CHAPTER 6

SEMANTIC SEARCH PROCESS

6.1 INTRODUCTION

Automated information retrieval through search engines has become a preferred choice to know about anything in any language in any part of the globe. Because of that the corporate sector, academicians have already begun to host their information in the form of web documents in the web aiming to reach to users across boundaries. Different interests of the users searching the web for different information is not satisfied by the search engines due to lack of context in the search process. This chapter describes the characteristics of the proposed search process, its architecture and the design issues. It also reports on the improvement in the relevance of the search results.

6.2 PROCESS DESIGN CHARACTERISTICS

The results of search engines lack precision because they don't include context of the document in the search process. Semantic search process proposed in the thesis aims to provide context and category meta data of the web document in the search process itself. The design characteristic of the semantic search process is discussed in this section.
6.2.1 Semantic entity headers using HTTP extension framework

HTTP request header fields allow the clients to pass additional information about the request and about the client itself, to the server. These fields act as request modifiers, with semantics equivalent to the parameters on a programming language method invocation. Entity headers fields define meta information about the entity body or if no body present, about the resource identified by the request. The extension header mechanism allows additional entity header fields to be defined without changing the protocol, but these fields cannot be assumed be recognizable by the recipient.

Semantic search process introduces two additional entity headers fields to define the context meta information about the entity body using the mechanisms provided by the HTTP extension framework (rfc 2774). HTTP extension framework is designed to accommodate dynamic extension of HTTP clients and servers by software components. RFC 2774 (Nielson et al, 2000) proposal introduces two types of extension declaration strength - mandatory, optional and two types of extension declaration scope - hop-by-hop and end-to-end.

Semantic entity headers are created using mandatory extension declaration that indicates that the ultimate recipient must consult and adhere to the rules given by the extension when processing the message o reporting an error. It also uses the end-to-end declaration, which is transmitted to the ultimate recipient of the declaration. The two semantic entity headers introduced include content category and content context. The content
category entity header defines the category of the entity body the HTTP request carries while the content context defines the context of the entity body. The web page authors provide the values to the headers. Sample requests and responses are given in Fig 6.1

<table>
<thead>
<tr>
<th>M_GET /request.htm HTTP/1.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Man: ContentCategory ; ns =16</td>
</tr>
<tr>
<td>16- ContentCategory: ComputerScience</td>
</tr>
<tr>
<td>Host: <a href="http://www.sample.com">www.sample.com</a></td>
</tr>
<tr>
<td>Contentlength: 1203</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>M_PUT /request.htm HTTP/1.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Man: ContentCategory ; ns =17</td>
</tr>
<tr>
<td>17- ContentContext:: Research</td>
</tr>
<tr>
<td>Host: <a href="http://www.sample.com">www.sample.com</a></td>
</tr>
<tr>
<td>Contentlength: 1203</td>
</tr>
</tbody>
</table>

Figure 6.1 Sample requests

The headers can be applied with GET and PUT requests.

6.2.2 Conceptual diagram

Semantic search process mainly involves in providing context and category information to various phases of the existing search process. Figure 6.3 depicts the phases, which introduces semantic information in their operations. Query formulation, repository index building, searching and ranking phases of the search process incorporate the semantic information.

This research aims to use the semantic information throughout search process unlike using them at some stage of the search process. Using semantic information throughout the search process helps to improve the precision of search results at every stage.
6.3 PROCESS ARCHITECTURE

With the increasing interest in hypermedia applications, it is becoming important to represent more data describing the domain information about the resources on the web. Semantic search process attempts to augment and improve traditional search process. A common way to add context is to add additional search terms. Unfortunately this places an added burden on the user and it is sometimes difficult even for an expert to select the right query terms for the desired subset of information to be retrieved. The proposed search process reduces the effort required from the user in formulating the query and optimizes the search process. The architecture of the semantic search process is given in Figure 6.4.

Figure 6.2 Schematic diagram of semantic search process.
6.3.1 Phases of Semantic Search

The important phase in the semantic search process is to provide the context information during the repository and index building process. The semantics of the web page is provided using the entity headers. The entity headers—content category and content context provide the category and context information of the web page. The values to the entity headers is given by the web page authors which is considered as a design constraint where the chances of users manipulating the values for the sake of better search engine result ranking is possible.

