Optimized Multi Agent Personalized Search Engine

DishaVerma
:disha.verma.in@gmail.com ,
VIPS, Guru Gobind Singh Indraprastha University
Research Scholar, Jagannath University,Jaipur

Dr. Barjesh Kochar
Professor,BPIT bjkochar@gmail.com

Prof. (Dr.) Y. S. Shishodia
PRO VICE CHANCELLOR, Faculty of Engg. & Tech. Jagan Nath University

Abstract
With the advent of personalized search engines, a myriad of approaches came into practice**.** With social media emergence the personalization was extended to different level. The main reason for this preference of personalized engine over traditional search was need of accurate and precise results. Due to paucity of time and patience users didn't want to surf several pages to find the result that suits them most. Personalized search engines could solve this problem effectively by understanding user through profiles and histories and thus diminishing uncertainty and ambiguity. But since several layers of personalization were added to basic search, the response time and resource requirement (for profile storage) increased manifold. So it's time to focus on optimizing the layered architectures of personalization. The paper presents a layout of the multi agent based personalized search engine that works on histories and profiles. Further to store the huge amount of data, distributed database is used at its core, so high availability, scaling, and geographic distribution are built in and easy to use. Initially results are retrieved using traditional search engine, after applying layer of personalization the results are provided to user. MongoDB is used to store profiles in flexible form thus improving the performance of the engine. Further Weighted Sum model is used to rank the pages in personalization layer.

Keywords:
Personalized Search Engine (PSE), Weighted Sum Model(WSM), Information Retrieval, User profiling, Web Personalization, MongoDB, Distributed Database

1. Introduction

Today, the World Wide Web provides us with a huge ever-growing source of information and has become a crucial part of our everyday lives. As an outcome of the speedy growth and dynamic content of the web, the traditional web search engines are becoming deficient.
Personalized search engines are replacing the traditional ones by catering the personalization on the basis of various parameters like user history, user profiles etc. With increased social media usage the users can be studied best using their social media accounts which further helps to produce fewer but personalized results. Thus next evident progression is integration of social media with search engines.

In this paper, we propose a multi layered search personalization approach. The architecture of the proposed model differentiates it from previous research [1] as it doesn't defy the vertical search engine instead it uses the results fetched by it and applies an additional layer of personalization. Personalization is done through several parameters like history, interest, profile, social media account etc. Another part of the model is levels of personalization. With personalization, privacy is affected and which is not appreciated by a lot of users so this engine given a power to user to decide the level of personalization. Also since layered architecture may increase response time thus backend used is unlike conventional databases. The concept of distributed database MongoDB is used as a core. The first part of the paper represents previous work in the field followed by theoretical explanation of the various layers used in the proposed architecture. Several parts of architecture like levels of personalization and distributed database technology Mongo DB is explained briefly. Following the conceptual description implementation is discussed comprehensively. The last section of paper focuses on Results and analysis. Future scope of the paper is also discussed in brief. on results concept introduced for the architecture (Evolution of Ontology in Multi Agent Systems). Problem definition section explains the current problem and proposed solutions for the area. Following the problem definition section is proposed architecture section which discusses the proposed architecture in detail. After which the analysis and evaluation section compares the traditional search engine with the proposed architecture. The last section explains the future scope and concludes the research.

Manuscript received September 5, 2024
Manuscript revised September 20, 2024
https://doi.org/10.22937/IJCSNS.2024.24.9.17

2. Literature

Owing to need for accurate results personalization came into existence. Personalization is not new, and the need for results according to the user preferences has led to many researches working on it. A lot of work has been done in last decade in field of personalization. Personalization has been proposed through various ways

A number of research groups have discovered and explored personalization and have broadly divided it into two categories: Explicit profiling and Implicit Personalization based Heuristics. Also Profiling could be done using several methods. Gauch et al. [1] explored user profiles from browsing history, Speretta and Gauch [2] created profiles using search history, and Chirita et al. [3] used profiles that users specified explicitly. Leung & Lee[4] on other hand proposed studying the logs creating profiles based on it. Captain Nemo project [19] implemented a functional search engine with personalized hierarchical search which extracts and displays search results according to retrieval models (personalization) and arrangement styles. In the WebNaut project [20] a multi-agent based search engine is proposed that consists of a set of interconnected agents and uses a meta-genetic algorithm for learning of the user's interests and personalizing search results. Contradicting experiment on small sample size showed the level of domain knowledge seems to have an effect on users search behavior, but not its effectiveness [22].

