INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH & DEVELOPMENT

ISSN: 2278 – 0211 (Online)

A Novel Approach For CBPW: Concept Based Personalized Web Search

Rajarajeswari.S
Assistant professor, Department of E-commerce,
V.H.N.Senthikumara nadar college, Virudhunagar, tamil nadu, India

Abstract:

The amount of information obtainable at online is increasing exponentially. Several examined projects and organizations are exploring the use of personalized applications that manage this scenario by molding the information presented to individual user. This process is usually called user profiling. In existing systems spotlight is made on personalized filtering and rating systems for electronic message, electronic newspapers, and web document. This dissertation presents ontology based personalized ontological mechanism for generating user profiles. Topic Ontology association is the process that takes two ontology's and produces a set of semantic correspondences between the group of elements and other. The recent personalized user profile generation is focused on improving steering efficiency by providing browsing assistants and adaptive links.

Key words: *Ontological user profiles, Semantic relations.*

1.Introduction

Personalized systems contract with the overload problem by building; managing and representing information customized for individual users. This customization may take the form of filtering out irrelevant information and/or identifying additional information of likely interest of the user. Research into personalization is ongoing in the fields of information retrieval, artificial intelligence, and data mining among others.

User profiles specifically are designed for personalized information access. There is wide variety of application to which personalization can be applied and a wide variety of different devices available on which to deliver the personalized information. Early personalization is focused on personalized filter, evaluation system for electronic mail, electronic journalists, UseNet newsgroups and netting document.

Most personalization systems are based on some type of user profile, a data instance of a user model that is applied to adaptive interactive systems. User profiles might include demographic information, like name, age, country, education level, etc, represents the interests or preferences of either a group of users or a single person. Personalization of Web portals, for instance, may focus on individual users, displaying news about specifically chosen topics or the market summary of stocks selected particularly, or groups of users for whom distinctive characteristics are identified, displaying targeted advertising on e-commerce sites.

In order to construct an individual user's profile, information might be collected explicitly, through direct user intervention, or implicitly, through agents that monitor user activity.

Profiles that can be modified or augmented are considered dynamic, in contrast to static profiles that maintain the same information over time. Dynamic profiles that take time into consideration may differentiate between short-expression and long-expression interests. Short-expression profiles represent the user's current interests whereas long-expression profiles indicate interests that are not subject to frequent changes over time. For instance, consider a musician who uses the Web for her daily research. One day, she decides to go on vacation, and she uses the Web to look for hotels, airplane tickets, etc. Her user profile should reflect her music interests as long-expression interests, and the vacation-related interests as short-expression ones.

Likewise in the user-profiling phase, user's individual information is collected. The information are based on explicit input of the user or implicitly collected by the agents. Depending upon user information different data collected on client side or application server side. The system implicitly collecting data must be installed specific software and/or explicitly feedback to be collected.

The user-profiling has some methods to collect information. The hierarchy of relationship among the data will be collected based on explicit information collection and implicit information collection.

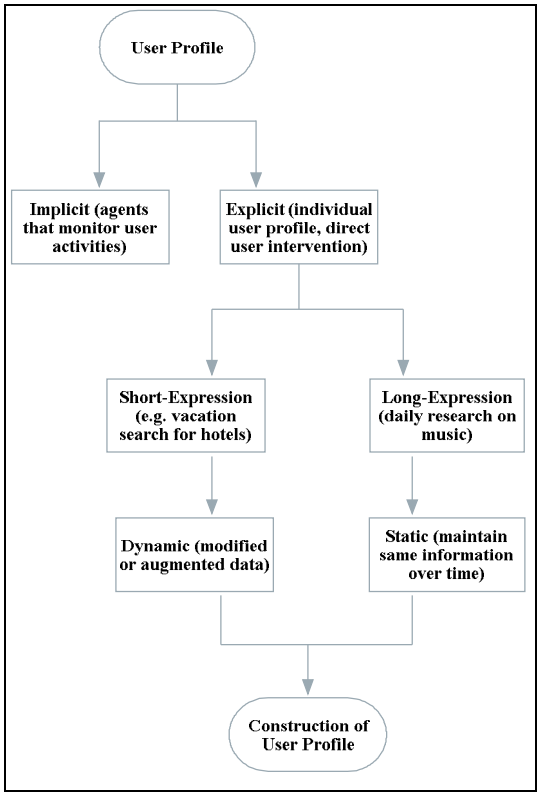

Figure 1: System for generating user profile

1.1.Unambiguous Data Aggregation

The explicit user information collection methodologies are called user feedback. User personalized information are collected via HTML pages. The demographic information is collected from user as input. Most complicated personalization projects based on explicit feedback have focused on navigation. One problem with explicit feedback is that it places an additional load on the user. Because of this, or privacy concerns, the user may not choose

to participate. Users may not accurately report their own interests or demographic data; because the profile remains static moreover the user's interests may change over time, in which the profile may become increasingly inaccurate over time.

1.2.Implicit User Information Collection

User profiles are often constructed based on implicitly collected information, often called implicit user feedback. The prime merits of this technique are, it does not require any extra intercession by the user during the process of constructing profiles. The type of information for each mechanism is able to collect, the wideness of applicability of the collected information. Because they only require a onetime setup, do not require new software to be developed and installed on the user's desktop and only track browsing activity, proxy servers seem to be a good compromise between easily capturing information and yet not placing a large burden on the user. Capturing activity at the site providing personalized services, for instance a search site itself, is also an option in some cases. It requires no special user activity, but not all personalized sites are used frequently enough by any single user to allow them to create a useful profile. We discuss some extensions of personalized search results in Section 2. In Section 3, we demonstrate the effectiveness and efficiency of the proposed techniques. A survey of related works appears in Section 4 concluding the paper.

2.Ontology Based Personalized Search Results

Ontology based personalized mechanism is an ontological representation of the topic of dissertation where user interests are defined. The ontological split takes the shape of a Semantic Relation of interrelated Topic concepts and the user profiles are initially described as weighted lists of those concepts.

