TMRF e-Book
Advances in Semantic Computing
(Eds. Joshi, Boley & Akerkar), Vol. 2, pp 79- 102, 2010

Chapter 5

Concept based Personalized Web Search

***S. Sendhilkumar*[*] *and T. V. Geetha*[*]**

Abstract Existing personalized web search systems does not take into account relevant pages that go unvisited by the user which might be direct answers to the user's information need. In addition, pages in the result set though not directly relevant to the user's information need might provide a link to relevant pages. Such links can be identified only by performing semantic analysis. This paper is aimed towards identifying such relevant pages through semantic search path analysis and provides an effective personalized web search. This improves web search by providing content and individual based relation between the search query and its relevant web pages.

1 INTRODUCTION

Information retrieval encompasses many types of information access. Web search is an important part of this spectrum of information systems. Web search now represents a significant portion of Web activity. The difficulties encountered when searching the Web fall into four categories (Lawrence Steve and Giles Lee (2000): 1) Problems with the data itself, 2) Problems faced by the users trying to retrieve the data they want, 3) Problems in understanding the context of search requests and 4) Problems with identifying the changes in user's information need.

One common solution to all the above-mentioned problems is personalization (Shahabi and Yi-Shin Chen 2003), which customizes the Web environment for users and helps them search their information need easily. Web personalization is currently known as the key to success in business today and in the future (Allen et al 2001, Eetu Makela 2005, Peng and Lin 2006).

1.1 Personalization

Personalization is about tailoring information about anything, like products, web pages, services, etc., to better fit the user. There are several ways of achieving this. To achieve effective personalization focus must be on the user needs, preferences, interests, expertise, workload, tasks, tailoring information either to large or smaller interest groups. The roots of personalization of information

[*]Department of Computer Science & Engineering, CEG, Anna University, Chennai – 25, India.

systems can be traced back to the early adaptive user-interfaces, personal assistants/agents, and adaptive information retrieval. Most of the approaches started with users' needs, preferences and expertise. Other approaches involve detecting patterns in user behavior when searching for information.

Personalization can be of two types: context oriented and individual oriented (Shahabi and Yi-Shin Chen, 2003). Context includes factors of personalization like the nature of information available, the information currently being examined, the applications in use, when, and so on. Individual oriented search encompasses other elements of personalization like the user's goals, prior and tacit knowledge, past information seeking behaviors, among others. These factors are more generic and are applicable to any Personalized Information Retrieval system.

The factors that affect web search personalization can be broadly divided into two categories: 1) spatial factors like Search queries used, Pages Visited, Semantics between the search queries and the visited pages, Order/sequence of page access, Browsing behaviors/actions, Context of search, User interests, Relation between context of search and the information currently being examined and 2) temporal factors like Page-view time, Query usage time and User's shift in interests.

Unobtrusive monitoring provides positive examples of what the user is looking for, without interfering with the users' normal activity. Heuristics can also be applied to infer negative examples, although generally with less confidence (Steffen Staab and Rudi Studer 2004). This idea has led to content-based recommender systems, which unobtrusively watch users browse the web, and recommend new pages that correlate with a user profile. Another way to recommend pages is by using collaborative recommender systems which does this by asking people to rate explicitly pages and then recommend new pages that similar users have rated highly. The problem with collaborative filtering is that it leads to initial difficulties in obtaining a sufficient number of ratings for the system to be useful. This approach often reduces the accuracy of the recommending results. Hybrid systems, attempting to combine the advantages of content-based and collaborative recommender systems, have proved popular to-date (Shahabi et al 2003; Stuart E. Middleton et al 2001). However none of these systems have tracked user behavior based on user search actions. In addition pages missed by search engines have not been considered for recommendations.

User modeling is typically either knowledge-based or behavior-based (Sendhilkumar and Geetha 2009). Knowledge-based approaches engineer static models of users and dynamically match users to the closest model. Behavior-based approaches use the users' behavior itself as a model, often using machine-learning techniques to discover useful patterns of behavior. As it becomes possible to gather and store more historical information about a user's interactions, it is necessary to develop models that span tasks and applications. The method in (Pretschner A. and Gauch S 1999) learns users' profiles from their surfing histories and re-ranks/filters documents returned by a meta-search engine based on the profiles.

Normally users browse through only a few pages among the top 10 pages (Joachims et al 2005). However, the pages that go unvisited sometimes prove to be important/ relevant to the user's context of search. Hence personalized search systems that heavily depend on user's browsing history might miss such pages. In order to identify relevant pages from unvisited page category, this part of research work focuses towards identification of content based link between the visited and unvisited pages. The content-keywords of relevant pages are compared with the content-keywords of the unvisited pages in the top 30 search results.

Any traces of semantic link between the contents of unvisited pages and that of visited pages aid in deriving the conceptual relation between the relevant visited and unvisited page thereby confirming the relevancy of the later. Such semantic relations between relevant visited pages and relevant unvisited pages provide an abstract view of a search query linked with a set of relevant pages from both visited and unvisited category. The system proposed generates a graph based user profile using the set of pages visited by a user. The page data and the query data needed for constructing user profile are collected from a matrix structure called the Transaction Feature Matrix (TFM) that highlights an individual's browsing history (Sendhilkumar and Geetha 2007).

1.2 Background

Web search is effective only when it is both context and interest-oriented. In other words, exploitation of the relation between search query and its relevant page based on factors like user actions and page-view time will prove to be the most important factors of personalization. Sendhilkumar S. and Geetha T.V. [Sendhilkumar and Geetha 2009] provide an architecture that binds the various factors of personalization to provide an effective and efficient web search. The proposed personalized search architecture [Sendhilkumar and Geetha 2009] is made up of the following six layers: Presentation, Pre-processing, Data Storage, Data Extraction, Knowledge and Analysis layers. The presentation layer is the place where the user interacts with the personalized search system and the data about the user's search is collected. The pre-processing layer is responsible for the following activities: HTML to text conversion, POS tagging, noun extraction, computing term-weight based on Term-Frequency (TF) and Inverse Document Frequency (IDF) and feature selection. The data collected by the browser from the user end are populated into tables present in the data layer. The data layer acts as a central warehouse of all the data needed for achieving an effective personalization. The data extraction layer is essential for deriving the conceptual relations between the search queries and their relevant pages.

In addition, to aid users who rely on the Internet for information, a new search aiding index [Sendhilkumar and Geetha 2005] called User Conceptual Index (UCI) has been proposed which is derived in the data extraction layer. The UCI is aimed towards providing both context and user-oriented search. Our approach is an attempt to derive and quantify a personalized relation between search queries and relevant pages based on user's previous experience.

The UCI is a vector having three components: 1) frequency component which is the sum of frequency components of the search query weight and the page weight, 2) time component which is the sum of time components of the search query weight and the page weight and 3) the user action component – number of actions (like save, copy/paste, bookmark, print, e-mail) performed by the user. Any (SQ_i, P_i) pair that has the highest UCI values indicates the user's interest on a topic and the page P_i relevant to that topic of interest. The weight of a search query is a vector comprising of three components: frequency, time and user action component, where all these components are factors that affect personalization. The hit-count of a particular search query is an implicit but direct indicator of user's interest and context of search. If the previous search results are not relevant to the user's information need, then the user might modify the search query to fit into their context of search. Hence the usage time of a SQ_i directly indicates whether previous search results were relevant or irrelevant to the user's information need. Thus two important factors of personalization namely, the user's interest and their context of search are thus incorporated into the basic UCI. Similar to search query weight, page weight is also a vector with the following components: frequency, time and user action component. The number of visits to a page P_i indicates page hits. The view time of a page P_i is normalized over the page size and it represents a very important temporal feature that directly affects personalization. High view time for a page P_i indicates the depth of user's interests on that particular page or it can be concluded that the page is of more important to the user's information need. Therefore, features that indicate user interests and context of search are considered in the calculation of page weight.

The UCI is computed based on the search queries issued and the pages visited by the user. However, this UCI does not record details about unvisited relevant pages in the top ranked result pages returned by search engines and hence this information about unvisited relevant pages must be identified and updated into UCI. Semantic approaches shall be used to overcome this limitation of UCI. Hence the knowledge layer has been introduced in the architecture where reference knowledge and personalized knowledge are being stored. Reference knowledge refers to domain ontology and

ODP taxonomy and the personalized knowledge is nothing but user profiles in the form of conceptual graphs. In addition the knowledge layer is the place where semantic search path analysis is performed. The personalized knowledge is generated and represented as conceptual graphs called Personalized Page-View (PPV) graph which is user specific. Analysis layer analyses the user behaviors, interests, search paths which are very essential for final page re-ranking.

This chapter is dedicated towards the semantic analysis performed in the knowledge layer of the personalized search architecture which aims to provide a concept based personalized web search. The following sub-section of this chapter highlights the motivation behind the identification of relevant pages in the unvisited category and construction of user profile.

1.3 Goals

The goals of this research work is two fold: 1) identify and recommend relevant pages that go unvisited and 2) identify shortest search paths which highlight pages not listed by search engine but explored by the user.

The automatically identified user profile is a graph based profile and called as Personalized Page-View (PPV) graph that provides a conceptual link between visited and unvisited pages [Sendhilkumar and Geetha 2008]. The PPV graph provides an optimal (shortest) search path that connects the pages that are direct answers to user's information need. Such conceptual links not only help a personalized search system to identify relevant pages from the unvisited category but also highlights those pages in the result set that links to a relevant page.