Another area explored was algorithm or model best suited for Ranking and personalization. Cho and Qiu[5] used Random Surfer Model for ranking pages. They also discussed extending the normal ranking model to Topic-Sensitive PageRank scheme (TSPR). This model was based on Topic Preference Vector. A Session-based personalized search algorithm was proposed by Daoud,Tamine & Boughanem which used correlation as background[6]. Shen,Tan and Zhai[7] proposed implicit profiling using decision-theory they also used TF-IDF weighting model for calculating information based on clickthroughs. Author of Excalibur project proposed a personalized search engine that extracts users preference implicitly and re rank them by using the Naive Bayesian classifier and the resemblance measure[16].

Besides the several algorithms and models, Ontology based personalization also helped to produce contextual results and improving strategic adaptation based on the knowledge obtainable from users' actions [8, 9, 21, 26]. Understanding the Perspective or category can only be done if related words are and subcategories are already explored for the domain thus Open Directory Project is used in several ontology based personalization [9,14,15]. Radovanovic and Ivanovic [17] proposed a meta-search engine, called CatS that utilizes text classification techniques to improve the view of search results and displays a tree of topics derived from the dmoz which is an Open Directory topic hierarchy that can be traversed by user. Another dimension explored in the area is multi source personalization. A multisource profiling and multi-application personalization approach that leverages diverse usage data collected from multiple service domains such as mobile and web[10].

Some personalization focuses of exploring the new area called semantic web instead of WWW. Under the assumption of a shared model of semantic concepts, one can represent the content metadata (categories) as well as the semantics of consumption acts (purchase events, viewing or browsing sessions) with the same terms in the user profile (interests) [11]. According to the overlay approach, the profile created is composed of a set of {concept, value} where in concept refers to user's interest and value refers to degree of interests[12]. Understanding user and creating profiles not helps search engine to cater precise results but it also opens a lot of revenue streams for companies based on information model[13].

Some authors also created a relation between long long-term search activity history activity and short-term search session behavior[14]. The short-term context, which is the is in regards with information that emerges from the current user's information need in a single session. The other context is long-term which refers to the user interests that have been inferred from profile explicitly created by him or his past sessions[8]. Kanteev, Minakov, Rzevski, Skobelev, and Volman [18] proposed a multi-agent content understanding system which is based on the semantics of pages. It generates the semantic descriptors as well as uses the knowledge about problem in the specific domain that is stored in the form of ontology.

Very few researches focused of storage and optimization. When personalization or ranking is implemented, it may result in increase of response time which is imperative to be dealt with. [23] Proposed the double-byte inverted index and virtual memory drives technology to ensure the system response time is not hampered [24]. Another concern is need for a database that is able to store huge amount of data produced by profiling. Normal relational database is not able to handle the concurrent queries and profiling a This leads to research in database technology one of which is NoSQL database. Compared to relational database, MongoDB (technology of NoSQL) supports schema-free storage(unlike relational), has great query performance with huge amount of data and provides easy horizontal scalability. It is more fitting for data storage in personalized search engines[24]. Cassandra another example of NoSQL database which is now deployed as the backend storage system for multiple services within Facebook[25].[28] proposed an algorithm OPIC to decrease CPU utilization while calculating page visits. It was essential to study all the above concepts for our study since our architecture has various layers which directly or indirectly implement the discussed concepts.

3. Background

This section of paper lays down the theoretical background of the architecture implemented.

3.1 Concept of Personalization

A traditional search engine returns the same results for the same query irrespective of user interest and choice. Unlike traditional system, personalized search engine cater to user needs.

Personalized search engine intends to customize search based on an individual user's interest, needs, requirements or his search history/ pattern having an effect on the user's relevance assessment. Thus basic categorization of personalization could be done on the basis of degree of personalization required.

The two major categories on which personalization is based are Explicit & Implicit profiling. Explicit profiling refers to creation of a profile by taking interest as input from users.