2.1.Topic Ontology Construction And Semantic Relations

2.1.1.Semantic Relations Based On User Profiles

In distinction to other mechanisms in personalized content retrieval, our mechanism formulates the use of explicit user profiles (as opposed to e.g. sets of preferred documents).

Functioning within an ontology-based personalization framework [1], Let user preferences are represented as vectors $u_i = (u_i^1, u_i, ^2 ... u_i ^{N)}$ where the weight u_i, j [0, 1] measures the intensity of the interest of user i for concept c_j in the Topic ontology, N being the total number of concepts in the ontology. Similarly, the objects d_k in the retrieval space are assumed to be describe by vectors $(d_k^1, d_k^2 ... d_k^N)$ of concept weights, in the same vector-space as user preferences. Based on this common logical representation, measures of user interest for content items can be computed by comparing preference and footnote vectors, and consequently these measures can be utilized for prioritizing, filtering and sorting contents (a collection, a catalog, a search result) in a personal way. The ontology-based representation is richer and less ambiguous than a keyword based or item-based model. It provides an adequate preparation for the representation of course to fine-grained user interests, and can be a key enabler to deal with the subtlety of user preferences. Ontology provides further recognized, computer progression meaning on the concepts, and makes it available for the personalization system to take advantage off. Furthermore, ontology standards, such as RDF and OWL, support inference mechanisms that can be utilized to enhance personalization. Eg. a user interested in homes (super class of furniture) is also recommended items about furniture. Inversely, a user interested in dressing table and cot can be inferred to be interested in furniture's. These characteristics will be exploited in our personalized retrieval model. In factual scenarios, user profiles tend to be very scattered, especially in those applications where user profiles have to be manually defined. Users are habitually not willing to spend time describing their detailed preferences to the system, even less to assign weights to them, especially if they do not have a clear understanding of the effects and results of this input. On the other hand, applications where an automatic preference learning algorithm is applied tend to recognize the main characteristics of user preferences, thus yielding profiles that may entail a lack of expressivity. To overcome this problem, we propose a semantic preference distribution mechanism, which expands the initial set of preferences stored in user profiles through explicit semantic relations with other concepts in the ontology. Our mechanism is based on the Constrained Distribution Trigger (CSA) strategy [2, 3]. The expansion is self-controlled by applying a decompose factor to intensity the preference when each time a relation is traversed. Thus, the system outputs sorted lists of content items taking into account not only the preferences of the current user,

but also a semantic distribution mechanism through the user profile and the Topic ontology. In fact, previous experiments are carried without the semantic distribution process and very poor results are obtained. Moreover the profiles are also very simple and the matching between the preferences of different users is low.

2.2.Topic Concepts

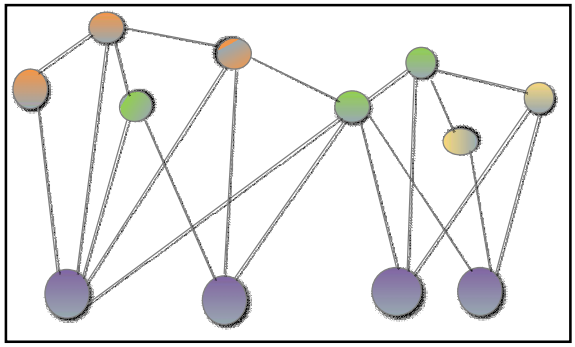

Figure 2: Semantic user preferences and individual interests.

User's interest

2.3.Topic concepts

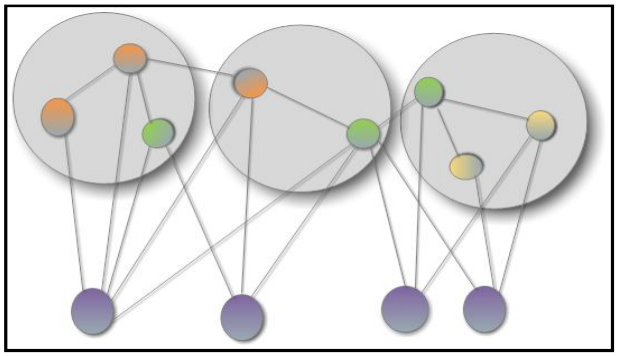

Figure 3: Semantic topic concepts relations and clusters, based on the user interest.

User's interest

2.4.Topic concepts

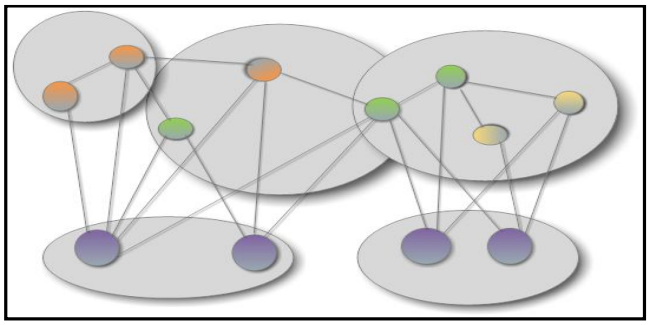

Figure 4: Users are clustered in order to identify the closest relations to each user

User's interest

This surveillance shows a better performance when using ontology-based profiles, instead of classical keyword-based preferences representations. We have conducted several experiments showing that the performance of the personalization system is considerably poorer when the distribution mechanism is not enabled. However, the basic things in the profile are quite much uncomplicated.

A perfect representative sample of user preferences are provided, but the actual user desires are not explored, therefore results is tending low overlaps between the preferences of different users. Therefore, the wings are not considered as significant for the performance of user personalization, but pave an effective role in the clustering strategy.

2.5.Topic Ontology's Based On Hierarchal Relations

Topic ontology's are means of categorizing Web pages based on their content. In these ontology's, topics are typically organized in a hierarchical scheme in such a way that more specific topics are part of more general ones. In addition, it is possible to include cross-references to link different topics in a non-hierarchical scheme. The open directory project ontology is the largest human-edited directory of the Web. It categorizes millions of pages into a topical ontology combining a hierarchical and non-hierarchical scheme. This topical directory can be used to measure Semantic Affiliations among massive numbers of pairs of Web pages or topics.