The user behavior analysis is an important part of personalized web search because it involves details regarding user navigation through the web. Such user behaviors aid identification of shortest search path or a path that leads to the relevant information. Hence semantic searches attempts to augment and improve traditional search results by using semantic information from resources like concept graphs and thesaurus (Eetu Makela 2005). The individual search behaviors like, the pages visited, the order of visit and the actions performed on a visited page can be used to confirm the context of search derived from the search query (Oard and Kim 1998; Shapira et al 2006). This way, an effective personalization system could decide autonomously: whether or not a user is interested in a specific webpage and, in the negative case, prevent it from being displayed or, the system could navigate through the web on its own and notify the user if it found a page or site of presumed interest.

All the factors that affect web search and that were discussed so far gives a complete scenario of what is still required to improve a users search through the World Wide Web and this requirement in fact makes personalized web search a compelling area for research. The various factors that motivated this research paper are: 1) Searchable index for collected data, 2) Concept based search and 3) Utilization of semantics in a web search like conceptual link between search queries and its relevant pages, shortest semantic search paths, concept based link between visited and unvisited pages.

2 GRAPH BASED USER PROFILE – PERSONALISED PAGE VIEW (PPV) GRAPH

The set of processes towards constructing the graph based user profiles explained here is part of the knowledge layer (Sendhilkumar and Geetha 2009). The knowledge layer collects information from the data layer and uses taxonomical data for generating the user profiles. The PPV Graph based system (Figure 1) comprises of the following modules: User Behavior Tracking, Pre-processing, PPV graph construction, Path Weight computation and Ranking, and Page Recommendation. The user queries are submitted to an existing search engine. The results given by the existing search engine are analyzed and re-ranked based on user interests and path weights. Finally the personalized search results are recommended to the users in the browser. The users submit their search queries through a specially designed browser. The browser keeps track of all the user data like the search queries, pages visited, time spent on a page and actions performed (save, copy, print and bookmark) by the users during their search sessions. Thus the user data is collected implicitly from the user end which is the primary functionality of user behavior tracking module. The preprocessing module is responsible for transforming user browsing data into representative concepts. The preprocessing module performs the following activities: 1) HTML to text conversion, 2) POS tagging, 3) noun extraction, 4) Computing term-weight based on Term-Frequency (TF) and Inverse Document Frequency (IDF) and 5) feature selection. PPV graph construction module generates the graph based user profiles called the PPV graph form the concepts extracted in the preprocessing module. A typical search can result in many semantic paths semantically linking the entities of interest. Because of the expected high number of paths, it is likely that many of them would be regarded as irrelevant with respect to the user's domain of interest. The path weight computation module determines the weight for various search paths explored by the user and thus helps to filter out irrelevant search paths. Thus, the semantic associations need to be filtered according to their perceived relevance. Ranking approach defines a path rank as a function of various intermediate path weights.

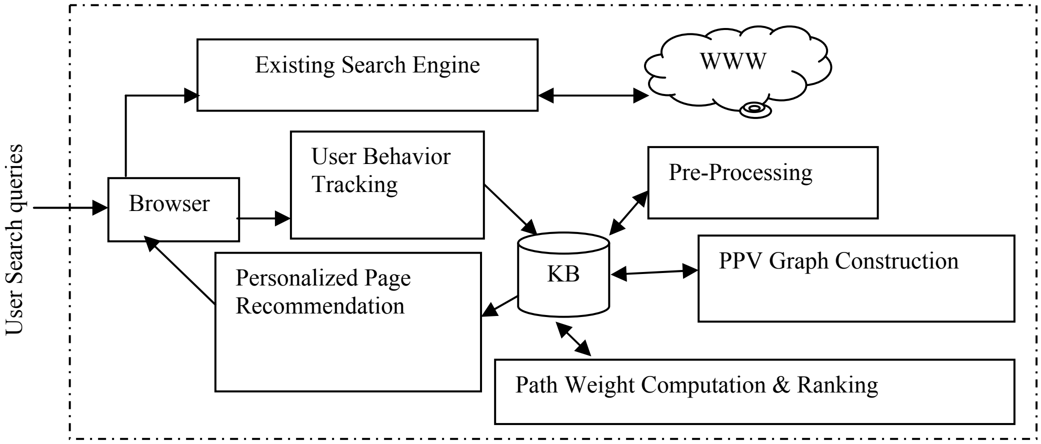


Figure 1. Personalized Page View (PPV) Graph Construction System

The index words extracted from page content in the pre-processing layer (Sendhilkumar and Geetha 2009) and the terms in search queries are used to build the graph based user profile which is represented as a PPV graph. The methodology proposed in this work for the generation of user profiles differs from the majority of other famous works (Shahabi et al 2002; Gauch et al 2003; Trajkova and Gauch 2004; Lora et al 2007). In this research work, the user profiles are represented by Personalized Page-View (PPV) graph that highlight content relation between visited pages and relevant unvisited pages, and they highlight search paths that lead to relevant information.

The PPV graph is an incremental graph and it is constructed based on the set of pages visited by the user. The page link weights based on user's actions (like save, print, copy, and bookmark) are computed and updated in the PPV graph. The PPV graph is constructed semi automatically from the set of pages visited by the user. The processes involved in the construction of PPV graph [Sendhilkumar and Geetha 2008a] is shown in Figure 2.

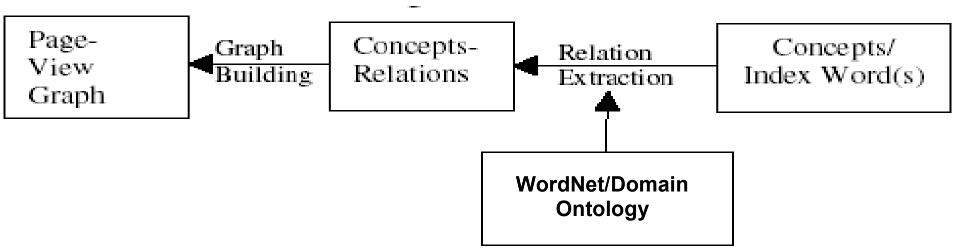
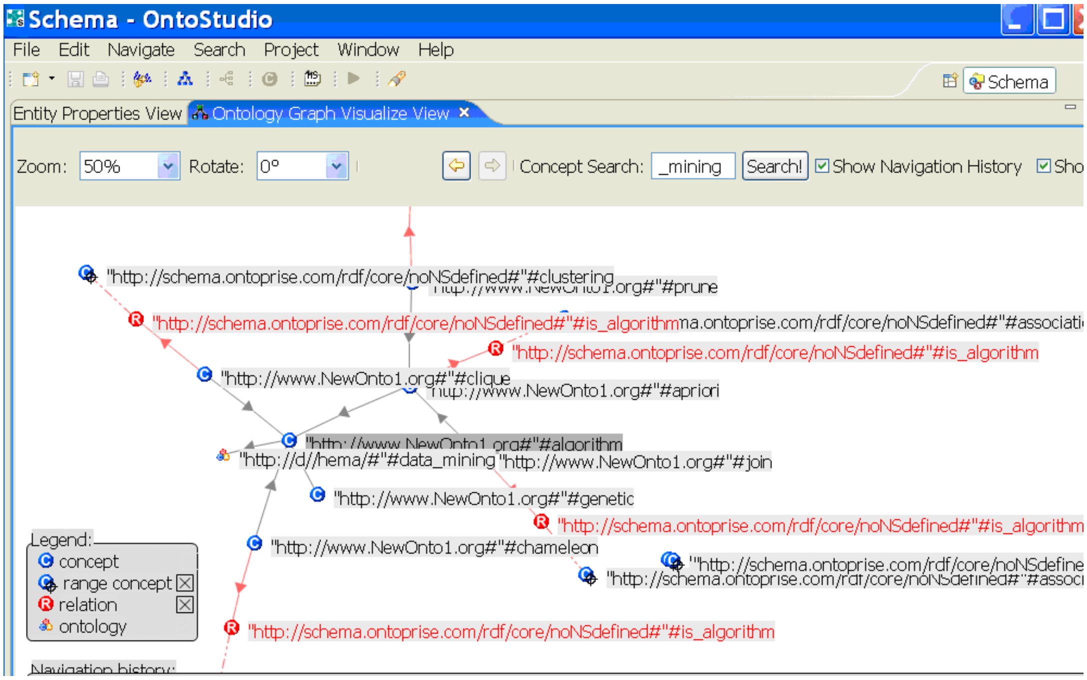


Figure 2. Steps in Personalized Page-view (PPV) Graph Construction

The pages visited, represented by their index words is the input to the PPV graph construction module. The content based relation between the visited and unvisited pages are extracted using WordNet (2006). Nouns in a natural language sentence are connected by the verbs that participate in that sentence. Apart from the relations extracted from WordNet, other relations that are defined by the verbs in the visited pages are also extracted automatically. The extracted concepts and relations are given to java API, JGraph (2009) to visualize the conceptual graphs. JGraph is a graph drawing open source software component written in the Java programming language; started by Gaudenz Alder as a University project in 2000 at ETH Zurich, Switzerland.