Explicit profiling is performed on server side where in every user interests are stored as separate entity. Implicit profiling is client side personalization which studies pattern of user search like his history, duration he spends on a page etc. After storing this data in cache and cookies an implicit profile can be created and thus results can be customized for specific user.

3.2 Multi criteria decision making (MCDM)

Multi criteria decision making (MCDM) refers to traversing, prioritizing, selecting the alternatives from among a finite set of substitutes or options in terms of the multiple criteria. Weights play a significant role in MCDM models which provide the degree of importance of criteria under consideration. Several different methods have been developed to compare these criteria's in account. Mostly weights are inferred through judgments and adhoc approaches which make them vague and inaccurate in nature. Thus weights cannot be exactly evaluated or calculated with numerical values, leading "true" weights almost nonexistent. Even if the precise weights are possible, it is very difficult and time consuming making it impractical for use [30]. Here the rank ordering weighting methods plays an important role in providing an approximation of "true" weights when rank ordering information is known.

In our research, we consider a problem with m page as alternatives $p_1, p_2, p_3, \ldots \ldots . p_m,$ and personalization criteria $C_1, C_2, p_3 \ldots \ldots . C_n.$.

For criteria, we have $w = [w_1, w_2, \ldots, w_n]$

Such that $w_1 + w_2 + \cdots + w_n = 1$

where w_j represents the weight of criterion $C_j, w_j \geq 0 (j = 1,2,3, \ldots n)$

Let x_{ij} denotes the performance value of each page alternative p_i in terms of criteria C_j.

Where $(i = 1,2,3, \ldots m; j = 1,2,3, \ldots n)$ The decision matrix

$$D = (x_{ij})_{m \times n}$$

represents the evaluation score x_{ij} of each page p_i with respect of each criterion C_j.

All criterions are then normalized using following formula

$$C_{ij} = \frac{x_{ij} - \min_i x_{ij}}{\max_i x_{ij} - \min_i x_{ij}}$$

Where x_{ij} is the score of i[th] page with respect to j[th] criterion before normalization.

After applying normalization higher C_{ij} is more preferred over lower C_{ij}

Let the normalized decision matrix be

$$C = (c_{ij})_{m \times n}$$

The method used in our research for re-ranking Simple Additive Weighting (SAW) which is also known as weighted linear combination or scoring method.

The three major steps in simple additive weighting is

1. Scaling of scores so that they are comparable
2. Applying criteria weights
3. Sum the values along rows and rank the pages according to the final score of each page.

Final score of each alternative is calculated as:

$$S_i = \sum_{j=1}^{n} c_{ij} w_j$$

Where S_i is score for i^{th} page, and c_{ij} is the normalized score of i^{th} page with respect to j^{th} criterion and w_j is the weight of criteria j.

The final scores are used to re-rank the pages. Higher the value of S_i for a page higher its rank (1 being the highest rank).

For calculating weights it was imperative to understanding importance of each criterion. So considering if we have *n* prioritized criteria (priority taken from user), each criteria has a has a rank. This rank is inversely propotional to the weight (r=1 denoting highest weight). After this understanding various methods were compared and ranks were converted to numerical weights. Further The rank sum method was used to calculate the weights of the criterions. The weight can be calculated as

$$n\text{-}r_j+1$$

where n is total number of criterions and r_j is the straight ranks assigned on the basis of importance.

Table 1 Weights and Normalized weights of criterions

Criterion	Straight Rank	Weight	Normalized Weight
A	4	2	.133
B	2	4	.267
C	5	1	.067
D	1	5	.333
E	3	3	.200
		15	1.00

The Weights can also be represented using a matrix as represented in table

Table 2 Pages scores corresponding to various criterions

	A	B	C	D	E
Page	.133	.267	.067	.333	.200
P1	5	4	2	4	3
P2	6	3	1	3	1
P3	5	3	3	3	1
P4	4	3	4	3	2

The fig 1 represents the normal weight and normalized weights corresponding to the various criterions. After deciding weight for the entire criterions fig 2 represents the final score using weighted sum model discussed previously