Several measures have been developed to approximate Semantic Affiliation in a set of connections representation. Early proposals have used for evaluating path distances between the nodes in the system. These frameworks are based on the principle that the stronger the

Semantic Affiliation of two objects, the earlier they will be in the system representation. However, there are number of sources which are examined, and issues are raised when attempting to apply distance-based schemes for the purpose of predicting object similarities in some network classes where links may not represent uniform distances.

2.6.Semantic Affiliation

Ontology association is the function that takes ontology's and produces a set of semantic correspondences between some elements of the other. The ontology association problem has an important background work in discrete mathematics for matching graphs [4] [5], in databases for mapping schemas [6] and in machine learning for clustering structured objects [7]. Most part of ontology association algorithms are just concentrating on finding close entities (the"="relationship), that rely on some *Semantic Affiliation* measure.

The information content of a class is measured by the negative log likelihood, −log Pr[t], where Pr[t] represents the prior probability that any object is classified under topic t. In practice Pr[t] can be computed for every topic t in taxonomy by counting the fraction of objects stored in the sub tree rooted at t (i.e., substance stored in node t and its offspring) out of all the objects in the taxonomy.

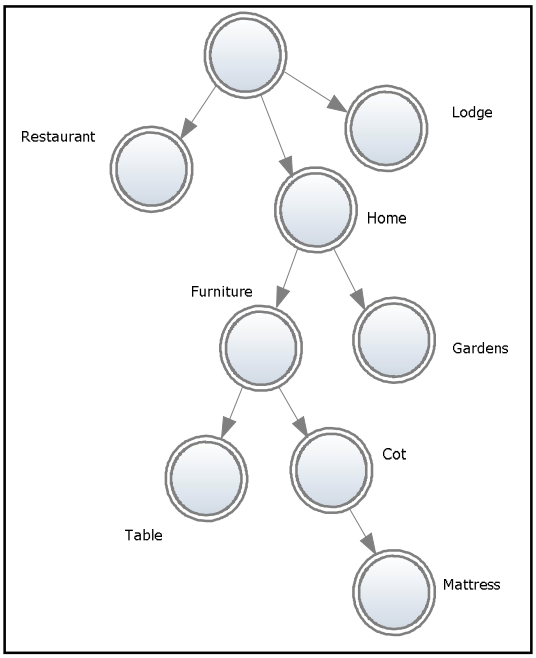

Figure 5: Topic relation.

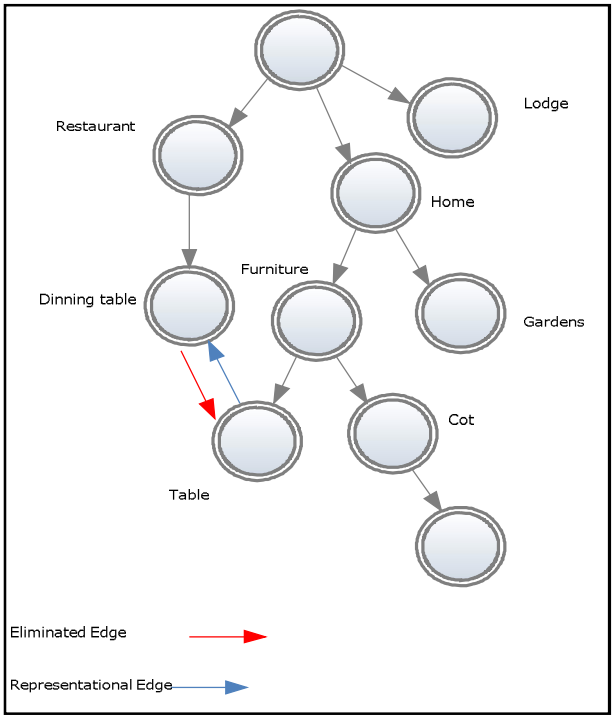

Figure 6: Topic relationship.

According to Lin's proposal, the Semantic Affiliation [7] between two topics t1 and t2 in taxonomy is measured as the ratio between the implications of lowest common ancestor and their individual meanings.

$\sigma Ts(t1,t2)=2.\log Pr[t0(t1,t2)]/\log Pr[t1]+\log Pr[t2]$

Where t0 (t1, t2) is the lowest common ancestor topic for t1 and t2 in the tree. Given a document d classified in a topic classification, we use topic (d) to refer to the topic node containing d. Given two documents d1 and d2 in topic taxonomy the Semantic Affiliation between them is estimated as $\sigma^T s$ (topic (d1), topic (d2)). In order to simplify notation, we use $\sigma^T s$ (d1, d2) as shorthand for $\sigma^T s$ (topic (d1), topic (d2)).