2.1 Construction of Domain Conceptual Graphs

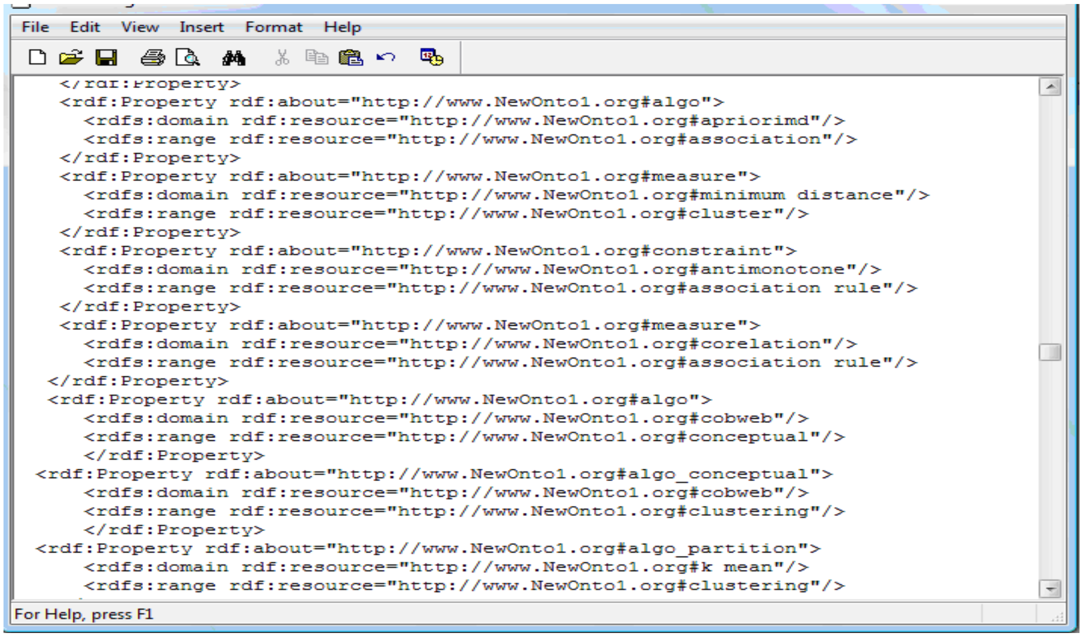
Apart from WordNet a domain specific conceptual graph for computer domain and medical domain has been used. Domain specific concepts from are collected from ODP taxonomy (ODP 2009) and a domain dictionary is manually constructed with those important concepts using domain experts. Using domain dictionary, domain specific conceptual graph is constructed and visualized using OntoStudio 2.0 (OntoStudio Manual 2007). Concepts, their properties, sub concepts, their relations with other concepts are identified from the dictionary and entered in the ontology tool. The domain specific conceptual graph is generated in RDF (Resource Description Framework) format and visualized as a graph. The steps involved in the construction of domain specific conceptual graph is shown in algorithm 1 where the input is the manually collected concepts and relations of the domain and the output is the domain conceptual graph in RDF schema.



Zoom: 50%   Rotate: 0°   Concept Search: _mining   ☑ Show Navigation History   ☑ Sho

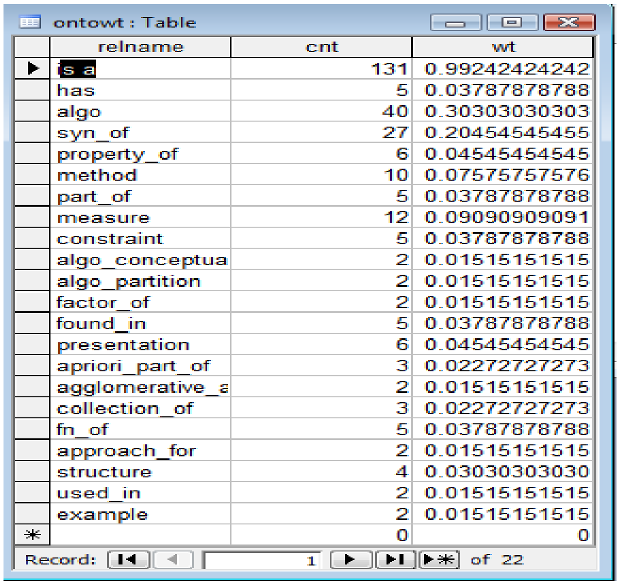
Schema - OntoStudio
File Edit Navigate Search Project Window Help
Schema
Entity Properties View
Ontology Graph Visualize View
Search!
"http://schema.ontoprise.com/rdf/core/noNSdefined#"#clustering
"http://www.NewOnto1.org#"#prune
"http://schema.ontoprise.com/rdf/core/noNSdefined#"#is_algorithm
"http://schema.ontoprise.com/rdf/core/noNSdefined#"#associati
"http://schema.ontoprise.com/rdf/core/noNSdefined#"#is_algorithm
"http://www.NewOnto1.org#"#clique
"http://www.NewOnto1.org#"#apriori
"http://www.NewOnto1.org#"#algorithm
"http://d//hema/#"#data_mining
"http://www.NewOnto1.org#"#join
"http://www.NewOnto1.org#"#genetic
"http://schema.ontoprise.com/rdf/core/noNSdefined#"#is_algorithm
"http://www.NewOnto1.org#"#chameleon
"http://schema.ontoprise.com/rdf/core/noNSdefine
"http://schema.ontoprise.com/rdf/core/noNSdefined#"#associ
"http://schema.ontoprise.com/rdf/core/noNSdefined#"#is_algorithm
Legend:
concept
range concept ☒
relation ☒
ontology
Navigation history:

Figure 3. Domain-Specific Conceptual Graph



```
    </rdf:Property>
    <rdf:Property rdf:about="http://www.NewOnto1.org#algo">
      <rdfs:domain rdf:resource="http://www.NewOnto1.org#apriorimd"/>
      <rdfs:range rdf:resource="http://www.NewOnto1.org#association"/>
    </rdf:Property>
    <rdf:Property rdf:about="http://www.NewOnto1.org#measure">
      <rdfs:domain rdf:resource="http://www.NewOnto1.org#minimum distance"/>
      <rdfs:range rdf:resource="http://www.NewOnto1.org#cluster"/>
    </rdf:Property>
    <rdf:Property rdf:about="http://www.NewOnto1.org#constraint">
      <rdfs:domain rdf:resource="http://www.NewOnto1.org#antimonotone"/>
      <rdfs:range rdf:resource="http://www.NewOnto1.org#association rule"/>
    </rdf:Property>
    <rdf:Property rdf:about="http://www.NewOnto1.org#measure">
      <rdfs:domain rdf:resource="http://www.NewOnto1.org#corelation"/>
      <rdfs:range rdf:resource="http://www.NewOnto1.org#association rule"/>
   </rdf:Property>
   <rdf:Property rdf:about="http://www.NewOnto1.org#algo">
      <rdfs:domain rdf:resource="http://www.NewOnto1.org#cobweb"/>
      <rdfs:range rdf:resource="http://www.NewOnto1.org#conceptual"/>
      </rdf:Property>
  <rdf:Property rdf:about="http://www.NewOnto1.org#algo_conceptual">
      <rdfs:domain rdf:resource="http://www.NewOnto1.org#cobweb"/>
      <rdfs:range rdf:resource="http://www.NewOnto1.org#clustering"/>
      </rdf:Property>
  <rdf:Property rdf:about="http://www.NewOnto1.org#algo_partition">
      <rdfs:domain rdf:resource="http://www.NewOnto1.org#k mean"/>
      <rdfs:range rdf:resource="http://www.NewOnto1.org#clustering"/>
```

Figure 4. Domain Specific Conceptual Graph in RDF schema



	relname	cnt	wt
▶	is a	131	0.99242424242
	has	5	0.03787878788
	algo	40	0.30303030303
	syn_of	27	0.20454545455
	property_of	6	0.04545454545
	method	10	0.07575757576
	part_of	5	0.03787878788
	measure	12	0.09090909091
	constraint	5	0.03787878788
	algo_conceptua	2	0.01515151515
	algo_partition	2	0.01515151515
	factor_of	2	0.01515151515
	found_in	5	0.03787878788
	presentation	6	0.04545454545
	apriori_part_of	3	0.02272727273
	agglomerative_a	2	0.01515151515
	collection_of	3	0.02272727273
	fn_of	5	0.03787878788
	approach_for	2	0.01515151515
	structure	4	0.03030303030
	used_in	2	0.01515151515
	example	2	0.01515151515
*		0	0

Figure 5. Relation Weight

Algorithm 1 Domain Specific Conceptual Graph Construction

```
1.  Create root concept
2.  For each concept in the domain dictionary
3.   If (main concept)
        Add concept below the root
    else
        Add concept below corresponding sub concept
4.  For each concept
5.      If (concept property exists)
            Add properties to the concept
6.      If (concept relation exists)
        Add relations to the concept
7.  Generate in RDF format
8.  End
```

The weights of each relation between concepts are computed using the relation term frequency (Figure 5). The relation term frequency is computed from the set of web pages visited by the user. The concepts and relations are visualized (Figure 3) with the help of OntoVisualize option in the tool. The constructed domain specific conceptual graph is then exported and saved in RDF format (Figure 4) for extracting information.

