Precise Environmental Searches: Integrating Hierarchical Information Search with EnviroDaemon

GEORGE CHANG
Kean University, Union, NJ

GUNJAN SAMTANI
Bear Stearns, Whippany, NJ

MARCUS HEALEY
Mobilocity, New York, NY

FRANZ KURFESS*
California Polytechnic State University, San Luis Obispo, CA

JASON WANG
New Jersey Institute of Technology, Newark, NJ

Abstract. Information retrieval has evolved from searches of references, to abstracts, to documents. Search on the Web involves search engines that promise to parse full-text and other files: audio, video, and multimedia. With the indexable Web at 320 million pages and growing, difficulties with locating relevant information have become apparent. The most prevalent means for information retrieval relies on syntax-based methods: keywords or strings of characters are presented to a search engine, and it returns all the matches in the available documents. This method is satisfactory and easy to implement, but it has some inherent limitations that make it unsuitable for many tasks. Instead of looking for syntactical patterns, the user often is interested in keyword meaning or the location of a particular word in a title or header. This paper describes some precise search approaches in the environmental domain that locate information according to syntactic criteria, augmented by the utilization of information in a certain context. The main emphasis of this paper lies in the treatment of structured knowledge, where essential aspects about the topic of interest are encoded not only by the individual items, but also by their relationships among each other. Examples for such structured knowledge are hypertext documents, diagrams, logical and chemical formulae. Benefits of this approach are enhanced precision and approximate search in an already focused, context-specific search engine for the environment: EnviroDaemon.

Keywords: information retrieval, context-specific search, environmental search engine, hierarchical information search

1. Introduction

Scientists and engineers have long harbored grand hopes for immediate, distributed network access to the entire science and technology literature. These hopes are well on

*Address correspondence to: Franz J. Kurfess, Computer Science Dept., California Polytechnic State University, San Luis Obispo, CA 93407. E-mail: franz.kurfess@computer.org

their way to being realized as a result of the steady improvement in the computing and communications infrastructure and the popularization of the Internet. The size of the organization able to perform search has decreased as groups of laypeople and scientists can now search ''digital libraries'' without the aid of trained reference librarians. Similarly, the document being sought has changed: from a citation with descriptive headers, to an abstract, to complete multi-media contents including text, audio, video, and animation. Often cited reasons for the rise in digital information are associated with the ideas of preservation of the contents of physical, paper-based texts, the convenience associated with maintaining, searching and retrieving electronic text, and the lowered cost of acquiring and maintaining bits as opposed to atoms [12].

A library collects, maintains, and indexes information for purposes of search, retrieval, and display. Since the advent of online information retrieval more than 30 years ago, the mechanism for retrieval has been syntactic: a user specifies a keyword-based query and all the documents containing those words are returned. In the 1960s there existed text search from technical citations. As the hardware improved and its cost plummeted, it became possible to store and retrieve more than just a reference. From the late 1960s to the mid-1980s, text search and retrieval of abstracts and then ''full text,'' which included the entire article and associated tables and figures, replaced bibliographic text searches. Since the mid-1980s to the present we have seen document search on local area networks and the Internet. This type of search is best characterized as structure-based because it involves dealing with structure and classification of complete documents and not just keywords. Whereas the software for information retrieval has remained virtually unchanged, the hardware has improved dramatically, and the mechanisms for retrieval of information are now changing significantly. Shortly after the turn of the century, information retrieval will migrate from keyword-based syntactic searches to concept-driven semantic searches across gigantic, distributed collections [16].

This article compares and contrasts tools for Internet search in a category-specific area: the environment. We provide context in terms of the history of information retrieval and the evolution of its research focus, describe the importance of current Web-based generic search engines, outline the need and tools employed to search the Internet, demonstrate how to measure effectiveness in terms of document retrieval, and compare and contrast our context-based search engine (http://cache.njit.edu:8080) to generic search engines.