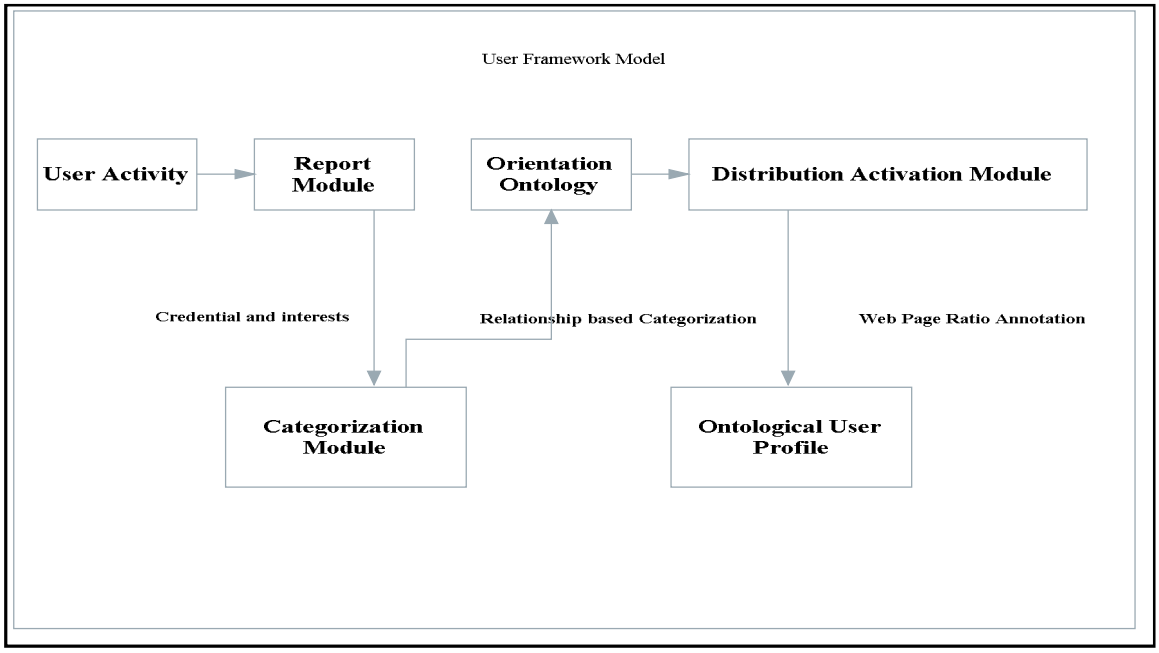

Figure 7: Overall Framework for ontological User Profile.

3.Proposed Mechanism

3.1.Personalization Based On Ontology

Our objective is to utilize the user perspective to personalized search results by re-sorting the results returned from a search engine for a given inquiry. Our unified context model for a user is represented as an instance of a reference Topic ontology in which concepts are annotated by interest ratios derived and updates implicitly based on the user's information access behavior. We express this representation as an ontological user profile.

Ontology is a specification of a conceptualization-description of the concepts and relationships that can exist for an agent/user or a community of agents/users. One increasingly popular method to mediate information access is through the use of ontology's. Researchers have been attempting to utilize ontology's for improving navigation effectiveness as well as personalization we search and browse, specifically in the combination with the notion of automatically generated semantically enriched ontology-based user profiles.

3.2.User Activity

The relationship is calculated between the web pages visited by a user and the concepts in topic ontology. A User Profile with non-zero weights is created after completing the annotation process of each concept with a weight based on an accumulated relationship ratio. In advance, the main purpose of including ontology in the proposed work is to identify the topics that might be of interest to a specific web user. Hence, we define the ontology as a topics hierarchy, where the topics are utilized for the categorization and group of web pages. The hierarchal relationships among the concepts are taken into consideration for building the ontological user profile updated the annotations for existing concepts using distribution trigger.

In the proposed framework, the context of the user is defined with an ontological user profile. This User Profile is an annotated instance of reference ontology. Figure 7 shows depict a high level picture of our proposed framework model based on an ontological user profile. When disambiguating the context, the topic knowledge inherent in existing reference ontology is called as a source of key topic concepts.

Initially, every ontological user profile is reference ontology's instance. Each concept in the user profile is annotated with an interest ratio which has an initial value of one. As the user interacts with the system by selecting of new documents to view it, updates are made to the ontological user profile and the annotations for existing concepts are modified by distribution trigger. Thus, the user context is maintained and updated incrementally based on user's ongoing behavior.

Precise information about the users' interests must be collected and represented with minimal user interference. This can be done by inactively observing the users browsing behavior overtime and collection web pages in which the user has shown interest. Numerous factors, including the frequency of visits to a page, the total amount of time spent on that particular page, and bookmarking a page can be used as bases for heuristics to automatically collect these documents.

3.3.Orientation Module

We utilize the web pages as training data for the representation of the concepts in the orientation ontology. The information in the form of texts that can get extracted from web

pages explains the semantics of the concepts and is learned as we build an expression vector representation for the concepts. We create a cumulative demonstration of the orientation ontology by computing an expression vector $\vec{r}$ for each concept r in the concept hierarchy. Each concept vector represents, all individual training documents indexed under that concept, as well as all of its sub concepts.

We begin by constructing a global dictionary of expressions extracted from the training documents indexed under each concept. A stop list is used to remove high frequency, but semantically non-relevant expressions from the content. Porter stemming is utilized to reduce words to their stems. Let d be the document in the training data that is represented as an expression vector $\vec{d} = (w1, w2, ..., wk)$, where each expression weight, w_i is computed using expression frequency and inverse document frequency. Specifically, $w_i = tf_i * \log (R/r_i)$, where tf_i is the frequency of expression i in document d, R is the total number of documents in the training set, and r_i is the number of documents that contains expression i. We further normalize each document vector, so that $\vec{d}$ represents an expression vector with unit length.

The aggregate representation of the concept hierarchy can be described more detailed as follows. Let S (r) is the se t of sub concepts under concept n as non-leaf nodes. Also, let $\{d_1^r, d_2^r \ldots d_{kr}^{r}\}$ be the individual documents indexed under concept n as leaf nodes. Docs (r), which includes of all of the documents that are indexed under the concept of n along with all of the documents that are indexed under all of the sub concepts of n is defined as:

Docs (r) = [U $r' \in$ S (r) DOCS (r')] U $\{d_1^r, d_2^r \ldots d_{kr}^{r}\}$

The concept expression vector $\vec{n}$ is then computed as:

$$\vec{r} = \left[\sum_{d \in docs(r)} \vec{d} \right] / \left| \text{Docs (r)} \right|$$

Thus, $\vec{r}$ represents the centroid of the documents indexed under concept n along with the sub concepts of n. The resulting expression vector I normalized into a unit expression vector.

3.4.Framework Model

Each node in the ontological user profile is a pair, $(C_j, IS(C_j))$, where C_j is a concept in the reference ontology and $IS(C_j)$ is the interest ratio annotation for that concept. The input expression vector represents the active interaction of the user query or his surprising activities. Based on the user's information access behavior, let's assume the user has shown interest in Fanatic, classical.

Since the input expression vector contains expressions that appear in the expression vector for the Fanatic concept, as a result of Distribution trigger, the interest ratios for the Fanatic, classical, styles, and music concepts get decreased. The Distribution Trigger algorithm and the process of updating the interest ratios are discussed in detail in the next section.