2.2 Evaluation of Domain Specific Conceptual Graphs

The constructed domain specific conceptual graph is evaluated using ontological evaluation metrics like relationship richness (equation 1) and inheritance richness (equation 2) (Cross and Pal 2005; Yang et al 2006).

$$\text{Relation Richness} \quad RR = \frac{|P|}{|P| + |SC|} \tag{1}$$

$$\text{Inheritance Richness } IR = \frac{\sum_{C_i \in C} \left| H^{C}(C_1, C_i) \right|}{|C|} \tag{2}$$

where, P is total number of relationships, SC indicates the number of subclasses, $H^{C}(C_1, C_i)$ is the number of subclasses C_1 for a class C_i .

Table 1 summarizes the components of computer domain specific conceptual graph. The relationship richness and inheritance richness are computed as per equations (1) and (2) using the data in Table 1. It is observed that Relationship Richness (RR) = 0.8154 and Inheritance Richness (IR) = 0.2262. The value RR is closer to one which indicates that most of the relationships are other than class-subclass which shows that the domain specific conceptual graph is richer. As the IR value is low, the domain specific conceptual graph developed is of vertical nature which implies that a very detailed type of knowledge is represented by it.

Table 1. Components of Computer Domain Conceptual Graph

Components	Count
Total number of relationships	**137**
Total number of class	**137**
Total number of subclasses	**31**
Total number of class and subclasses	**168**

2.3 Construction of PPV Graphs

Consider the sample index words {DIVISIVE, DECISION TREE, ENTROPY, CLUSTER, SUPERVISED} from a relevant visited page http://ieeexplore.ieee.org/iel5/69/4362708/04358941.pdf. The extracted concepts are then mapped with the concepts in domain specific conceptual graph represented in RDF scheme. For concepts which do not have exact matching in the domain specific conceptual graph, taxonomical information from ODP taxonomy is used for identifying equivalent concepts which are then used for mapping. SAX parser is used to parse the RDF file and gather the related concepts for the input concept given. The neighbouring concepts of each extracted concept like its parent, its child, its sub-child, relations are found from the ontology using the parser. The index words are then mapped with the concept in the domain specific conceptual graph and the results of the mapping is shown in Figure 6.

After the mapping process it is found that terms like {DATA MINING, MACHINE LEARNING, DATA AND HIERARCHICAL} are semantically related to the input index words {DIVISIVE, DECISION TREE, ENTROPY, CLUSTER, SUPERVISED}. Finally the entire set of terms {DATA MINING, MACHINE LEARNING, DATA AND HIERARCHICAL, DIVISIVE, DECISION TREE, ENTROPY, CLUSTER, SUPERVISED} are used to search for related pages from the page set that has the pages left unvisited by the user in the top 30 result pages returned by existing search engine as a result for the query 'automatic divisive approaches for clustering'.

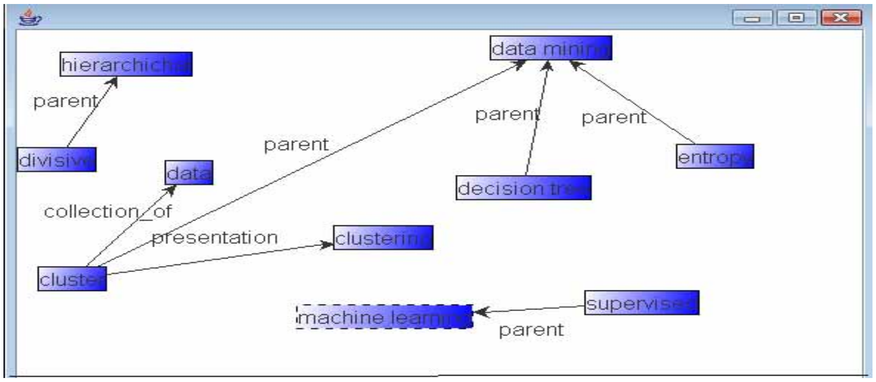


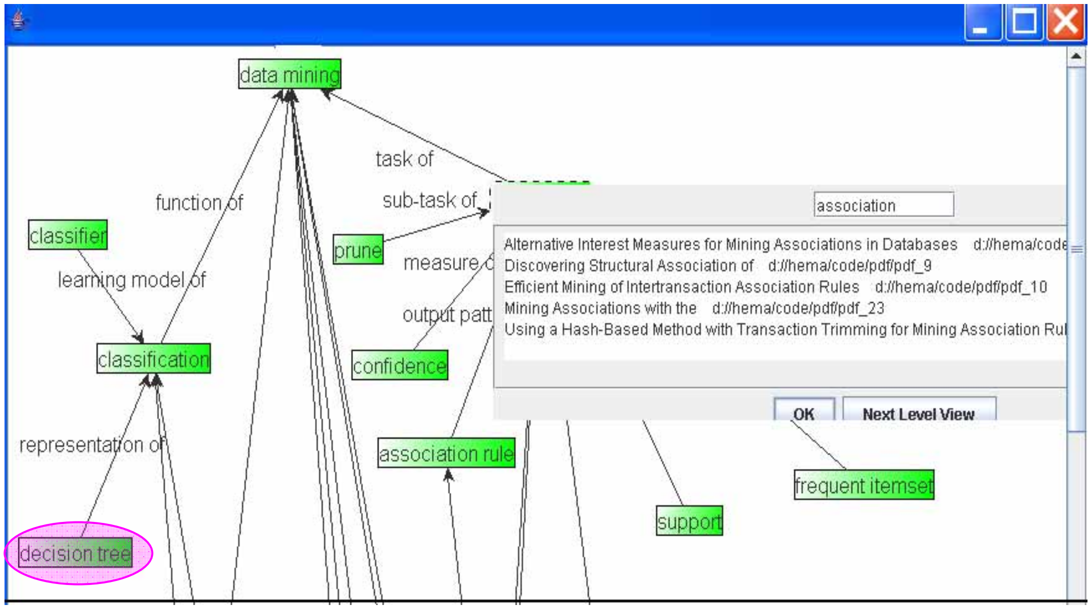
Figure 6. Mapping Index Words of Pages Visited with Domain-Specific Conceptual Graphs

The index words are denoted by a vector with their weights and each retrieved page is represented by a vector with their page weight. Finally cosine similarity is computed between the index words selected and the index words of the unvisited pages. As a result of these analysis the page titled 'Interpretable hierarchical clustering by constructing an unsupervised decision tree' (ieeexplore.ieee.org/iel5/69/29880/01363769.pdf) from the unvisited page category in the top 30 results returned by existing search engine is found relevant to the given query. Therefore the page is recommended to the user.

As to provide good user interaction with the PPV graph constructed, a graphical representation of the PPV graph is implemented in java using JGraph interface. In the graphical representation in Figure 7, the index words are represented in nodes and relations in edges. When the user double clicks a node, the pages related to the given index word are represented by title and the path are displayed in a new window and this feature is supplied as an add-on to the browser. The index words of visited page are highlighted with different coloured nodes where a node with index word 'decision tree' is highlighted with a circle.

The relevant pages that go unvisited by the users are thus identified and recommended to the user. Such pages prove to be the direct answers to the users information need. Thus our system takes into account those pages that are left unnoticed by the user for final page recommendations.

Similarly for sample queries like 'Heart Attack Causes', 'Heart Failure Causes', 'Drugs for Heart Attack' and 'Blood Pressure' issued in a search session, pages that are visited by the user are collected. Few of the sample Index Words that are extracted like {PRESSURE, BLOOD, HEART, STRESS, DRUGS, ASPIRIN, BLOCKER, HYDROCHLORIDE HEART_FAILURE, STROKE, KIDNEY_FAILURE, HEART_ATTACK, and BLOOD_PRESSURE} represent the pages visited by the user during his/her search session. These index words and the extracted relations/properties like {'Leads to', 'Drugs for'} are given to the java visualization module.



association

```
Alternative Interest Measures for Mining Associations in Databases    d://hema/code
Discovering Structural Association of    d://hema/code/pdf/pdf_9
Efficient Mining of Intertransaction Association Rules    d://hema/code/pdf/pdf_10
Mining Associations with the    d://hema/code/pdf/pdf_23
Using a Hash-Based Method with Transaction Trimming for Mining Association Rul
```

Figure 7. Visualization of PPV Graphs with Links to Relevant Unvisited Pages

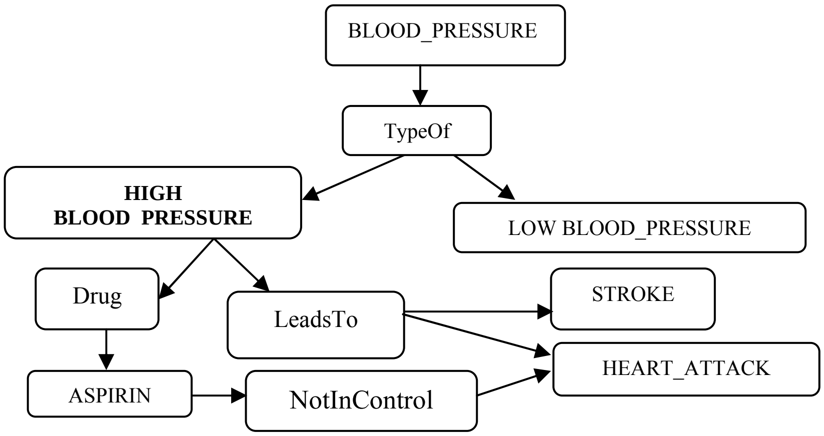


Figure 8. Sample Page-View Graph for the query 'Blood Pressure'

The sample PPV graph constructed for the query 'Blood Pressure' is shown in Figure 8. The RDF generated highlights that HIGH_BLOODPRESSURE property 'leads to' HEART_FAILURE, STROKE, KIDNEY_FAILURE and HEART_ATTACK. The concepts HEART_FAILURE, STROKE, KIDNEY_FAILURE and HEART_ATTACK are the subclass of BLOOD_PRESSURE. The subclasses of BLOOD_PRESSURE are grouped by '[protege-owl] Union of' tag. The next important task of our personalized search system is to identify search paths and rank them based on their conceptual path length weight. The path length weight also accounts for personalized factors like user actions.

