Multi Agent Architecture for Search Engine

Disha Verma
Research Scholar,
Jagannath University,
Jaipur, India

Dr. Barjesh Kochar
Professor (IT),
Jagan Institutre of Management Studies
Delhi, India

***Abstract*—The process of retrieving information is becoming ambiguous day by day due to huge collection of documents present on web. A single keyword produces millions of results related to given query but these results are not up to user expectations. The search results produced from traditional text search engines may be relevant or irrelevant. The underlying reason is Web documents are HTML documents that do not contain semantic descriptors and annotations.**

The paper proposes multi agent architecture to produce fewer but personalized results. The purpose of the research is to provide platform for domain specific personalized search. Personalized search allows delivering web pages in accordance with user's interest and domain. The proposed architecture uses client side as well server side personalization to provide user with personalized fever but more accurate results. Multi agent search engine architecture uses the concept of semantic descriptors for acquiring knowledge about given domain and leading to personalized search results. Semantic descriptors are represented as network graph that holds relationship between given problem in form of hierarchy. This hierarchical classification is termed as Taxonomy.

Keywords—Search engine; Data mining; Multi agent systems (MAS); Semantic mapping; Hozo

I. INTRODUCTION

Internet is one of the biggest repositories of information of web documents or HTML documents. Various search engines were created to access these web pages on basis of keywords entered by user. The main drawback of these traditional search engines is that they perform searching by focusing on keywords rather than on meaning of content. It is the reason that they lack semantic dependency among user entered keywords.

The users are getting addicted to latest technology trends and they want their services at their doorstep. Due to diverse search results, the accuracy of retrieved information is decreasing. Different users have different interests but owing to algorithm without interest as a parameter search engines produce same result for all users that does not satisfy individual user needs. This led to development of personalized and semantic search engines. With the advent of Vertical search engines also called Vortals, the personalized search engines were the next palpable step. Vertical search engines (VSE's) emphasize on single topic or specific domain while personalized search engines extracts user preferences and provide them tailored results.

The search engine proposed is an intelligent domain specific personalized search engine which personalizes the results pertaining to a specific domain. The concept which differentiates this research from previous works is personalization at both ends (client and server) and domain specific searching. Unlike previous engines the proposed architecture keeps the sophisticated architecture of Google and then adds layers of personalization to it. In the process of adding layer it doesn't let to significant increase of processing time. Thus making the engine more personalized results in reasonable time.

The first part of the paper represents previous work in the field followed by theoretical explanation of the 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.

II. LITERATURE

Various studies have been conducted and results are deduced regarding personalized view of search engines as follows:

Gauch, & Chaffee [1] captured user interest by browsing their history stored in form of cookies. Speretta & Gauch [2] developed personalization by identifying user profiles. The personalization model used methods to grab user's search history in order to retrieve results precisely. Chirata & Nejidi [3] found that queries are handled in three categories viz. clear queries, semi-ambiguous and ambiguous queries. There is no need of personalized search in handling non ambiguous queries but for handling semi ambiguous and ambiguous queries, personalization is necessary. According to [19], the user based personalization is the most significant contextual factor, which can alleviate an ambiguous web search in an ad hoc retrieval task. Some works model the user interests as being a rich repository of personal information extracted from the user search history like past click through data [20], browsing features or desktop information [21], etc. The use of multi agent technology is widespread among large number of applications like distributed systems, data mining, logistic management and sensors [4]. The technology is based on working of nature of agents to be employed for retrieval of information. Each agent has its independent tasks and they work in collaboration with other agents in system. Another variant of multi agent systems is non-equilibrium systems that have been studied by Nicolis & Prigogine [5]. Non-equilibrium system means previous keywords and phrases related to query can be modified during next phase of searching. [14] investigated, compared various Text categorization algorithms and proposed profile based personalization using Open directory project. [15] proposed the search engine which took in account positive as well as negative preferences of the user. [16] proposed click content and location entropies to measure interest in the content and/or location information in the results. [18] proposed a special log system to capture users' interactions in web systems. The system consisted of two subsystems: log capturing and log analyzer system. The log system makes use of several attributes identifying user session, such as embedded session identifiers, cookies and IP-address of the accessing host. All the above researches where based on HTML pages. Finding semantic categorization was difficult. [17] proposed an architecture based on creating profiles using semantic descriptors. This paper proposes a similar architecture.