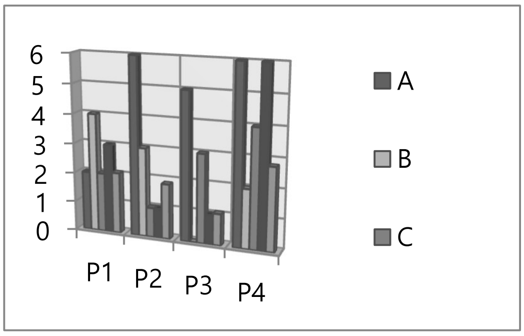


Fig. 2 Final scores coresponding to Pages

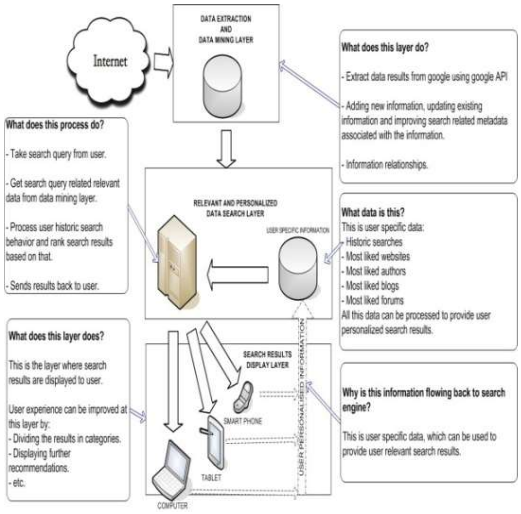


Fig. 3 High level view of proposed architecture [27,29]

4. Proposed Work

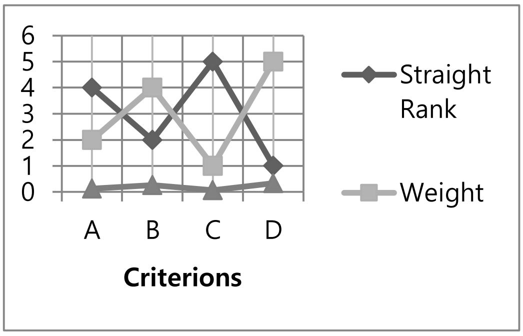


Fig. 1 Normal weights and Normalized Calculated Weight

The proposed work deals with search engine that incorporates the feature of fetching results from a search engine, and further personalizing it using implicit and explicit profiling The following modules have been used in this personalized search engine:

4.1 Search engine module

This layers/module will fetch the results from search engine in our paper traditional search engine is used as search engine. API's were used to fetch the contents from search engine and social media. The focus of this layer is to fetch the results from search engine in a format where in we can filter and perform other operations on it.

A company building search engine takes years of research to reach to a working model so our purpose is not to defy their working instead we propose to improvise on the results provide by them.

4.2 Personalization Module

This module takes input from the previous layer and personalizes it through various parameters like history etc. This layer is based on two types of personalization. The first one being Profile based, in which user is required to input his/her interests explicitly also prioritize his result on the electronic form provided to him. This form further is used to personalize and re-rank the results. Priority here works as weights in algorithm. The second type of personalization is heuristic based. User's browsing history plays an important role in depicting his interests. Thus the architecture proposed tracks user browsing history and re-rank the page accordingly. Another type of personalization implemented in our architecture is social media based. Our architecture fetches the user interests from his/her Facebook page and re-rank his results accordingly. The prerequisite of this type of personalization is Facebook Log-in. User will be required to log-in through his Facebook id and password.
This type of personalization lays down a prototype that personalization can be done using any kind of social media profile. This is one of the features that differentiate this architecture from the previous ones.

4.3 Storage and Processing Module

As a backend to the second layer this layer is responsible in producing re-ranked results in optimized time. The process of fetching the result from search engine and re-ranking may increase response time. Thus this layer introduces the concept of distributed databases.
Relational databases have been underpinning search applications from long. But researchers now are increasingly considering alternatives to traditional relational infrastructure. There exist several motivations behind it. In few cases the motivation is technical that is necessity to handle new, multi-structured data types or scale beyond the current capacity contingencies of legacy systems. Another important motivation being agility or speed required to deal with mammoth amount of data generated by use of social media.

4.4 Presentation Module

The final module focuses on presenting the re-ranked result compatible to all the devices like desktops and hand held.