3 PATH WEIGHT COMPUTATIONS AND RANKING

A typical search can result in many semantic paths semantically linking the entities of interest. Because of the expected high number of paths, it is likely that many of them would be regarded as irrelevant with respect to the user's domain of interest. Thus, the semantic associations need to be filtered according to their perceived relevance. Ranking approach [Saravanakumar et al 2008] defines a path rank as a function of various intermediate weights. The weights involved in the path-weight computation are: 1) Concept Weight (C_i), 2) Path Length Weight (PL_i) and 3) Personalized Context Weight (C_P).

3.1 Concept Weight

When considering concepts in graph, those that are lower in the hierarchy can be considered to be more specialized instances of those further up in the hierarchy. For the sample blood pressure hierarchy given in Figure 8, the HEART_ATTACK conveys more meaning than BLOOD_PRESSURE, HIGH_BLOOD_ PRESSURE. Higher weights are assigned to more 'specific' semantic associations because they convey more meaning then 'general' associations. The weight for a concept in the hierarchy is computed based on the formula in equation (3).

$$\text{Concept Weight } C_i = \frac{H_{Ci}}{H} \qquad (3)$$

where H_{Ci} is the level of ith concept in the PPV hierarchy; H refers to total height of the hierarchy. The concept weight computed for various concepts in Figure 8 using equation (3) is as given below:

BLOOD_PRESSURE $C_1 = H_{c1} / H = 1/6 = 0.17$
HIGH_BLOOD_PRESSURE $C_2 = H_{c2} / H = 3/6 = 0.50$
LOW_BLOOD_PRESSURE $C_3 = H_{c3} / H = 3/6 = 0.50$
ASPIRIN $C_4 = H_{c4} / H = 5/6 = 0.83$
HEART_ATTACK $C_5 = H_{c5} / H = 5/6 = 0.83$
STROKE $C_6 = H_{c6} / H = 5/6 = 0.83$

3.2 Path Weights

The path weights are context specific. For some queries, a user may be interested in the most direct paths (i.e., the shortest path – a link from one page leading to the most other relevant page(s)). This may infer a stronger relationship between the concepts in two different pages. Hence, path length must be determined and should be used. The path weight PL_i is computed using equation (4).

$$\text{Path Weight } PL_i = \frac{1}{|C|} \tag{4}$$

where, $|C|$ is the number of components in the path P (excluding the start and end concepts). A component in a path refers to both the concept and relation.

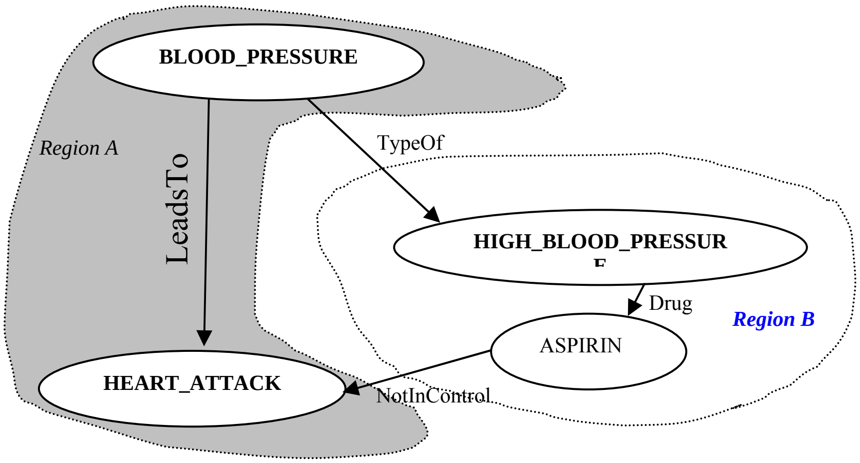


Figure 9. Sample Paths in Conceptual graphs

The sample paths given in figure 9 highlight that there exists two paths: 1) direct path PL_1 in region A (from the page that contains the concept BLOOD_PRESSURE the user has clicked a link leading to another page containing the concept HEART_ATTACK which is the direct answer to user's information need) and 2) longer path PL_2 in region B (from the page that contains the concept BLOOD_PRESSURE the user visited another page containing the concept HIGH_BLOOD_PRESSURE, which in turn linked the user to a new page about a drug ASPIRIN and finally the user moved from ASPIRIN page to the relevant page that speaks about HEART_ATTACK which is the direct answer to the user's information need) between BLOOD_PRESSURE and HEART_ATTACK. The path weights are computed as given below:

For the direct path $PL_1 = 1/C = 1/1 = 1$

For the longer path $PL_2 = 1/C = 1/5 = 0.20$

Thus it can be observed that $PL_1 > PL_2$, hence the longer paths are ranked lower than the shortest paths. A higher path weight highlights the closeness of the concept(s) currently searched by the user with respect to the page that is currently viewed by the user and hence path weights are context specific.

3.3 Personalized Context Weight

The user can perform any of the actions mentioned in Table 2 on any page which, he/she visits. According to the action performed weights are assigned to the respective pages.

Table 2. User Actions and Respective Weights

Actions	**Page Relevancy to user's context of search**	**Weights Assigned**
Save	Highly relevant	1
Print	Highly relevant	1
Copy	Partially relevant	0.25 to 0.75 depending upon the amount of the content copied
Book Marking	High/Partial according to the page usage	1 – used in same session 0.5 - used in another session

The user has performed the action 'Save' on the page represented by the concept/index word HEART_ATTACK, hence, a weight 1 is assigned to that page. Similarly the user has copied one third and almost half of the textual contents of the pages represented by the concepts HIGH_BLOOD_PRESSURE and ASPIRIN; hence, a weight 0.25 and 0.5 respectively.

The weight assignment highlights that the user is more interested in the page represented by the concept/index word HEART_ATTACK and also wants to consider the associations between BLOOD_PRESSURE and DRUGS, but with lesser priority. Now the personalized context weight C_P for a path traversed by the user while searching for his/her information need is calculated using the context weight formula given in equation (5).

$$C_P = \frac{1}{|C|}\sum_{i=1}^{m}\left[r_i \times (c_i \in R)\right] \qquad (5)$$

where, r_i is the weight assigned according to the action performed on a page P_i (if no action performed ignore r_i and just use the concept weight alone), c_i is the concept weight in the path P, m is the maximum number of concepts along a path P and $|C|$ is the number of components in the path (again excluding the start and end entities) i.e. for each concept that P passes through, sum the total number of components in P that are in the region R_i and multiply it by the weight attributed to that region r_i. The overall path weight of a path P denoting a semantic association between two pages P_i and P_j will be a linear function as in equation (6).

$$W_{P_i \to P_j} = C_P + P_L \qquad (6)$$

Some of the sample relations that can be extracted from the Figure 6 are:

Relation 1: BLOOD_PRESSURE<LeadsTo>HEART_ATTACK
Relation 2: BLOOD_PRESSURE<TypeOf>HIGH_BLOOD_PRESSURE
<Drugs> ASPIRIN <NotInControl>HEART_ATTACK

Relation 3: BLOOD_PRESSURE <TypeOf> HIGH_BLOOD_PRESSURE <LeadsTo> HEART_ATTACK

These semantic relations are then ranked according to the overall path weight $W_{P_i \to P_j}$. Table 3 depicts the weight computation for the extracted semantic relations and the ranks assigned according to the computed weights. Consider the semantic relation 1 and 2 in region A and region B respectively (Figure 9). Let us assume that the user is interested in region A. While computing the personalized weight the page-view time is added with the action weight. The page-view time (in seconds) is normalized by dividing it by 1530 seconds (25.5min) which is the average session out time. From Table 3 it can be observed that the overall weight of relation 1 is greater than that of relation 2 and according to their weights, ranks has been assigned.