III. Evolution of Ontology in Multi Agent Systems

With advent of concept of Ontology and Semantic Web, Knowledge Management (KM) solutions have started employing in different environments where agents can define their own ontology. Ontology allows an agent to perform following tasks:

- Agents can distribute collection of data and enable communication of data in different environments and applications.
- Agents increases information retrieval performance by searching results in less time.
- Agents use learning algorithms to give examples regarding retrieved results on basis of query entered by user.
- Interface Description Languages and services are provided for different environments where Interface Language refers to defining of data objects and their location.
- Agents are able to access some distributed models to enable interaction between processes like CORBA (Common Object Request Broker Architecture), RMI (Remote Method Invocation).

A. Multi Agent Systems (MAS)

1) Problem: - While using KM solutions for integration of knowledge processes, there used to work on Peer to Peer (P2P) architecture for enabling distributed control of knowledge resources at centralized location via peers. Peers are directly connected to Hub. But this architecture is not suitable for accessing knowledge from complex systems.

2) Solution: - Multi Agent Systems (MAS)

3) Analysis: - Systems where individual self-authorized agents derive new facts with the help of other agents are called MULTI AGENT SYSTEMS.

In these systems, individual agents create their different models and prototypes instead of following standard ontology. *An agent is defined as an entity with/without body*. Agents use different kinds of knowledge sources and resolve differences among themselves to provide best answers of queries in complex KM environment.

Ontology is evolved by learning concepts and relationships by taking guidance from other agents. Agents are trained explicitly as well as they learn in the course of their use. With time agents improve their knowledge also communication among various agents adds to their learning. Fig.1. represents the characteristics which are imperative for software agents.

4) Approach involved: - There are following assumptions to be kept in mind which are as follows:

- Assume given organization has n agents *Ag1…..Agn* where each agent manages knowledge for its organization.
- Each agent knows concept which is denoted by *Ck.*
- Each agent has positive and negative thoughts (*ti*) with respect to that concept.
- Each agent has learning algorithm (*Li*) or classification mechanism.
- Each agent has its unique Ontology denoted as *Oi.*
- Each agent has set of features for representing concepts. It is denoted as *fi.*

System is represented as: Agn = {Li, ti, fi, Oi}

a) Learning Algorithm (Li): - An agent learns a concept under supervision of teachers. It is also possible that teachers are not well expert about given queries, in that case they can use learning algorithm to give example regarding queries.

b) Set of thoughts/examples (ti): - In this, positive and negative examples are taught to agents by teacher. These examples are related to given problem domain. Using this classification capability, learner agent is able to decide whether example is positive or negative.

c) Set of features (fi): - They are most important factor to represent concepts. Features are selectively used by each agent to represent different concepts. In Multi Agent concept learning, we have to collect most related features from different sources of knowledge in order to develop a comprehensive concept.

d) Ontology (Oi): - In terms of concepts and knowledge sharing, it is defined as mixture of Meta concepts and fine grained structure. Meta concepts are concepts which are divided into sub concepts until fine grained concept level is reached. It is preferred to use own ontology rather than using standard ontology.

Ontology- based semantic integration [6,7, 8] can be achieved to resolve differences that arises during run time interaction of system and agents. It is implemented in following ways:

- Using a single centralized ontology for all agents and application domain.

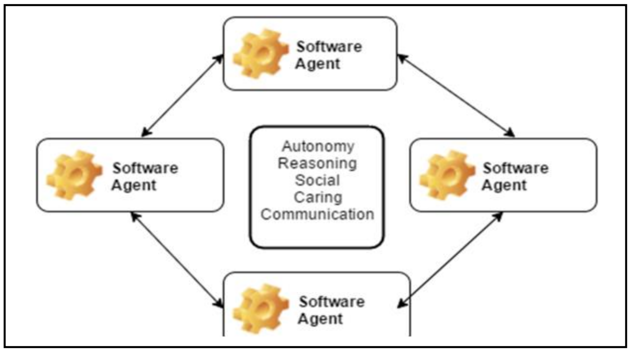

Fig. 1. Characteristics of the agents

- Merge source ontology into a common ontology to prevent overlapping of concepts by various agents.
- Search set of mappings or matches between two ontology when it is difficult to merge them due to the difference.

