative study of the results of the both systems that retrieves the documents based on similarity between the query and the collected documents. This experiment shows the average precession that is based on retrieving results for different query of single user and single query of multiple users. Graph shows that the system gives high precision during retrieving documents.

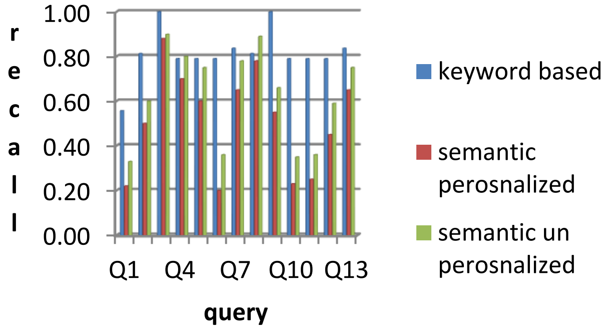

Fig. 2. AP measure precision of personalized vs. un personalized (for single user)

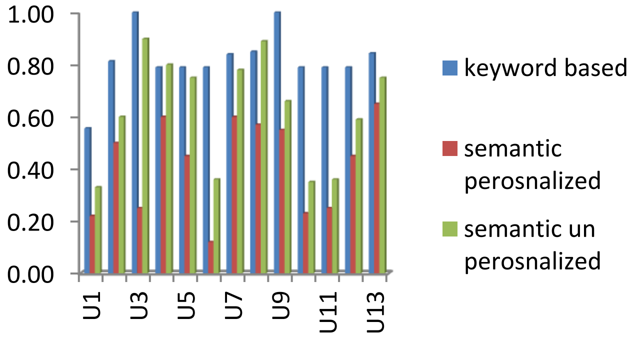

Fig. 3. AP measure precision of personalized vs. un-personalized (for multiple users with multiple keywords)

The retrieval efficiency is a major challenge when the size of the database increases. This shows the importance of semantic similarity during determining the documents that are relevant to the user query. The second sets of experiments, which are user centered, are focused on the overall performance of the search engine and the evaluation of real interactions with users. Fig.4 discusses the performance efficiency of both systems when the system uses to retrieve the result. This graph shows that agent based personalized search is better than other method because it highlight user profile and study user behavior to determine ranking for each time. It can also be observed that the contextualization technique consistently results in better performance with respect to simple personalization, as can be seen in the average precision and recall depicted by Fig.5, which shows the average PR results over the different user cases

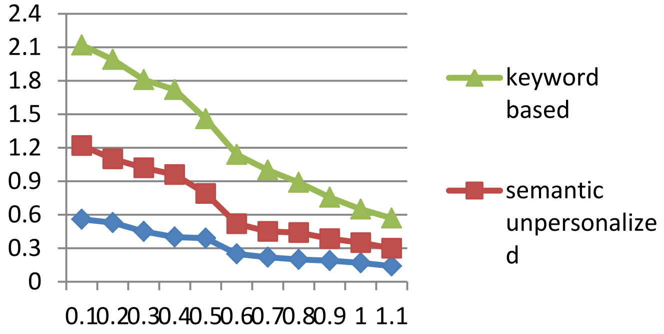

Fig. 4. Performance evaluation

The next experiment aims at determining the importance of personalization by using generated dynamic user model during using the system. The user model is used to re-rank the retrieved documents to match the user interest.