Table 3. Weight Assignments and Ranking of Semantic Relation

Semantic Relation No.	**Path Length Weight** (PL_i)	**Personalized Context Weight** (C_P)	**Overall Path Weight** $W_{Pi \to Pj} = PL_i + C_P$	**Rank**
1	= 1/1 = 1	= 1/1 *(0.17+1*0.83) =0.1411 [No action performed in page represented by C1 ; hence ignore r1]	= 1+ 0.1411 = 1.1411	1
2	= 1/5 = 0.20	=1/5 * (0.17+0.25*0.50+0.5*0.83+1*0.83) = 0.318	=0.20+0.318 =0.518	2

Such kinds of relations between various concepts from the collected web pages provide semantic relations between the pages. Also the constructed graph is very useful to analyze the relationships between the collected most frequently occurring patterns and the links available in a web page. Thus the extracted semantic relations from the domain ontology can be ranked according to the user actions and can be used in the page re-ranking process. The PPV graph that is developed for personalized web search is thus different from others by the above-mentioned concept weights. The various semantic paths and the respective path ranks are updated into the data base in the Knowledge layer (Sendhilkumar and Geetha 2009) which will be used for further page recommendation process.

4 EVALUATION OF PPV GRAPHS

The development of Semantic Web has encouraged the creation of conceptual graphs and ontologies in a great variety of domains. The simplest method of evaluation of ontology population task is based on precision and recall. These are typically used in IE (Information Extraction) evaluations such as MUC (Message Understanding Conferences) (Lozano-Tello et al 2003) and CONLL (Introduction to the CONLL shared task 2002 and 2003). Because much of the research in IE in the last decade has been connected with these competitions, the MUC evaluation metrics of precision, recall and F-measure (Chinchor 1992) have been the most widely used in this field, albeit with slight variations from time to time. The OntoMetric (Lozano-Tello et al 2003) method presents a set of processes that the user should carry out to obtain the measures of suitability of existing ontologies, regarding the requirements of a particular system. This work describes methodologies for evaluating the content of conceptual graphs with respect to natural language applications. Natural Language methods can be used for both concept population and semantic metadata creation. The first involves populating ontology of concepts with instances drawn from textual data; the second involves associating the text with the correct concepts in the ontology.

The conceptual graphs constructed as indicated in section 2 is evaluated [Sendhilkumar and Geetha 2008b] using the following metrics suggested by OntoMetric (Lozano-Tello et al 2003) like: 1) Total number of paths (TNOP) 2) Total number of relations (TNOR) and 3) Information content. Two sample concepts 'HIGH_BLOODPRESSURE' and 'HEART_ATTACK' in the PPV graph (figure 8) are considered for explaining the process of evaluation. TNOR is the sum of relations of each concept and is equal to 8 for 'HIGH_BLOODPRESSURE' concept and 7 for 'HEART_ATTACK' concept in WordNet which is shown in Table 4. TNOP is the sum of paths of each concept and is equal to 20 for 'HIGH_BLOODPRESSURE' concept and 31 for 'HEART_ATTACK' concept in the PPV graph constructed that is shown in Table 4.

Table 4. TNOP and TNOR metric

Concept	WordNet		PPV graph	
	TNOP	***TNOR***	***TNOP***	***TNOR***
HIGH_BLOODPRESSURE	**20**	**8**	**10**	**6**
HEART_ATTACK	**31**	**7**	**11**	**10**

Information Content (IC) denotes the level of information content a concept conveys and can be calculated for a concept c using equation (7).

$$IC(c) = \frac{\log\left[\frac{hypo(c)+1}{\max}\right]}{\log\frac{1}{\max_{wn}}} \tag{7}$$

$$= 1 - \frac{\log\left(hypo(c)+1\right)}{\log\left(\max_{wn}\right)}$$

where $hypo(c)$ is the number of hyponyms of a concept and $\max_{wn}$ is the maximum number of concepts. IC of 'HIGH_BLOODPRESSURE' concept is found to be 0.386 and 'HEART_ATTACK' concept is 0.3007 in WordNet and IC for the concepts 'HIGH_BLOODPRESSURE', 'HEART_ATTACK' in PPV graph are found to be 0.3701 and 0.3472. These IC values show the content level of concepts in the whole PPV graph. The primitive measure like TNOC, TNOR and TNOP when measured over the number documents is given in Table 5.

Table 5. Primitive measures Vs Number of Documents

No. of Doc/Primitive measures	50	100	150	200
TNOC	312	456	524	568
TNOR	125	280	489	671
TNOP	180	293	567	725

The other measures used are μ the average relations per concept and ρ the average paths per concept. μ is the ratio of TNOR to TNOC and ρ is the ratio of TNOP to TNOC. Computation of μ and ρ is done according to the equations (8) and (9) respectively. μ indicates the average connectivity degree of a concept and ρ examines the concept aggregation and coherence of conceptual graph.

$$\mu = \frac{TNOR}{TNOC} = \sum_{i=1}^{m} \frac{r_i}{m} \tag{8}$$

$$\rho = \frac{TNOP}{TNOC} = \sum_{i=1}^{m} \frac{p_i}{m} \tag{9}$$

For any conceptual graph, ρ must be greater than or equal to 1 (each concept must have a parent except for the general concept). If ρ =1, then the conceptual graph is a tree (each concept has a single parent, and thus a single path to the most general concept).

The graph in Figure 10 gives the primitive measures with the number of documents. It can be inferred from the graph that as the number of documents increases the number of distinct concepts, relations and paths also increase in Y-axis in units of numbers. But the growth rate in concepts will be reduced as the number of documents increases because of redundant concepts occurring in multiple documents.

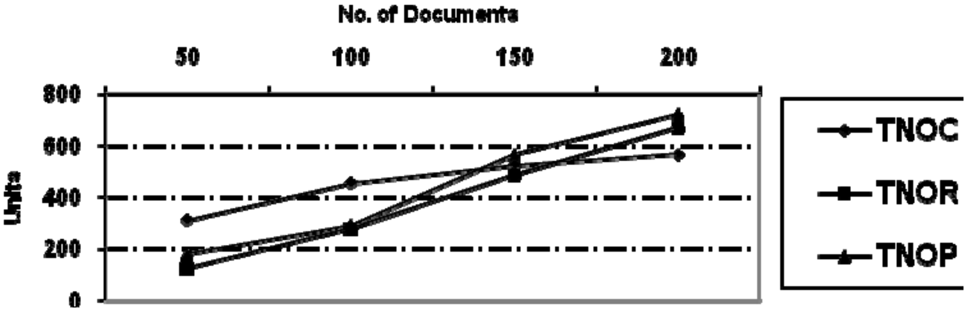


Figure 10. Primitive Measures Vs Number of Documents

However the relations grow at a faster rate than that of concepts. Because new relations will be connected with concepts as more concepts are evolved. Same is true for paths, but it has more exponential growth than relations. Hence as the number of documents increases the conceptual graph gets more connected and new paths evolve at a faster rate.

In Figure 11 X-axis shows the increase in number of documents. The variation is not linear with the growth because the concepts and paths are growing with less correlation. ρ gets to one only after the total number of documents used in corpus has crossed more than hundred and is growing at a better rate after that. The average relations per concept is also growing at better rate which tell us that more connections are established per concept when the number of documents grows above hundred. Hence from the experimental results it is found that it is desirable to have more than hundred documents to be in the document corpus such that the conceptual graph thus constructed is effective to practical use.

A sample search behavior that shows the various semantic search paths and the pages involved in those paths is given in Figure 12. The rectangular box on top indicates the search query and the rest of the boxes are the various pages visited by the user in the order of visit. The pages indicated in green colored boxes in Figure 12 are those pages in which the user has performed some actions. The system actually provides a visual interpretation of pages like: green colored boxes for pages with user actions, red colored boxes for irrelevant pages, etc. The graph in Figure 12 also exhibits the four search paths P_1, P_2, P_3 and P_4 which are indicated by dashed arrows. Such visual representation of various search paths and their ranks are provided to the user as add-on to the browser and the user if needed can check that option and view the visualizations. Hence the users get a visual aid to choose the right search path for reaching their information need.

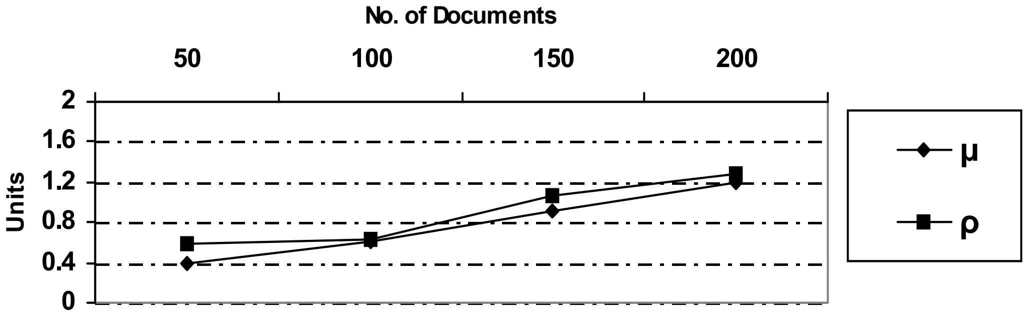


Figure 11. Variation of μ and ρ with respect to Number of Documents

Table 6 indicates the details of the search paths detected in the experiment. The search paths are grouped into three categories: 1) paths involving two pages, 2) paths involving three-five pages and 3) paths involving more than five pages. The pages involved in such path categories are then evaluated based on the user feedbacks. Fifteen users are involved in the evaluation process. Among the 15, eight of them are post graduate students, five are undergraduate students in computer science and engineering, and the other two are research scholars in the same. Hence they all had atleast five years experience of working with computers. All the 15 users performed regular searches and the individual's behavior on every page was tracked. During the evaluation phase the users provided explicit feedbacks for the relevancy of each page which is a minimal disruption to their regular work, but necessary for the experiment. During web search, the users are asked to select one among the following five options: 0 – No Idea, 1- Not Relevant, 2 – Leads to Useful Link, 3 - Partially Relevant, and 4 – Exactly Relevant. The users are also asked to rate the search transaction (definition 3.1) whenever they issue a new search query or modify the previous search query by choosing one of the following four options: 0 – No Idea, 1 – Not Useful, 2 – Partially Useful and 3 – Very Useful. During the evaluation phase 1716 pages are collected from which 28 pages are removed since the users had no opinion about the page's relevancy (rated as 0). Hence the remaining 1688 pages are used for analysis.