B. Semantic Mapping on Ontology

1) Importance: - Multi agent systems are used to convert input query into semantic annotations and computes their rank by semantic mapping. Then, there is need of mapping in ontology because Ontology gives complete description of a given problem that can be communicated among people and application systems. Each concept has its own attributes and relationships.

2) Semantic Mapping: - It is defined as process of finding solution to given problem with the help of relative concepts used in other domain. It is needed in order to realize full growth of Semantic Web and processes information between ontology. Consider a scenario: - There are two beers belonging to different countries. One is Australia and other is Denmark.

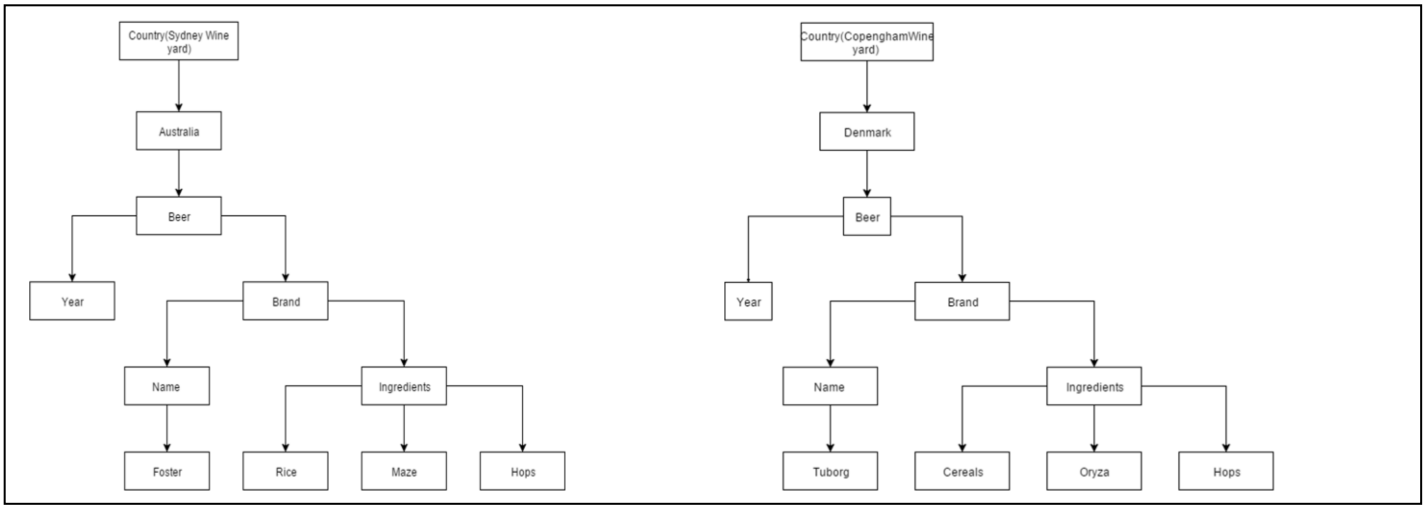

Fig. 2. Semantic Mapping[9]

Let two countries develop their web pages and decided to enable their web content. They make use of ontology in making web pages. Here is mapping between the ingredients of two beers belonging to different countries. Both beers contain different ingredients but some are common to both due to which we can find suitable beer and give validation to concept of Semantic Mapping. Figure 2 represents semantic mapping of sample beer. We have seen that Rice and Oryza are treated as equivalent concepts in different countries. Oryza is biological name of rice. Such a correspondence is called Semantic Mapping. Thus because the context was same the semantic matching could be achieved. The context was interpreted using ontology.