3.5.Erudition Profiles By Distribution Trigger

We use Distribution Trigger to incrementally update the interest ratio of the concepts in the user profiles. Consequently, the ontological based user profile is subjected to be the semantic relations and the interest ratio is updated based on trigger values.

In our mechanism, we use a very specific configuration of distribution trigger, depicted in Algorithm 1, for the individual purpose of maintaining interest ratios within a user profile. We assume a model of user behavior learned through the passive observation of user's information access activity and web pages in which the user has shown interest in that can be automatically collected for user profiling.

The algorithm has an initial set of concepts from the assigned initial trigger value. The main idea is to activate other concepts following a set of weighted relation during propagation and at the end obtain a set of concepts and their respective triggers.

The source and destination concepts play an important role in trigger. Since the nodes are organized into a concept hierarchy derived from the topic ontology, we compute the weights for the relations between each concept and all of its sub concepts using a measure of containment. The containment weight produces a range of values between $0 \leq x \leq 1$, whereas a value of one indicates complete overlap.

The weight of the relation W_{is} for concept i and one of its sub concepts s is computed as

$W_{is} = \vec{n}_i . \vec{n}_s / \vec{n}_i . \vec{n}_i$, where $\vec{n}_i$

is the expression vector concept i and $\vec{n}_s$ is the expression vector for sub concepts s. Once the weights are computed we process the weights again to ensure the total sum of the weights of the relations between a concept and all sub concepts equal to 1.

- Input: Ontological user profile with interest ratios and a set of documents
- Output: Ontological user profile concepts with updated trigger values

CON = {C_1... C_n}, concepts with interest ratios
IS (C_j), interest ratio
IS (C_j) = 1, no interest information available
I = {d_1... d_n}, user is interested in these documents
For each $d_i \in$ I do
Initialize stack;
For each $C_j \in$ CON do
C_j .Trigger = 0; // Reset trigger value
End
For each $C_j \in$ CON do
Calculate sim (d_i, C_j);
If sim (d_i, C_j) > 0 then
C_j.Trigger = IS (C_j) _ sim (d_i, C_j);
Stack. Add (C_j);
Else
C_j .Trigger = 0;
End
End
While Stack. Count > 0 do
Sort stack; // trigger values (descending)
Cs = Stack [0]; // first item (distribution concept)
Stack.Dequeue (Cs); // remove item
If pass Restrictions (Cs) then
Linked Concepts = GetCorrelatedconcepts (Cs);
For each C_l in linkedConcepts do

C_l.Trigger+ =C_s.Trigger _ Cl.Weight;

Stack. Add (C_l);

End

End

End

End

Algorithm 1: Distribution Trigger Algorithm.

The algorithm considers each of the documents assumed to represent the current framework. For each iteration of the algorithm, the initial trigger value of every concept in the user profile is reset to zero. We compute an expression vector for each document d_i and compare the expression vector for d_i with the expression vectors for each concept C_i in the user profile using a cosine relationship measure. Those concepts with a relationship ratio, sim (d_i, C_j), greater than zero are added in a stack, which is in a non-increasing order with respect to the concepts trigger values. The trigger value for concept C_j is assigned to IS (C_j) *sim (d_i, C_j), where IS (C_j) is the existing interest ratio for the specific concept. The concept with the highest trigger value is then removed from the stack and processed.

If the current concept passes through restrictions, it propagates its trigger to its neighbors. The amount of trigger that is propagated to each neighbor is proportional to the weight of the relation. The neighboring concepts which are activated and are not currently in the priority queue are added to queue, which is then reordered. The process repeats itself until there are no further concepts to be processed in the stack.

The neighbors for the distribution concept are considered to be the linked concepts. For a given distribution concept, we can ensure that the algorithm processes each edge only once by iterating over the linked concepts only one time. The order of the iteration over the linked concepts does not affect the results of trigger. The linked concepts that are activated are added to the existing stack, which is then sorted with respect to trigger values. The interest ratio for each concept in the ontological user profile is then updated using Algorithm 2. The interest ratios for all a concept are then treated as a vector, which is normalized to a unit length using a pre-defined constant, k, as the length of the vector. Rather than steadily

increasing the interest ratios, we utilize normalization so that the interest ratios can get decremented as well as gets incremented. The concepts in the ontological user profile are updated with the normalized interest ratios.