5. Dataset Characteristics and Experiment

In this section we will discuss the working prototype build for experiment and result analysis

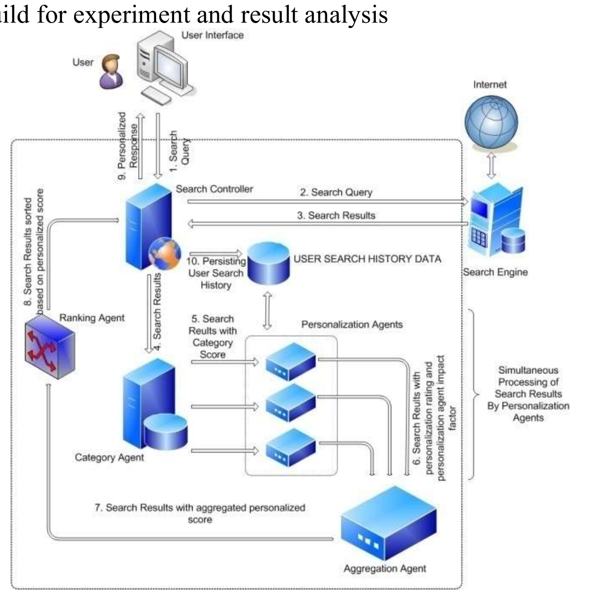


Fig.4 Reranking Steps in Proposed Architecture[29]

5.1 Working model

The detailed architecture of the model is represented in Fig 4. The following steps are being performed to find refined results:

a) Search a query on the GUI provided
b) First layer will fetch results from traditional search engine corresponding to the query.
c) Category agent categorizes the result into various categories and provides suitable score.
d) Various weights and scores are assigned to all the parameters like history, interest social media etc using algorithm discussed above.
e) The results are re-ranked using distributed databases.
f) The results are displayed on user's screen.

5.2 Data Maintained by engine

Our engine is storing large amount of data few of the crucial tables are

a) User interest table which is created on the basis of profile build on user interest.
b) Hits table keep track of the user history and log

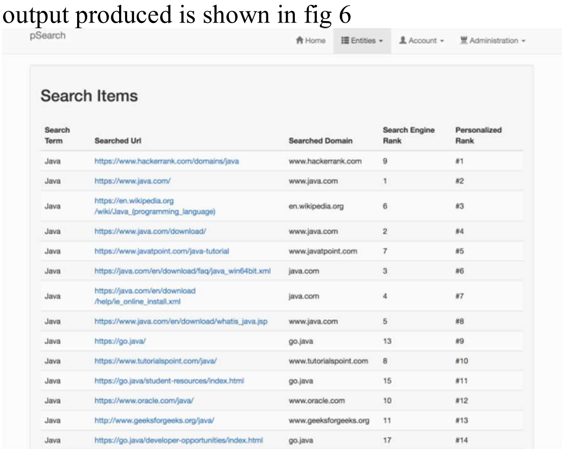
c) Social Media Data table is responsible for fetching data from user's social media profile and storing it in semi structured form

5.3 Re-ranked results

The last step of the engine is to re-rank the result on the basis of proposed methodology. We performed the test by fetching first ten results and then re-ranking them accordingly.

Fig. 5 Home Page of the engine

The engine's home page is represented using fig 5. The output produced is shown in fig 6



Search Term	Searched Url	Searched Domain	Search Engine Rank	Personalized Rank
Java	https://www.hackerrank.com/domains/java	www.hackerrank.com	9	#1
Java	https://www.java.com/	www.java.com	1	#2
Java	https://en.wikipedia.org /wiki/Java_(programming_language)	en.wikipedia.org	6	#3
Java	https://www.java.com/download/	www.java.com	2	#4
Java	https://www.javatpoint.com/java-tutorial	www.javatpoint.com	7	#5
Java	https://java.com/en/download/faq/java_win64bit.xml	java.com	3	#6
Java	https://java.com/en/download /help/ie_online_install.xml	java.com	4	#7
Java	https://www.java.com/en/download/whatis_java.jsp	www.java.com	5	#8
Java	https://go.java/	go.java	13	#9
Java	https://www.tutorialspoint.com/java/	www.tutorialspoint.com	8	#10
Java	https://go.java/student-resources/index.html	go.java	15	#11
Java	https://www.oracle.com/java/	www.oracle.com	10	#12
Java	http://www.geeksforgeeks.org/java/	www.geeksforgeeks.org	11	#13
Java	https://go.java/developer-opportunities/index.html	go.java	17	#14