IV. Problem Definition

Most of search engines are based on client server approach. The personalization is implemented either at client side using histories, location or click model [16]. This approach suffers from the fact if the machine is used by multiple users or user chooses to delete the history. The other approach focuses on creating profiles explicitly for each individual user and storing these profiles on server. This approach requires a sophisticated system of storing and fetching profile information in small amount of time. Also this approach raises a concern if user is not interested in explicit profile. There is a need to create an optimum level of dependency on both client and server side to achieve favorable personalization.

The next area of concern is efficiency. Since our architecture works on fetching results from basic search engine like Google and then applying personalization the added layer will increase the response time. The higher the response time lower the efficiency. So it is essential to introduce a technology which reduces this time to achieve personalized yet quick results.

A. Plan of Solution

- Introducing Levels of Personalization: The solution of first problem is achieved in our paper by introducing four levels of personalization:

Level 3: Personalization based on both explicit profile and implicit profile.

Level 2: Personalization based on only explicit profile (history disabled)

Level 1: Personalization based on only Implicit profile (based on history)

Level 0: No personalization.

Level of personalization decreases from Level 3 to 0. Depending upon user requirement the personalization can be achieved using both(client side personalization & server side personalization), either or none.

- The technology proposed for second problem is in-memory data grid or distributed cache. An in-memory data grid (IMDG) is a data structure that resides entirely in RAM (random access memory), obviating the need for electromechanical mass storage media. The memory is divided into redundant nodes. The data is processed using these nodes. Initially complex problem is divided into several small subparts and then executed simultaneously. This not only decreases the processing time significantly but also makes it practical to store terabytes of data completely in RAM.
- Another significant role in the architecture is played by Agents. The use of multiple agents refines the results which are then passed to semantic descriptor module that will parse entire results and convert them into semantic blocks of data so that they can be mapped to given ontology. Mapping to ontology requires ontology development phases like specification phase, design and formalization phase. All these techniques are incorporated in proposed multi agent search engine architecture.

V. PROPOSED APPROACH

The proposed semantic multi agent search engine processes query retrieved from traditional search engines and analyzes these results according to user's priority with the help of multiple agents. This search engine is beneficial because it has ontology domain module which is used to represent relationship between user's preferences and produced search results.

A. *Modules Interaction*

The search engine has three modules namely Agent module (consisting of various agents), semantic descriptor module and ontology development module.

a) Agent Module: - It involves use of various agents that interacts with user and produces refined results. Agents are interface agent, facilitator agent, resource agent, mining agent and many more. Their functions are listed in table 1.

Pre decision made in order to mine given data sources is called Data Mining Task Planning. Data mining task planning requires compensation between Facilitator Agents and Mining Agents through message passing. Figure 3 represents this interaction

TABLE I. SOFTWARE AGENTS ANF THEIR ROLES

AGENTS	ROLE
(a) Interface Agent (User Agent)	It interacts with user to provide requirements and displays results. It has interface module that contains method for inter agent communication.
(b)Facilitator Agent (Management Agent)	It activates different agents. It receives questions from interface agent and may take the help of group of agents to solve those questions.
(c) Resource Agent (Data Agent)	It maintains Meta data information about data sources. It generates queries based on user request and sends their results to user agent.
(d) Mining Agent	It implements Data mining techniques and algorithms.
(e) Result Agent	It observes mining agents and other results from them. After obtaining results, these agents show results to user agent by integrating with manager agent.
(f) Broker Agent	It is advisor agent that can send reply to query of an agent with name and ontology of respective agent.
(g)Ontology Agent	It maintains and provides knowledge of ontology to solve queries related to ontology

Consider User Agent is denoted U. Facilitator Agent is denoted by X. Broker Agent is denoted by Y. Mining Agent is denoted by Z. If U sends request to X to ask for Data mining with other agents in system. Then X tries to compensate with Y to determine which agents are suitable for performing task. Mining Agent Z is responsible for completion of task whileX is used for planning. When Z is completed, it shows results to X and X passes them to U.