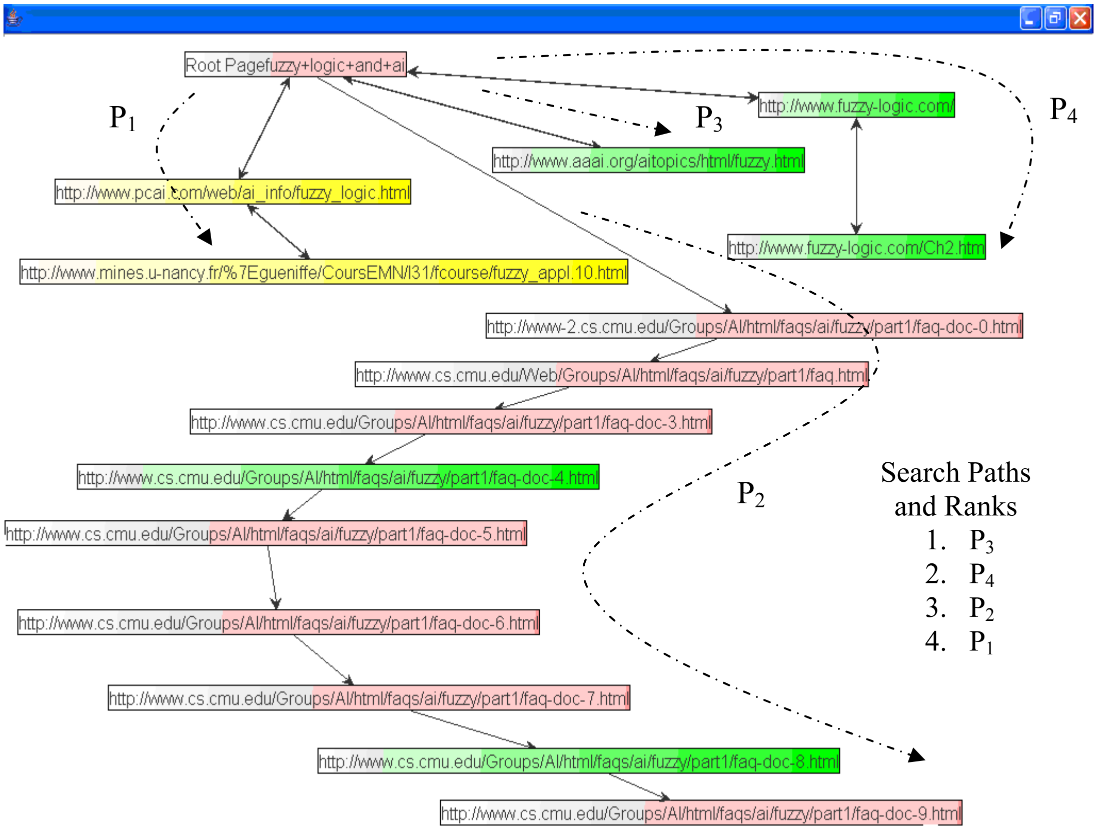

Root Pagefuzzy+logic+and+ai
P1
P3
P4
http://www.fuzzy-logic.com/
http://www.aaai.org/aitopics/html/fuzzy.html
http://www.pcai.com/web/ai_info/fuzzy_logic.html
http://www.fuzzy-logic.com/Ch2.htm
http://www.mines.u-nancy.fr/%7Egueniffe/CoursEMN/I31/fcourse/fuzzy_appl.10.html
http://www-2.cs.cmu.edu/Groups/AI/html/faqs/ai/fuzzy/part1/faq-doc-0.html
http://www.cs.cmu.edu/Web/Groups/AI/html/faqs/ai/fuzzy/part1/faq.html
http://www.cs.cmu.edu/Groups/AI/html/faqs/ai/fuzzy/part1/faq-doc-3.html
http://www.cs.cmu.edu/Groups/AI/html/faqs/ai/fuzzy/part1/faq-doc-4.html
P2
Search Paths
and Ranks
1. P3
2. P4
3. P2
4. P1
http://www.cs.cmu.edu/Groups/AI/html/faqs/ai/fuzzy/part1/faq-doc-5.html
http://www.cs.cmu.edu/Groups/AI/html/faqs/ai/fuzzy/part1/faq-doc-6.html
http://www.cs.cmu.edu/Groups/AI/html/faqs/ai/fuzzy/part1/faq-doc-7.html
http://www.cs.cmu.edu/Groups/AI/html/faqs/ai/fuzzy/part1/faq-doc-8.html
http://www.cs.cmu.edu/Groups/AI/html/faqs/ai/fuzzy/part1/faq-doc-9.html

Figure 12. Sample Search Behavior with Semantic Search Paths

Table 6. Search Path Categories

Total number of search paths detected	156
No. of search paths involving 2 pages (Category A)	38
No. of search paths involving 3-5 pages (Category B)	52
No. of search paths involving >5 pages (Category C)	66

The search path categories and their distribution with respect to explicit user feedbacks are shown in Figure 13. It can be inferred from the graph that majority of the search paths (45 search paths) that are marked 'Not Useful' by the users are Category C paths.

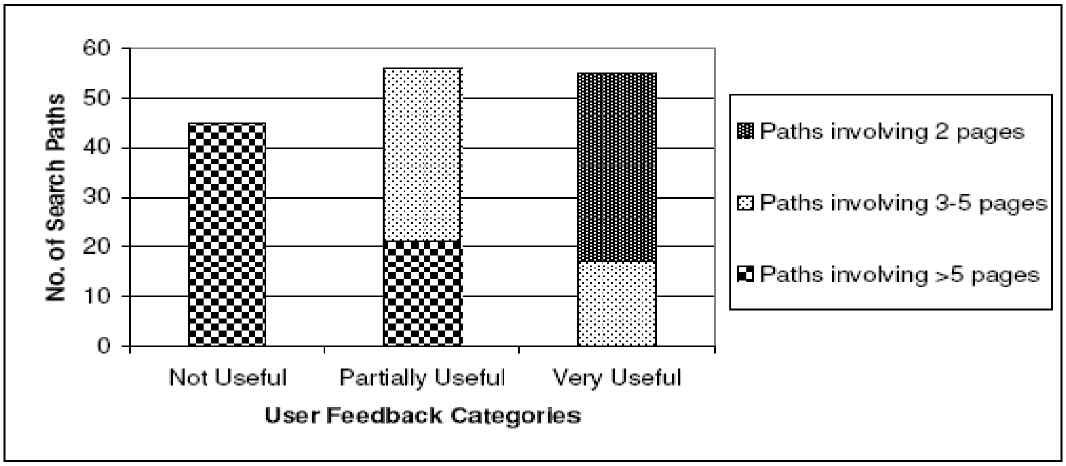


Figure 13. Search Path Distribution

A few number of Category C search paths (21 search paths) are marked ‘Partially Useful’ along with 35 other Category B search paths. This is because the pages that participated in Category C search paths are marked ‘Leads to Useful Link’ by the user. Finally the entire Category A search paths and 17 of the Category B search paths are marked ‘Very Useful’ by the user. Thus from Figure 13 it can be concluded that the entire Category A and Category B search paths are relevant to the users search. It can be further concluded that majority of user actions (save, print, bookmark and copy) are traced in the Category A and Category B search paths which is evident from the user action distribution graph in Figure 14.

Thus from the user feedbacks it is concluded that most of the search paths that are recommended by the personalized search system are relevant to the user’s context of search and it has also reduced the time spent in choosing the correct page that satisfies the user’s information need.

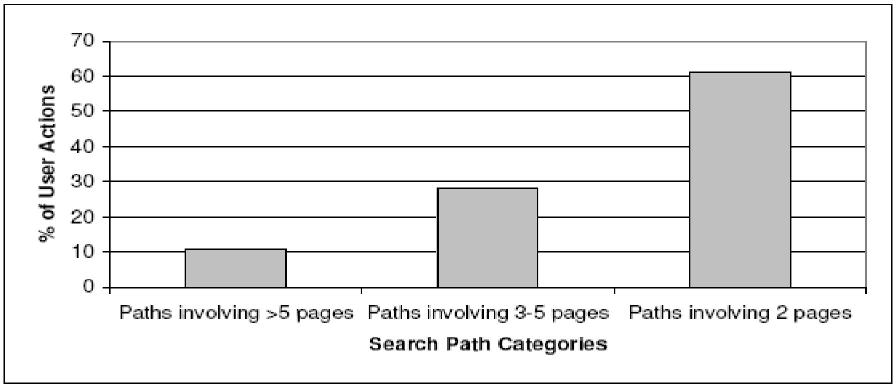


Figure 14. User Action Distribution in Search Paths

The construction of Personalized Page-View graph is automatic and hence does not require any human interference. Semantic search paths help the user to identify their information need more easily. Personalization using such semantic search paths can produce better results as compared to keyword-based searching by highlighting the users with various relevant search paths, thus, providing them a foresight of where they will end up if they choose a particular page.