Fig. 6 Re-ranked vs old Rank results

We have studied how profile based and history based personalization interact, and how each of them may be used in combination with each other in isolation to optimally contribute to gains in relevance through search personalization. Through an experiment we have shown how traditional search engine result can further be refined on the basis of user interests and history. Importantly, we also showed that how by changing backend technology the response time. Also since we have written generic code this engine can be customized to any vertical search engine provided an API is available for it. Also more them one social media personalization can be performed with few changes. The mongo DB used as background help to attain horizontal scalability.

6. Conclusion

The large amount of data proposed by traditional search engines is no more able to satisfy the user's requirement. Researchers then progressed to vertical search engines and personalized search engine. Several researchers also advocated in favor of meta search engine over tradition ones. Personalization could be done in different ways and different levels. The aim is to attain maximum precision in understanding user's interests and preferences.

The search engine proposed and implemented in our paper focuses on working on layered architecture. It enables user to take advantage of traditional search engine as well as provide user the power to control the personalization on basis of his interest, history, social media account. The algorithm used is weighted sum with prioritized weight calculation. The backend technology used is distributed database MongoDB which helps in providing the re-ranked results in timely manner. The architecture proposed is different from various proposed methodologies on several levels.

a) The traditional search engine as a layer,
b) The social media integration.
c) Introduction of priority weights.
d) Optimization through MongoDB.

The focus of our work is providing refined fewer results in an efficient optimized manner. In our future studies we will apply this architecture with increased data set before making it live for users to work on.

References

[1]Gauch, S. J Chaffee, A ontology based personalized search and browsing. Web intelligence and agent systems, 1 (3), 219-234 (2003).

[2]Speretta M, Gauch S. Personalized search based on user search histories. In Web Intelligence, 2005. Proceedings. The 2005 IEEE/WIC/ACM International Conference on 2005 Sep 19 (pp. 622-628). IEEE.

[3]Chirita, P. A, Nejidi, Using odp metadata to personalize search.In proceedings of 28th annual international ACM SIGIR conference on research and development in information retrieval, ACM August 2005.

[4] Leung K. W. T. ,Lee D. L., Deriving Concept-Based User Profiles from Search Engine Logs, in IEEE Transactions on Knowledge and Data Engineering, vol. 22, no. 7, pp. 969-982, July 2010.

[5] Qiu, F., & Cho, J., Automatic identification of user interest for personalized search. In Proceedings of the 15th international

conference on World Wide Web (pp. 727-736). ACM. (2006, May).

[6] Daoud M, Tamine-Lechani L and Boughanem M., Learning user interests for a session-based personalized search. In Proceedings of the second international symposium on Information interaction in context (IIiX '08), ACM, 57-64,(2008) DOI=http://dx.doi.org/10.1145/1414694.1414708

[7] Shen, X., Tan, B., & Zhai, C., Implicit user modeling for personalized search. In Proceedings of the 14th ACM international conference on Information and knowledge management (pp. 824-831). ACM. (2005, October).

[8]Tamine-Lechani, L., Boughanem, M., & Zemirli, N. Personalized document ranking: Exploiting evidence from multiple user interests for profiling and retrieval. *JDIM*, *6*(5), (2008), 354-365.

[9]Xiang B, Jiang D , Pei J , Sun X ,Chen E , Li H, Context-aware ranking in web search, Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval, July 19-23, 2010, Geneva, Switzerland [doi>10.1145/1835449.1835525]

[10] A. Aghasaryan *et al*., Personalized application enablement by web session analysis and multisource user profiling, in *Bell Labs Technical Journal*, vol. 15, no. 1, pp. 67-76, June 2010.

[11]Shah, A.; Jain, S.; Chheda, R.; Mashru, A., Model for Re-ranking Agent on Hybrid Search Engine for E-learning,"Technology for Education (T4E), 2012 IEEE Fourth International Conference on , vol., no., pp.247,248, 18-20 July (2012)

[12] P. Brusilovsky and E. Millan, User Models for Adaptive Hypermedia and Adaptive Educational Systems, The Adaptive Web: Methods and Strategies of Web Personalization (P. Brusilovsky, A. Kobsa, and W. Nejdl, eds.), Springer, Berlin, Heidelberg, New York, 2007, pp. 136–154.