b) Semantic Descriptor Building Block:- It automatically converts text or sentences into meaningful blocks of data and put them in semantic descriptor which is nothing but graph showing concepts and values. The concepts are derived so that they can be designed to ontology in hierarchical manner.

c) Ontology Development Module

The term Ontology [10] can be defined in different ways. Ontology is abbreviated as FESC (Formal, Explicit, and Specification of Shared Conceptualization) which is defined as:

- Formal: It specifies that it should be machine understandable.
- Explicit: It defines type of constraints used in model.
- Shared: It means that ontology is shared by group. It is not restricted to individuals.
- Conceptualization: It refers model of some phenomenon to identify relevant concepts of that phenomenon.

Ontology is also defined as set of concepts and relationships arranged in hierarchical fashion. Ontology development [11] needs well defined methodology that must follow certain guidelines:

- Ontology being developed should follow Software Engineering standards.

- Ontology development strategy should be simple and practical.

B. Phases Involved in Developing Ontology

The phases that are being used in developing ontology also satisfy Software Engineering principles and thus called as Software Development Life Cycle (SDLC) phases. They are described below:

a) Specification Phase: - This phase has its few activities.

- Domain Vocabulary definition: - It defines common name and attributes for domain concepts.
- Identifying Resources: - A Resource is anything that has URI. So, if some concepts have number of instances, then they can be grouped into a class.
- Identifying Axioms: - They are structures that represent behavior of concepts.
- Identifying relationships: - Relations are defined within resources.
- Identifying data characteristics: - Defines features of types of resources and their relationship.
- Applying constraints: - Constraints represent named relationships between domain and range class.
- Verification: - After designing preliminary web ontology model, it is necessary that it should be tested for its correctness.

b) Design Phase: - The phase is backbone of Semantic Web. The physical structure of designed ontology is based on RDF model which is associated with three triples- Subject, Predicate and Object.

- Predicate: - All characteristics of resources and relationship are taken as Predicate. E.g. each train is assigned unique PNRNo called as ‘HasPnrNo’.
- Subject: - All domain classes of characteristics and relationships of resources are taken as Subject.E.g. there are various passengers travelling to city each having unique URI, so they are grouped in ‘CityPassengersGroup’.
- Objects: - Refers to Range class relationships. E.g. HasPnrNo contains range class ‘NUMBER’ which is literal.

c) Formalization Phase: - This phase is result of output of ontology obtained in design phase with the help of some tools.

VI. Evaluation and Analysis

Firstly, the user entered query related to computer components or appreciations. The search results are produced through GOOGLE search engine. It produces millions of results that are relevant or non relevant.

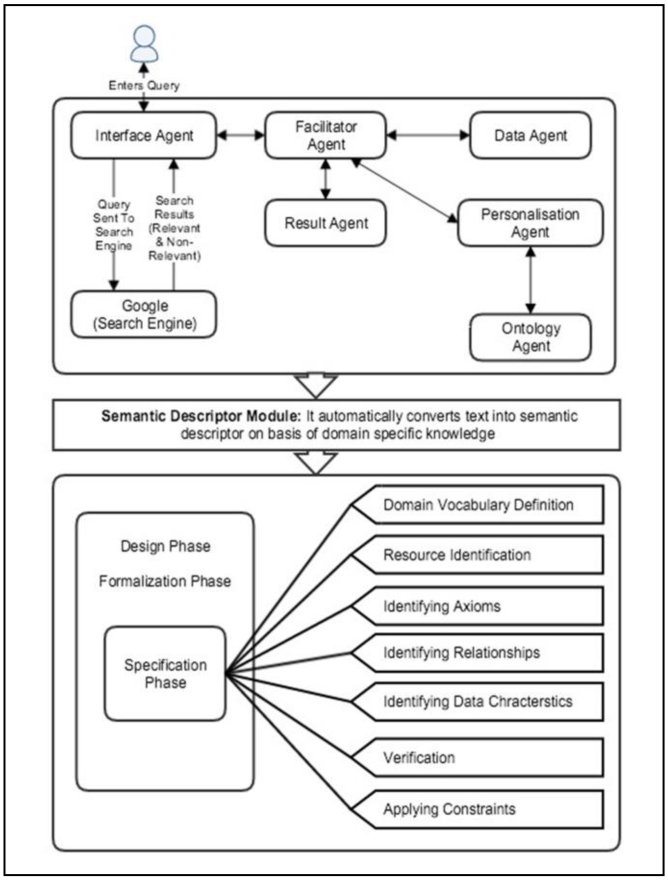
Enters Query
Interface Agent
Facilitator Agent
Data Agent
Query Sent To Search Engine
Search Results (Relevant & Non-Relevant)
Result Agent
Personalisation Agent
Google (Search Engine)
Ontology Agent
Semantic Descriptor Module: It automatically converts text into semantic descriptor on basis of domain specific knowledge
Design Phase
Formalization Phase
Specification Phase
Domain Vocabulary Definition
Resource Identification
Identifying Axioms
Identifying Relationships
Identifying Data Characterstics
Verification
Applying Constraints