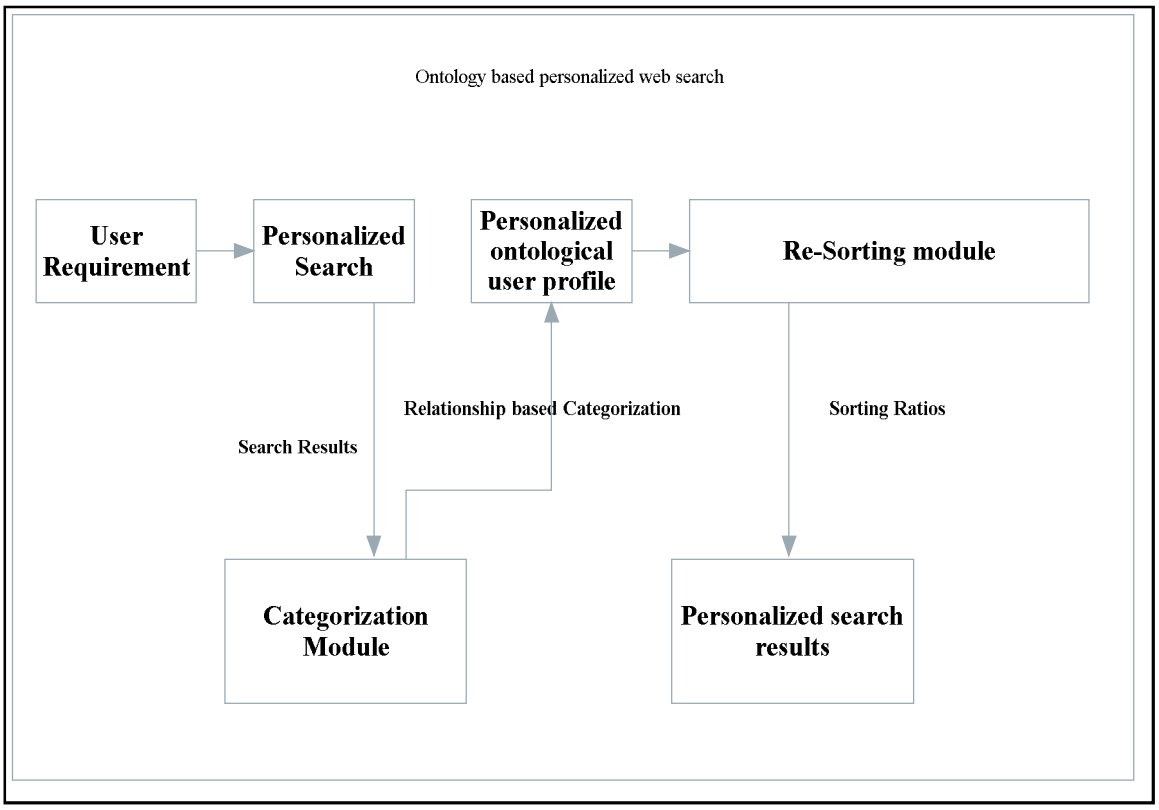

Figure 8: Ontological User Profile for personalized search results

3.6.Personalized Search

Our objective is to adopt the user framework to personalize search results by re-sorting the results returned from a search engine for a given query. Figure 3 displays our mechanism for search personalization based on ontological user profiles. Superciliously ontological user profile with interest ratios exists and we have a set of search results, Algorithm 3 is utilized to re-sort the search results based on the interest ratios and the semantic facts in the user profile.

- Input: Ontological user profile concepts with updated trigger values
- Output: Ontological user profile concepts with updated interest ratios

CON = {C_1... C_n}, concepts with interest ratios

IS (C_j), interest ratio

C_j .Trigger, trigger value resulting from Distribution Trigger

K, constant

n = 0;

For each $C_j \in$ CON do

IS (C_j) = IS (C_j) + C_j.Trigger;

n = n + (IS (C_j)) ^{2}; // sum of squared interest ratios

n = p_n; // square root of sum of squared interest ratios

End

For each $C_j \in$ CON do

IS (C_j) = (IS (C_j) _ k)/n; // normalize to constant length

End

Algorithm 2: Normalization and Updating of Interest Ratios in the Ontological User Profile.

An expression vector $\vec{r}$ is computed for each document $r \in R$, where R is the set of search results for a given query. The expression weights are obtained using the tf.idf formula described earlier. To calculate the sort ratio for each document, first the relationship of the document and the inquiry is computed using a cosine relationship measure. To identify the best matching concept, we compute the relationship of the document with every concept in the user profile. Once the best matching concept is identified, a sort ratio is assigned to the document by multiplying the interest ratio for the concept, the relationship of the document to the query, and the relationship of the specific concept to the query. If the interest ratio for the best matching concepts is greater than one, it is then boosted by a tuning parameter α. Once all documents have been processed, the search results are sorted in descending order with respect to this new sort ratio.

- Input: Ontological user profile with interest ratios and a set of search results.
- Output: Re-sorted search results.

CON = {C_1... C_n}, concepts with interest ratios

IS (C_j), interest ratio

R = {d_1... d_n}, search results from query q

For each $d_i \in$ R do

Calculate sim (d_i, q);

MaxSim = 0;

For each $C_j \in$ CON do

Calculate sim (d_i, C_j);

If sim (d, C_j) $\geq$ maxSim then

(Concept) c = C_j;

MaxSim = sim (d_i, C_j);

End

End

Calculate sim (q, c);

If IS(c) > 1 then

SortRatio (d_i) = IS (c) * α * sim (d_i, q) * sim (q, c);

Else

SortRatio (d_i) = IS(c) * sim (d_i, q)* sim (q, c);

End

End

Sort R based on sortRatio;

Algorithm 3: Re-sorting Algorithm.

4.Experimental Results

Since the queries of average web users tend to be tiny and indefinite. Our objective is to exhibit that re-sorting based on ontological user profiles can help in disambiguating the user's intent particularly when such queries are used. We measure the effectiveness of re-sorting in terms of Topic-n recall and Topic-n Precision.

4.1.Evaluation And Experimental Results On Data Sets

For experimental purposes, we decided to use a branching factor of three with a depth of ten levels in the hierarchy. Our experimental data set contained 506 concepts in the hierarchy and a total of 8857 documents that were indexed under various concepts. We processed the indexed documents into three separate sets including a training set, a test set, and a profile set.

For each concept, we used 60 percent of the associated documents for the training set, 20 percent for the test set, and the remaining 20 percent for the profile set. For all of the data sets, we kept track of which concepts these documents were originally indexed under in the hierarchy. The training set was utilized for the representation of the reference ontology, the profile set was used for distribution trigger, and the test set was utilized as the document collection for searching.

The training set consisted of 5157 documents which were used for the one-time learning of the reference ontology. The concept expressions and corresponding expression weights were computed using the formula described in the Representation of Reference Ontology section.

Query	**Terms**	**Criteria**
Set 1	1	Highest weighting term in concept term vector
Set 2	2	Two highest weighting term in concept term vector
Set 3	3	Three highest weighting term in concept term vector
Set 4	2 or more	Overlapping terms within highest weighting 10 terms

Table 1: Set of Keyword queries.

A total of 1675 documents were included in the test set, which were used as the document collection for performing our search experiments. Depending on the search query, each document in our collection can be treated as a signal or a noise document. The signal documents are those documents relevant to a particular concept that should be ranked high in the search results for queries related to that concept. The noise documents are those documents that should be ranked low or excluded from the search results.

The test set documents that were originally indexed under a specific concept and all of its sub concepts were treated as signal documents for that concept whereas all other test set documents were treated as noise. In order to create an index for the signal and noise documents, a tf.idf weight was computed for each term in the document collection using the global dictionary of the reference ontology.