[13] M. Beccue and D. Shey, Service Personalization: Subscriber Data Management, Subscriber Profiling, Policy Control, Real-Time Charging, Location, and Presence, ABI Research, 2Q 2009.

[14]Bennett, P. N., White, R. W., Chu, W., Dumais, S. T., Bailey, P., Borisyuk, F., & Cui, X. Modeling the impact of short-and long-term behavior on search personalization. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval* (pp. 185-194). ACM. (2012, August).

[15] Middleton, S., D.D. Roure and N. Shadbolt; Capturing Knowledge of User Preferences: Ontologies in Recommender Systems, in 1st Int. Conference on Knowledge Capture, ACM Press, pp. 100-107, 2001

[16] L. Yuen, M. Chang, Y.K. Lai, and C.K. Poon, Excalibur: a personalized meta-search engine, Proc. 28th Annual International Computer Software and Applications Conference (COMPSAC'04), 2004, vol. 2, pp.49-50.

[17] M. Radovanovic, and M. Ivanovic, CatS: A classification-powered meta-search engine, Advances in Web Intelligence and Data Mining, Springer-Verlag, vol. 23, 2006, pp. 191–200.

[18] M. Kanteev, I. Minakov, G. Rzevski, P. Skobelev, and S. Volman, Multi-agent meta-search sngine based on domain ontology, Springer, vol. 4476, July 2007, pp. 269-274, doi: 10.1007/978-3-540-72839-9_22.

[19] S. Souldatos, T. Dalamagas, and T. Sellis, Captain Nemo: A metasearch engine with personalized hierarchical search space, INFORMATICA -LJUBLJANA-, vol. 30, 2006, pp. 173-182.

[20] N. Zacharis, T. Panayiotopoulos, SpiderServer: the meta-search engine of WebNaut, Proc. the 2nd Hellenic Conf. on Artificial Intelligence, 2002, pp. 475-86.

[21]Prates, J.C.; Siqueira, S.S.M., Using Educational Resources to Improve the Efficiency of Web Searches for Additional Learning Material, Advanced Learning Technologies (ICALT), 2011 11th IEEE International Conference on , vol., no., pp.563,567, 6-8 July (2011)

[22]Zhang, X., Anghelescu, H.G.B., and Yuan, X. Domain knowledge, search behavior, and search effectiveness of engineering and science students. Inf. Res., 10(2): 217. (2005).

[23]Suping X, Mu L, Ziqi Z, Research of index technology for topic search engine, IT in Medicine and Education (ITME), 2011 International Symposium on , vol.1, no., pp.93,95, 9-11 Dec. (2011)

[24]Gu, Y., Shen, S., & Zheng, G. Application of nosql database in web crawling. *International Journal of Digital Content Technology and its Applications*, *5*(6), 261-266, (2011).

[25] Lakshman A , Malik P, Cassandra: a decentralized structured storage system, ACM SIGOPS Operating Systems Review, v.44 n.2, April 2010 [doi>10.1145/1773912.1773922]

[26] Joung, Y., M. E. Zarki and R. Jain; A User Model for Personalization Services, in 4th Int. Conference on Digital Information Management, ICDIM 2009, pp. 247-252, 2009.

[27]Verma D, Kochar B Multi Agent Architecture for Search Engine (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 7, No. 3, 224-220 (2016)

[28] Serge Abiteboul, Mihai Preda, and Gregory Cobena. Adaptive on-line page importance computation. In *Proceedings of the 12th international conference on World Wide Web* (WWW '03). ACM, 280-290.2003 [DOI=http://dx.doi.org/10.1145/775152.775192]

[29] Verma D, Minocha K, Kochar B, A Multi-Agent Based Personalized Search Engine with Topical Crawling Capabilities, The IUP Journal of Computer Sciences, Vol-8, No. 3, 20-33 (2014)

[30]Roszkowska, E. Rank ordering criteria weighting methods—A comparative overview. *Optimum. Stud. Ekono. 5*, 65 (2013).