Figure 6.3 Phases of semantic search process
The entity headers can be used with any of the HTTP methods as given in the Figure 6.1. For example if the system uses its link crawling process to build its database or uses the notification of the web servers, the entity headers carry the semantic information. All the servers under communication should understand the additional entity headers in order to execute the semantic search process.

Framing query to perform search and providing the context and category of the query under search is necessary to get more relevant results. The category and context of the user query is received in two ways. They are received either through user interface or from the user profile generated. The user profile consolidates the interests of the user by keeping track of the context and category of the page he visits. The query with the context and category is given as input to the query processor, which will convert them into respective identifiers and searched in the semantic index. The search results of the web pages are displayed after applying semantic ranking algorithm to rank the results in the order of relevance and importance.

Semantics is included in the ranking phase by extending the pagerank computation to include the pages with similar context and category. The intension of including semantics in the ranking process is to identify the most important page in a particular category and context. For example web pages with similar semantics only should be ranked to provide the search results. The general pagerank is calculated using the pagerank of pages which link to page A and the number of outbound links.
Semantic ranking calculates the semantic pagerank as follows

$$SPR(A) = (1-d) + d\left(\frac{SPR(T_1)}{SC(T_1)} + \ldots + \frac{SPR(T_n)}{SC(T_n)}\right)$$

where $SPR(T_i)$ – Pagerank of pages which link to same context and category to page $A$.

$SC(T_i)$ – number of outbound links of same semantics on page $T_i$.

Because of this links when a query word "neural networks" is given with computer science as category and academics as context, search engine results displayed will be ranked among the pages with similar semantics. This will reduce the rich get richer problem and give chance for web pages which lie deep inside of the search results to progress further.

6.3.2 Data structures

Semantic search process uses new data structures and extends the old ones to provide semantic information to the phases. The data structures that are used in text indexing, pagerank indexing and in generating the user profile are discussed in this section. Semantic information is provided to other phases using the protocol headers.

Semantic text indexing

The words are extracted from the documents gathered by the web crawlers and are indexed. The words and documents are given Ids and they are stored for each document gathered. The context and category of the document is extracted when the crawler issues request and receives the web
pages. In case if the new web page or updation of a web page is notified to the search engine repository, the semantic headers can transfer the information related to context and category of a web page. The structure of semantic index is given in figure 6.5 and 6.6.

Figure 6.4 Forward semantic index

Figure 6.5 Reverse semantic index

For every valid combination of category ID, context ID and word ID in the semantic lexicon, there exists a pointer document ID that points to the corresponding record in the reverse barrel.
Semantic rank index

Pagerank factor is considered as an important factor in calculating the rank of a web page. Calculating pagerank is also an offline process and it is calculates based on the links the web page possesses. Semantic pagerank index has the semantic page rank of the web pages which are indexed. The semantic page rank is used to improve the ranking of those pages with same semantics by eliminating the web pages with different semantics which may be of less interest to the users. The structure of the semantic rank index is given in Figure 6.7.

<table>
<thead>
<tr>
<th>Doc ID</th>
<th>Category ID</th>
<th>Context ID</th>
<th>IR</th>
<th>SPR</th>
<th>Rank</th>
</tr>
</thead>
</table>

Figure 6.6 Semantic rank index structure

6.4 RESULTS

Semantic search process is aimed at improving the relevance of search results and refines the ranking procedure to rank only those web pages which belong to the same semantics. The results of the semantic search process should test on the retrieval effectiveness and improvement in the ranking procedure.

6.4.1 Relevance test

Relevance is difficult to determine on an automated basis, because the concept itself is subjective. According to chief scientist AltaVista the relevancy depends on the task of the user, the context etc. The most typical
measurements of IR effectiveness are recall and precision. Recall measures the proportion of relevant documents in a database that is actually retrieved. Precision is the proportion of retrieved documents that is relevant.

\[
\text{Recall} = \frac{\text{relevant answers retrieved}}{\text{Total relevant answers}}
\]

\[
\text{Precision} = \frac{\text{relevant answers retrieved}}{\text{Total answers retrieved}}
\]

In order to include information related to inter document link structure different precision measures were described by Jace and Mark ( ). The evaluation of semantic search process includes the best precision and useful precision measure to determine the effectiveness.