The profile set consisted of 2000 documents, which were treated as are presentation of specific user interest for given concept to stimulate ontological user profiles. As we performed the automated experiments for each concept/query only the profile documents that were originally indexed under that specific concept were utilized to build an ontological user profile by updating the interest ratios with the distribution trigger algorithm.

We constructed keyword queries to run our automated experiments. We decided to extract the query terms from the concept term vectors in the ontology. Each concept term vector was sorted in descending order with respect to term weights. Table 1 depicts the four query sets that were automatically generated for evaluation purposes. Our keyword queries were used to run a number of automated search scenarios for each concept in our reference ontology. The first set of keyword queries contained only one term and included the highest weighting term for each concept. In order to evaluate the search results when a single keyword was typed by the user as the search query, the assumption was that the user was interested in the given concept.

The second set of queries contained two terms including the two highest weighting terms for each concept. The third set of queries was generated using the three highest weighting terms for each concept. As the number of keywords in a query increase, the search query becomes less ambiguous. Even though one to two keyword queries tend to be vague, we intentionally came up with a fourth query set to focus specifically on ambiguous queries. We generated this query set by computing the overlapping terms using the highest weighting ten terms in each concept term vector. Only the overlapping concepts were included in the experimental set with each query

consisting of two or more overlapping terms within these concepts. Our evaluation methodology was as follows. We used the system for performing a standard search for each query. As mentioned above, each query was designed for running our experiments for a specific concept. In the case of standard search, a term vector was built using the original keyword in the query text. Removal of stop words and stemming was utilized. Each term in the original query was assigned a weight of 1.0. The results of the search process were retrieved from the test set, then the signal documents and noise document collection, by using the cosine similarity measure for matching. Using an interval of ten, we calculated the Topic-n Recall and Topic-n Precision for the search results.

Starting with the top one hundred results and going down to top ten search results, the values for n included n={100,90,80,70, …… ,10}. The Topic-n Recall was computed by dividing the number of signal documents that appeared within the Topic n search results at each interval with the total number of signal documents for the given concept. We also computed the Topic –n precision at each interval by dividing the number of signal documents that appeared within the Topic n results with n. For instance, at n=100, the top 100 search results were included in the computation of recall and precision, whereas at n=90, only the top 90 results were taken into consideration. Subsequently, the documents encountered in the profile set are utilized to simulate user interest for the specific concept. For each query, we started with a new instance of the ontological user profile with all interest ratio initialized to one. Such a user profile represents a situation where no initial user interest information is available. We performed our distribution trigger algorithm to update interest ratios in the ontological user profile. Following to build the ontological user profile, the original search results is sorted based on our re-sorting algorithm and computed the Topic-n Recall and Topic-n Precision with the personalized results.

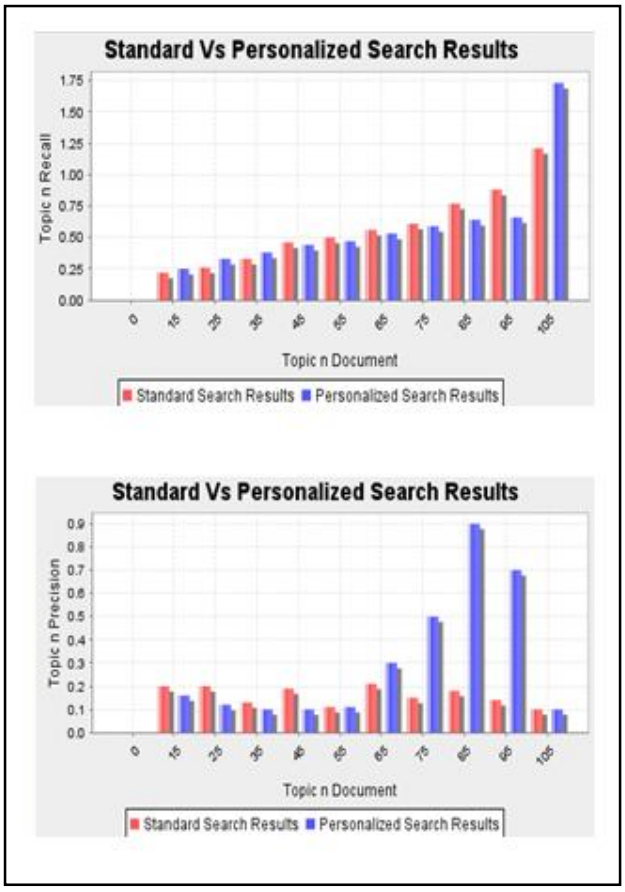

Figure 9: Average Topic-n Recall and Topic-n Precision comparisons between the personalized search and standard search using overlap queries.

In order to compare the standard search results with the personalized search results, we computed the average Topic-n Recall and Topic-n Precision, depicted in Figure 9. We have also computed the percentage of improvement between standard and personalized search for Topic-n Recall and Topic-n Precision, depicted in Figure 10.