Personalization time:

Time to retrieve any information depends on the type of search engine, size of data set, relevancy between query and doc. user history & re-ranking algorithm used. The personalization performance can be expressed:

Personalization performance= $\sum_{i=1}^{n} Grank + UserRank$ and

For each page find UseRank=$\sum_{i=1}^{n} UR + VT + FC$ Where UR – user rating VT page view time and FC-frequency count and n represents threshold value

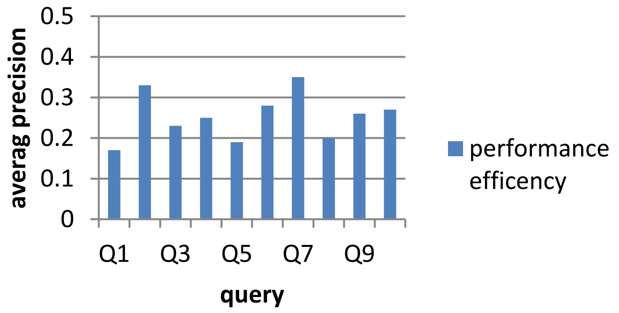

Fig. 5. Performance evaluation

This Fig.5 focuses the usage efficiency of the systems when the system uses to retrieve the result.

It is observed that 80% users, have found improved precision with the proposed approach in comparison to the standard search engine (Google) results, while 20% users have achieved equal precision with both the approaches. It has been observed that users who posed Queries in unpopular context than well liked context got better performance In addition, when the system can extract the exact context of user's need, the Precision and recall is found better than other search engine results.

The experimental result indicates that the efficiency of information retrieval, by the use of the above-mentioned personalized information retrieval system based on multi-agent, precedes evidently current information retrieval tools in common use to sum up the precision is improved 15% - 35%.

The system realizes individuation and intelligence of information retrieval for providing personal service to user via multi-agent collaboration according to user interest characteristics and different information needs. The construction algorithm and update algorithm of user interest model, which are based on user browsing history record and user browsing behavior, can discover user interest in time, control safely the scale of user interest model, and increase effectively document filtration efficiency.

V. Conclusion

In this study, a new information retrieval system based on Semantic Web and Multi-Agent has been presented to effectively the offset existing defects and constraints of the traditional keyword-based search, and help users to obtain required information.

The proposed system experimentation shows that, it can improve the accuracy and effectiveness for retrieving the web documents. It aims at providing the relevant web-document in certain domain that is matched to user's request. It can be used in other domain by editing the domain ontology using export option of APSIR and building the domain concepts weight table. A user model is proposed to improve the ranking of the relevant documents retrieved to user based on its interest.

Reference

[1] Ahu Sieg, Bamshad Mobasher, Robin Burke Ontological User's Profiles for Personalized Web Search In Proceedings of Conference of AAAI 2007

[2] Bettina Berendt, Andreas Hotho, and Gerd Stumme, Towards Semantic Web Mining ISWC 2002, LNCS 2342, pp. 264-278

[3] Filip Randlinski, Thorsten Joachims Query Chains : Learning from Implicit Feedback. In the Proceedings of KDD'05 August 21-24 2005

[4] Jingqiu XU, Zhengyu ZHU, Xiang REN, Yunyan TIAN, Ying LUO Personalized Web search using User Profile,Journal of Computational Information System 2007

[5] Kazuneri Sugiyama,Kenji Hatano,Masatoshi Yoshikawa Adaptive Web Search based on user Profile Constructed without Any Effort from Users ACM Press 2004

[6] Kazienko P, Kiewra M (2003) Link Recommendation Method Based on Web Content and Usage Mining. In: the Personalized Web Search: User Modeling Using Implicit Feed Back From Click Through Data

[7] Maciej Kiewra Iterative Discovering of User;s Preferences Using Web Mining , International journal of Computer Science and Applications Vol. II No. II pp. 57-66

[8] Mobasher B, Dai H, Luo T, Sun Y, Zhu J (2000) Integrating Web Usage and Content Mining for More Effective Personalization. LNCS 1875 Springer Verlag, 156-176

[9] Bernes-Lee T., Hemdler Agents and the Semantic Web IEEE Intelligent Systems Journal (march/April 2001)

[10] Paul Buhler and José M. Vidal (2005) Towards Adaptive Workflow Enactment Using Multiagent Systems ,Information Technology and Management Journal , volume 6 pages:61-87