6.4.1.1 Experiment

The effect of the semantics on the search results are tested with the database of approximately 600 web pages containing pages uploaded from Open Directory Project (ODP). The size of the database considered being very small but the sample is designed to study the impact of the semantics in an accurate manner. In order to find the relevance of the results, novice and professional users perform the searches. The subjective relevance scores are calculated based on the relevance model given in table 6.1. The operational model used is the same proposed by Jack and Mark ( ).
Table 6.1 Relevance model used

<table>
<thead>
<tr>
<th>RELEVANCE SCORE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>The most relevant</td>
</tr>
<tr>
<td>2</td>
<td>Partly relevant or contains a link to a page with a score 3</td>
</tr>
<tr>
<td>1</td>
<td>Somewhat relevant, for example short mention of a topic within a larger page, technically correct (i.e. terms appear on a page – including META tags) or contain a link to page ranked 2</td>
</tr>
<tr>
<td>0</td>
<td>Not relevant; no query terms found (META tags were examined well) or a bad hit.</td>
</tr>
</tbody>
</table>

The document cut off values considered for the analysis has been a set to 20 and set to 100. The two cut off values will help to determine how the precision deteroits with the nth hit. The graphs are generated for the following precision measures.

**Best precision**

This measure takes into account only the most relevant hits. It maps relevance scores to a binary measure, and thus additively of the values need not be considered in this case.

\[
\text{PrecBest (minFnHits)} = \frac{\sum_{i=1}^{\text{minFnHits}} \text{Count of (score} = 3)}{\text{minFnHits}}
\]

**Useful precision**

This measure takes into account only the most relevant hits and hits containing links to the most relevant ones.
User

The search is performed for calculating the relevance by two type of searcher – Novice and Professional. Two different type of searcher is used to test the system for its response to refined and unrefined queries. The searchers were allowed to uses the system and mark the relevance scores for the retrieved results. The relevance scores were used to draw graphs to depict the difference in the average precision of the searchers. The graphs are drawn for DCV 20 and 75 and the precision is recorded. The precision graphs are shown in figures 6.8 – 6.11.
Figure 6.7 Average precisions of results for small DCV

Document Cutoff Values (1-20)
Figure 6.8 Average precisions of results for large DCV
Figure 6.9 Best precisions of search results
Useful Precision of search results

Number of scores

Figure 6.10 Useful precision of search results
6.4.2 Impact on additional headers

So as to not affect the web client or server, the extensions to HTTP were implemented between two proxy servers. The design provided a convenient means to test the implementation with real clients and a real server without the need to modify either the client or server. This also demonstrates an architecture allowing for a point-to-point use of protocol extension on bandwidth constrained regardless of if the clients and servers support the protocol extension. Figure 5.11 gives the block diagram of the implementation.

Simulation was performed using a close to real world conditions as possible. A state enabled proxy was set up as the server. A second proxy was set up to act as state enable browser using the first proxy as its proxy server. Each proxy server gathered the semantic information. The list of URLs was
obtained from the list provided to the crawlers of the search engines of Apache server. The size of the request headers is tabulated in table 5.2.

<table>
<thead>
<tr>
<th>Request header</th>
<th>Header Tag</th>
<th>Size bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content _Category</td>
<td></td>
<td>82</td>
</tr>
<tr>
<td>&amp; Content _Context</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

6.5 DISCUSSION

The recorded scores of novice and professional users for average precision of the search results showed no major difference among the precision of the users. The search process with context in itself has helped in producing relevant results to the search query with context generated through user profile or received through input. The precision of the results reduces gradually with the increasing document cutoff values rather than irregular difference between the consecutive DCVs. The semantic ranking has helped in bringing more relevant pages to the top ranking filtering the web pages with required keywords and different context out of the top rankings. It is also recorded that with the increasing DCVs, the precision is not deteriorating like the conventional search results where the relevance approximately after page 5 is not much appreciated. In the search results of traditional search engines, the duplication of the links increases with the increasing DCVs. The results of semantic search process also demonstrate existence of duplicates in the result, which demands attention.
Best precision and useful precision results of search recorded, depict some variation in the way different users use the search engine. Best precision records the most relevant results to the query, and the novice user results recorded lower scores compared to the professional user. This is because the novice users are not proficient in framing appropriate query word for their requirement and he is not persistent. Professional users who have the idea of functioning of the search engines try to use Boolean operators to make their query more understandable to the search engines. Hence the best precision score of the professional users seem to be satisfactory.

Useful precision results recorded no variation in the scores, which indicated that the results identified as relevant are useful to further the search. The performance evaluation of the semantic search process is carried out in chapter 7 where it is compared with the results of traditional search engine.