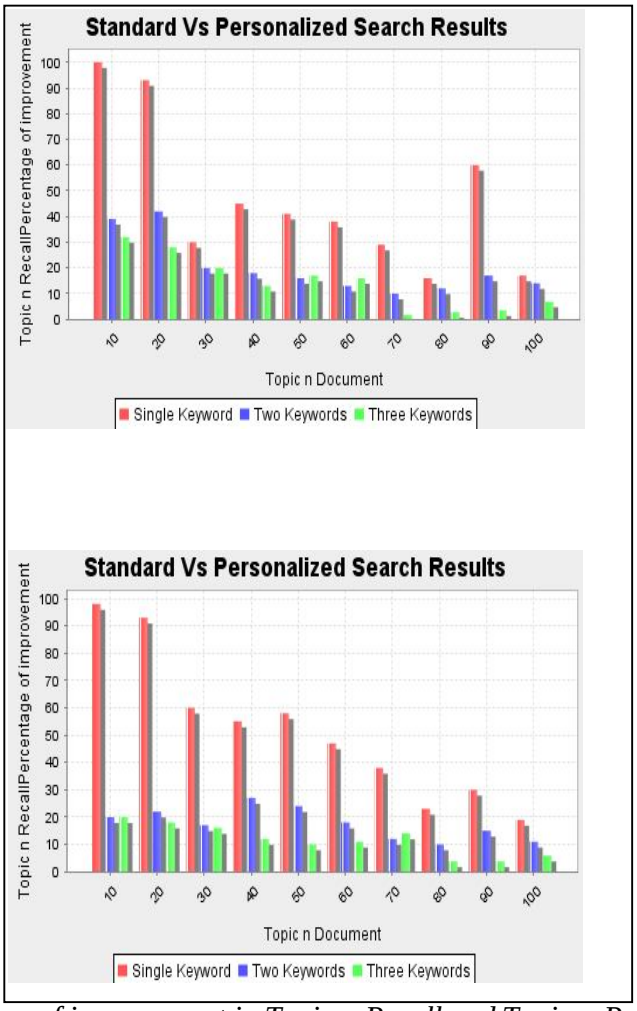

Figure 10: Percentage of improvement in Topic-n Recall and Topic-n Precision achieved by personalized search relative to standard search with various query sizes.

Every user has a unique goal and background while searching for information through entering key word queries into a search engine. The user queries are typically ambiguous and contain between one to three keywords. The search results that are turned from the search engine may satisfy the search criteria but often fail to meet the users search intention. Personalized search provides the user with results that accurately satisfy their specific goal and intent for the search. The queries used in our experiments are intentionally designed to be short to demonstrate the effectiveness of our web search personalization mechanism, especially in the typical case of web users who tent to use very short queries.

Simulating user behavior allowed us to run automated experiments with a larger dataset. In the worst case scenario, the user would enter only a single keyword. The evaluation results show significant improvement in recall and precision for single keyword queries

as well as gradual enhancement for two expression and three-expression queries. As the number of keywords in a query increase, the search query becomes clearer. In order to the one, two, and three keyword queries, we ran experiments with the overlap query set to focus on ambiguous queries. Two users may use the exact same keyword to express their search interest even though each user has a completely distinct intent for the search. For example, the keyword Fine may refer to Fine as a health as well as the Fine as a punishment sense. The purpose of the overlap queries is to simulate real user behavior where the user enters a vague keyword query as the search criteria. Our evaluation results verify the using the ontological user profile for personalizing search results is an effective mechanism. Especially with the overlap queries, our evaluation results confirm that the ambiguous query expressions that are disambiguated by the semantic evidence in the ontological user profiles.

5.Conclusion

We have presented a framework for appropriate information access using ontology's that established the semantic knowledge entrenched in an ontology combined with long-expression user profiles that to effectively tailor search results based on users interests and preferences. In our future work we plan to evaluate the stability and convergence properties of the ontological profiles as interest ratios are updated consequently which the system.

We sketch to design experiments to determine when a user profile becomes established and starts accurately representing user interests. Every time a new web page, which the user has shown interest in, is processed via distribution trigger, the interest ratio for the concepts in the ontological user profile are updated. Initially, the interest ratio for the concepts in the profile will continue to change. However, once enough information has been processed for profiling, the amount of change in interest ratios should decrease. Our expectation is that eventually the concepts with the highest interest ratios should become relatively established. Therefore, these concepts will reflect the user's primary interests. Since we focus on implicit methods for constructing the user profiles, the profiles need to adapt over time. Our future work will also involve designing experiments that will allow us to monitor user profiles over time to ensure the incremental updates to the interest ratios accurately reflect changes in user interests.

6.Reference

1. Vallet, D., Mylonas, P., Corella, M. A., Future enhancement, J. M., Castells, P., Avrithis, and Y.: A Semantically-Enhanced Personalization Framework for knowledge-Driven Media Services. IADIS WWW/Internet Conference. Lisbon, Portugal, October 2005.
2. Cohen, P. R. and Kjeldsen, R.: Information Retrieval by Constrained Distribution Trigger in Semantic Networks. Information Processing and Management 23(2), pp. 255-268, 1987.
3. Crestani, F., Lee, P. L.: Searching the web by constrained distribution trigger. Information Processing & Management 36(4), pp. 585-605, 2000.
4. E. Rahm and P. A. Bernstein. A survey of mechanisms to automatic schema matching. VLDB Journal: Very Large Data Bases, 10(4):334–350, 2001. Cite seer. ist.psu.edu/rahm01survey.html.
5. J. E. Hop croft and R. M. Karp, An $O(\frac{n5}{2})$ algorithm for maximum matching in bipartite graphs. SIAM J. Comput., 4:225-231, 1973.
6. C. H. Papadimitriou and K. Stieglitz. Combinatorial optimization: algorithms and Complexity. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1982.
7. M. Bisson. Learning in following with a relationship measure. In Proceedings of the 10th American Association for Artificial Intelligence conference, San-Jose (CA US), pages 8287, 1992.
8. Alani, H.; O'Hara, K.; and Shad bolt, N. 2002. Onto-copi: Methods and tools for identifying communities of practice. In Proceedings of the IFIP 17th World Computer Congress - TC12 Stream on Intelligent Information Processing, 225–236.
9. Trajkova, J., and Gauch, S. 2004. Improving ontology-based user profiles. In Proceedings of the Recherché d'Information Assiste par Ordinateur, RIAO 2004, 380–389.
10. Singh, A. and Nakata, K. 2005. Hierarchical classification of web search results using personalized ontology's.
11. Salton, G and Buckley, C. On the use of distribution trigger methods in mechanical in turn. In Proceedings of the 11th annual international ACM SIGIR conference on Research and Development in Information Retrieval, SIGIR 1988, 147–160. Salton, G., and McGill. 1983.

12. Aktas, M Nacar, M and Menczer, F. 2004. Using hyperlink features to personalize web search. In Proceedings of the 6th International Workshop on Knowledge Discovery from the Web, WebKDD 2004.
13. J. H. Lee, M. H. Kim, and Y. J. Lee. Information retrieval based on conceptual distance in is-a hierarchy. Journal of Documentation, 49(2):188–207, 1993.
14. C. H. Papadimitriou and K. Stieglitz Combinatorial optimization: algorithms and Complexity. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1982.