5 CONCLUSIONS

To aid semantic search, this work proposes a graph based user profile called Personalized Page-View (PPV) graph that highlights shortest paths to (1) relevant pages that are missed by search engines and (2) relevant pages in the result set that go unnoticed by the user. Semantic based approaches are used to identify shortest paths that lead to relevant pages during a search. Content based link between relevant visited and relevant unvisited pages are established. Such links/relations highlighted those relevant pages that are unnoticed by the user. Thus the proposed personalized search accounted for relevant pages missed by the user as well as by the search engines.

The construction of Personalized Page-View graph is automatic and hence does not require any human interference. Semantic search paths help the user to identify their information need more effectively with appropriate visualizations as add-ons to the browser. Personalization using such semantic search paths can produce better results as compared to keyword-based searching by highlighting the users with the various relevant search paths, thus, providing them a foresight of where they will end up if they choose a particular page.

The user profile constructed highlights two important abstract information: 1) links between search query and relevant pages both from visited and unvisited page categories, and 2) shortest paths that lead to relevant information. However effective inference mechanisms must be used for inferring personalized information from such graph based user profiles. A graph based profile though found effective for producing concept based search results, it has increased the search time. Hence PPV graph based search system has made the final recommendation process time consuming. Also the graph based information highlights user's interests in terms of the web pages visited by the user during the various search sessions and hence does not highlight the users' interests directly. Performing similar search analysis for identifying users with similar interests also becomes difficult when user profile is modeled as a graph. In addition, current context of search can be reflected only by inferring the fluctuations in the user's interests on day to day basis which is another missing factor from the graph based user profile. Keyword based representation of user interests by using categorical labels will be more effective to analyze fluctuations in the user interest. Finally a clear distinction of user's long and short term interests is also missing in the PPV graph.

Classification of user's interest into long and short-term shall be improved by calculating the rate at which the user's interest on a topic decreases, i.e., by computing interest decay. Additionally, this idea shall be extended for improving community based web search by evolving interest based user groups and a group based conceptual index that provides a conceptual link between the user groups and the relevant pages. The PPV graphs proposed in this work shall be extended to automatic query refinement, by refining the query with appropriate user's short-term interest related keywords. The concept based approach for personalized web search explained in this work can be extended for a large scale personalized web search in existing search engines.

References

Allen C., Kania D., Yaeckel B. 2001, One-to-One Web Marketing: Build a Relationship Marketing Strategy One Customer at a Time, 2nd edition. John Wiley and Sons, New York.

Chinchor N. 1992, 'Evaluation Metrics', In Proceedings of the Fourth Message Understanding Conference, Princeton, New Jersey, pp. 22-29.

Cross V. and Pal A. 2005, 'Metrics for Ontologies', In Proceedings of Annual Meeting of North American Fuzzy Information Processing Society, pp. 448-453.

Eetu Makela 2005, 'Survey of Semantic Search Research', Proceedings of the Seminar on Knowledge Management on the Semantic Web, Department of Computer Science, University of Helsinki.

Gauch S., Chaffee J. and Pretschner A. 2003, 'Ontology-based personalized search and browsing', Web Intelligence and Agent Systems, Vol. 1, No. 3-4, pp. 219-234.

JGraph 2009, http://www.jgraph.com/, accessed latest by 2009.

Joachims T., Granka L., Pan B., Hembrooke H. and Gay G. 2005, 'Accurately Interpreting Clickthrough Data as Implicit Feedback', In Proceedings of the Conference on Research and Development in Information Retrieval (SIGIR), Salvador, Brazil, pp. 154-161.

Lawrence Steve and Giles Lee 2000, Accessibility of information on the Web, Intelligence, vol. 11, pp. 32-39.

Lora A., Paul D.B. and Vadim C. 2007, 'Semantic Web-based Adaptive Hypermedia', In Proceedings of 18th International Conference on HyperText and Hypermedia, Manchester, UK, pp. 81-84.

Lozano-Tello A., Gómez-Pérez A. and Sosa E. 2003, 'Selection of Ontologies for the Semantic Web', LNCS, Vol. 2722, pp. 343-350.

Oard D. and Kim J. 1998, 'Implicit Feedback for Recommender Systems', In Proceedings of the AAAI Workshop on Recommender Systems, Madison, WI, pp. 81-83.

ODP 2009, http://dmoz.org/, accessed latest by March 2009.

Peng W.C. and Lin Y.C. 2006, 'Ranking Web Search Results from Personalized Perspective', CEC/EEE 2006, *p. 12.*

Pretschner A. and Gauch S. 1999, Ontology Based Personalized Search, Proc. Eighth IEEE Int'l Conf. Tools with Artificial Intelligence (ICTAI), pp. 391-198.

Saravanakumar C., Sendhilkumar S. and Geetha T.V. 2008, 'Ranking of Semantic Search Paths Using Personalized Weights for Aiding Web Search', Journal of Research in Computing Science: Special Issue on Advances in Computer science and Engineering, Vol: 34, pp: 237-248.

Sendhilkumar S. and Geetha T.V. 2005, ‘Web Search Using Personalized User Conceptual Index’, In proceedings of the 2nd Indian International Conference on Artificial Intelligence (IICAI ‘05), Pune, India, pp: 1719 - 1728.

Sendhilkumar S. and Geetha T.V. 2007, ‘Personalized Web Search Using Enhanced Probabilistic User Conceptual Index’, Journal of Intelligent Systems, Vol. 17, No. 1-3, pp. 199-213.

Sendhilkumar S. and Geetha T.V. 2008a, ‘Personalized Ontology for Web Search Personalization’, in proceedings of the 1st ACM Bangalore annual Compute conference - *Compute 2008, Bangalore, India, January 18-20, ACM Digital Library, Article No. 18.*

Sendhilkumar S. and Geetha T.V. 2008b, ‘Ranking and Evaluation of Automatically Constructed Semantic Search Paths’, International Journal of Artificial Intelligence, ISSN 0974-0635: Special Issue on Theory and Applications of Soft Computing, Autumn 2008, Vol. 1, No. A08, pp: 133-148.

Sendhilkumar S. and Geetha T.V. 2009, ‘Architecture for Effective Personalized Web Search’, International Journal of Computer Applications in Technology: Special Issue on computer Applications in Knowledge-Based Systems, Vol. 35, No.2/3/4, pp. 219 - 233.

Sendhilkumar S. and Geetha T.V. 2009, ‘Personalized Web Search Using A Modified User Conceptual Index Based On A Search Flow Graph’, in proceedings of the Fourth Indian International Conference on Artificial Intelligence (IICAI ‘07), Tumkur, India pp:1598-1617, December 2009.

Shahabi C. and Yi-Shin Chen Y.S. 2003, Web information personalization: challenges and approaches, Third International Workshop on Databases in Networked Information Systems, pp. 5-15.

Shahabi C., Banaei-Kashani F., Chen Y. S., McLeod D. 2003, Yoda: An Accurate and Scalable Web-based Recommendation System. In Proc. of 6th Int.Conf. on Cooperative Information Systems, pp. 418-432.

Shahabi C., Kaghazian L., Mehta S., Ghoting A., Shanbhag G. and McLaughlin M.L. 2002, ‘Understanding of User Behavior in Immensive Environments ‘, In Touch in Virtual Environments: Haptics and the Design of Interactive Systems, McLaughlin M.L., Hespanha J. and Sukhatme G. (Eds.), All of University of Southern California Prentice Hall, ISBN 0-13-065097-8, pp. 239-259.

Shapira B., Maimon M. and Anny M. 2006, ‘Study of Effectiveness of Implicit Indicators and their Optimal Combination for Accurate Inference of Users Interests’, Journal of Digital Information Management, Vol. 4, No. 3, pp. 168-173.

Steffen Staab and Rudi Studer 2004, Handbook on Ontologies, International Handbooks on Information Systems, Springer.

Stuart E. Middleton, David C. De Roure and Nigel R. Shadbolt, (2001) Capturing knowledge of user preferences: Ontologies in recommender systems, Proceedings of the 1st international conference on Knowledge capture, pp. 100--107.

Trajkova J. and Gauch S. 2004, ‘Improving ontology-based user profiles’, In Proceedings of the Recherche d’Information Assist e par Ordinateur, pp. 380-389.

User OntoStudio Manual 2007, http://www.ontoprise.de/content/ e799/e893/e938/e954/e958/User_Manual_OntoStudio_2.0_eng.pdf, accessed latest by 2007.

WordNet 2006, 'WordNet - a lexical database for the English Language', WordNet 3.0 Reference Manual, Princeton University Press.

Yang Z., Zhang D. and Chuan Y.E. 2006, 'Evaluation Metrics for Ontology Complexity and Evolution Analysis', Proceedings of the IEEE International Conference on e-Business Engineering, Shanghai, China, pp. 162-170.