Fig. 3. Outline of proposed Multi-Agent system architecture

Figure 4 represents the complete architecture of the proposed search engine.

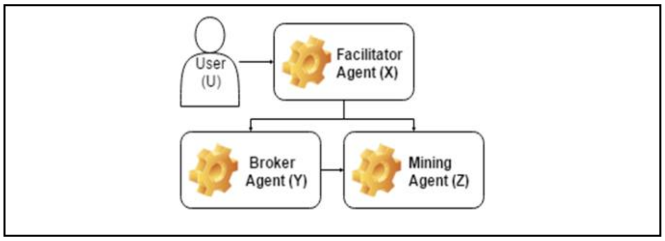

Fig. 4. Flow between Agents

TABLE II. Comparing Effectiveness of Search Using Keywords and Agent Technology

Query	Search using keywords (GOOGLE)	Search using agent search engine (semantic descriptors)
Query 1	45%	54%
Query 2	56%	87%
Query 3	64%	85%
Query 4	60%	70%
Query 5	62%	78%
Query 6	44%	65%
Total	55.2%	73.2%

The results are sent to various agents for achieving refined results by involvement of data mining agent as well. Ontology agent is responsible for learning knowledge about given domain. It does not build respective ontology. Then descriptors of search results are built. It can be done automatically. The results are analyzed by block of automatic descriptor building in order to improve accuracy of results. For example, user has searched about computer components. The block of query descriptor building creates the following query descriptor after passing through ontology module. Domain ontology is built on computer appreciations using one of ontology editor called as HOZO. Hozo is different from other ontology editors in following aspects as it is user friendly environment lets users to work easily on it. Hozo has API named as HozoAPIver 1.15 that accesses existing ontology Inheritance information is clear and easily accessible by two options: One is from Super Classes through is-a link. Other is from Class constraint.

A. Analysis of search results using GOOGLE and proposed search engine

After manual analysis of results, the first 50 relevant documents results are put in table for both traditional search engine GOOGLE and semantic descriptors used in search engine. The results are given in table II. From table II, it can be said that results retrieved from search engine with semantic descriptors are 20-22% higher than that obtained from GOOGLE.

VII. Conclusion and Future Work

A large number of crawlers are present for collecting information from web pages. They retrieve data securely but their distribution and indexing is not satisfactory. The resources are scattered at one central location that creates penalty on network bandwidth. The solution to this bottleneck is use of multiple agents that access information from various pages. The goal of our study in this paper is to propose search engine that allows use of agents to perform various operations in retrieving search results of user. Semantic descriptor is used to map produced results to specific knowledge. The engine is based on ontology domain that is built using ontology development phases to ensure hierarchical representation of results. Use of agent technology allows more refinement in search results thus providing description about concepts used in results. The researched focused on creating personalized result using profiles (implicit and explicit). User is provided with option to choose the level of personalization which provides comfort to the user. Our future work will focus on improving the response time by comparing various technologies. Also we would focus on implementing levels of security to personalization which is still the area of concern for personalization. Users are interested in getting personalized results but owing to security concern of their personal information they are hesitant towards it. So we will try and focus on developing a mechanism which help user to have personalized research keeping the security intact.

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. InWeb 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] U. Fayyad, R. Uthuruswamy, "Data Mining and Knowledge discovery in databases", "*Communications of the ACM, 39(11)*", 1996, pp 1-15

[5] Nicolis G., Prigogine I.: Self-organization in non equilibrium systems: New-York, Jonh Wiley & Sons (1977)

[6] Gagandeep Singh, Dr. Mayank Singh, "Implementation of multi agent systems with ontology in data mining", "*International Journal Of Research In Computer Application & Management (IJRCM), Vol. No. 3, Issue No.1 ISSN 2231-1009*", January 2013, pp 111-117

[7] N.F. Noy, M.A. Musen, "Algorithm and tool for automated ontology merging and alignment", "*Proceedings 17th National Conference on Artificial Intelligence (AAAI-2000)*", Nov 2000

[8] A. Doan, J. Madhavan, A. Halevy, "Ontology matching: machine learning approach", "*Handbook* on Ontologies in Information Systems, S. Staab and R. Studer", May 2004, pages 397-416

[9] Gagandeep Singh, Comparative study of search engine and semantic search engine: a survey"in National Conference named NCACT-2013, pp 57-61, organized by Department of Computer Science & Applications, M.D. University, Rohtak (Haryana) on March 30, 2013

[10] Gagandeep Singh, Vishal Jain, "Information retrieval (ir) through semantic web (sw): an overview", In proceedings of CONFLUENCE-The Next Generation Information Technology Summit, 27-28 September 2012, pp 23-27.

[11] Gagandeep Singh Narula, "Ontology development using hozo and semantic analysis for information retrieval in semantic web" in 'ICIIP-2013 IEEE Second International Conference on Image Information Processing with IEEE Conference Record Number 31034 on December 9-11, 2013'.

[12] T. Gruber, "The role of common ontology in achieving sharable, reusable knowledge bases" principles of knowledge representation and reasoning: "*Proceedings of the Second International Conference, Cambridge, MA*", 1991, pages 601-610

[13] C. Giannella, R. Bhargava and H. Kargupta, "Multi-Agent systems and distributed data mining", "*Lecture Notes in Computer Science*", 2005, pages 1-15

[14] Fang Liu, C. Yu and Weiyi Meng, "Personalized Web search for improving retrieval effectiveness," in IEEE Transactions on Knowledge and Data Engineering, vol. 16, no. 1, pp. 28-40, Jan. 2004. K. Elissa, "Title of paper if known," unpublished.

[15] K. W. T. Leung and D. L. Lee, "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.

[16] K. W. T. Leung, D. L. Lee and W. C. Lee, "Personalized Web search with location preferences," Data Engineering (ICDE), 2010 IEEE 26th International Conference on, Long Beach, CA, 2010, pp. 701-712.

[17] L. Zhuhadar and O. Nasraoui, "Personalized Search Based on a User-Centered Recommender Engine," Web Intelligence and Intelligent Agent Technology (WI-IAT), 2010 IEEE/WIC/ACM International Conference on, Toronto, ON, 2010, pp. 200-203.

[18] T. Robal and A. Kalja, "Learning from users for a better and personalized web experience," Technology Management for Emerging Technologies (PICMET), 2012 Proceedings of PICMET '12:, Vancouver, BC, 2012, pp. 2179-2188.

[19] T. Park. Toward a theory of user-based relevance: A call for a new paradigm of inquiry. J. Am. Soc. Inf. Sci., 45(3):135–141, 1994.

[20] X. Shen, B. Tan, and C. Zhai. Context-sensitive information retrieval using implicit feedback. In SIGIR '05: Proceedings of the International ACM SIGIR Conference, pages 43–50, New York, NY, USA, 2005. ACM.

[21] S. Dumais, E. C. J. J. Cadiz, G. Jancke, R. Sarin, and C. R. Daniel. Stuff i've seen: a system for personal information retrieval and re-use. In SIGIR'03: Proceedings of the International ACM SIGIR Conference, pages 72–79. ACM Press, 2003.