1.1. Overview

For the layperson, information retrieval, document retrieval, and text retrieval are often regarded as synonymous. Initially, information retrieval was the sole province of the high priests of libraries: reference librarians. With the advent of word processing software and the CD-ROM, machine readable text could be created and disseminated by individuals with less specialized knowledge, which led to widespread growth of information retrieval systems. Document retrieval is fundamentally about indexing and searching, which have formed the focus of most of the research carried out by the information retrieval community to date. This is changing as a result of the explosion of interest occasioned by

the advent of the World Wide Web (WWW). While indexing and searching will remain the core focus of automated retrieval, browsing, with its sophisticated presentation through the delivery of graphical user interfaces, has also emerged as a research area, as the focus becomes more end-user-oriented. People now want to know how end-user-information retrieval systems should best be organized to facilitate effective retrieval, how best to categorize, filter, and route different types of information (e.g., text, audio, video, multimedia), in multiple languages, and an assortment of file types (e.g., GIF, MPEG, e-mail). As such, the concept of an ''end document'' disappears in a blizzard of information entities.

The WWW was started to facilitate the sharing of data in various formats by physicists at CERN, the European Particle Physics Laboratory. The WWW is a mammoth, heterogeneous, non-administered, distributed, global information system connected by hypertext links that is revolutionizing the information age. It is organized as a set of HTTP (hypertext transmission protocol) servers. A hypertext file format, HTML (hypertext markup language) is used to construct links between documents, which supports a hypertext data organization. It has strongly impacted the end user and the many potential benefits it augurs has spurred research in WWW site change detection [4, 18], information search/ filtering [3, 10], web/ database integration [2, 13], web querying systems [1, 15], website management [6, 4], web mining [5], and web visualization [8]. It is also bringing together researchers from such diverse areas as communications, electronic publishing, language processing, and databases as well as from specific domain areas such as the environment and manufacturing [9].

The most common technology employed for searching the WWW depends on sending information requests to ''index servers'' that index as many documents as they can find by navigating the network. A salient problem is that users must be aware of the various index servers, their strengths, weaknesses, and the peculiarities of their query interfaces. The need for querying information in the Web has led to the development of a number of tools that, based on keywords specified by the enduser, search the Web and return information related to the keywords. The limitations of browsing as a search technique are well-known, as well as the disorientation resulting in the infamous ''lost-in-cyberspace'' syndrome. Another problem arises in that these current queries cannot exploit the structure and topology of the document network.

The importance of search on the Internet has risen to the point that interactive analysis of digitized libraries will become a fundamental and indispensable part of virtually all scientific research. This is likely to be important in the future because there exists a plethora of information, the rate of information growth is accelerating, and it is difficult to precisely find the exact information sought. It is estimated that the ''solid'' science and technology literature is approximately 10^{13} bits, or 1 terabyte in size. In science and technology there are roughly 100 fields, and 1000 sub-fields, so that a sub-field's literature is roughly 1 gigabyte [16]. If the literature in a field doubles every 20 years, a field's literature growth rate will be approximately 4% or 250 megabytes per annum. The Web is likened to a 15 billion word (approximately 100 gigabytes) encyclopedia that contains, at its lower limit, some 320 million pages of information [11]. Finding a piece of information precisely and quickly, often with what is termed a ''fuzzy query,'' in this ever-increasing sea of information, represents a significant challenge.

Difficulties with locating relevant information about a particular topic not only translate into frustrated users of the Internet and the WWW, but may have substantial impacts on important aspects of our personal and professional lives, the economy, the environment, and many other areas. In the environmental domain, for example, sustainable development is influenced by environmental decisions throughout the entire lifecycle of products (in a wide sense, including non-tangibles and services). Especially in the early stages of the lifecycle, for example, in requirements gathering and design, decisions may be made with far-reaching consequences for the rest of the product's lifecycle. At this point in time, it is clearly impractical to teach all the people involved in such decisions all relevant environmental considerations. A more practical approach is to make tools available that allow those people to evaluate various alternatives of the product, and to assess the environmental impact of their decisions. Such tools will rely heavily on locating relevant information, and making it available in a suitable form when it is needed. Current tools for locating information, such as Web search engines, are mostly keyword based, and are not adequate for identifying relevant items and filtering out irrelevant ones. As a consequence, needed information is not easily available, and the quality of the work performed may suffer.

This illustrates the need for advanced tools and techniques to locate and access information relevant for a particular purpose. This topic, of course, has been investigated, especially in the area of information retrieval, for a number of years. Both the popularity of the WWW and advances in computer performance and processing techniques are paving the way now for a more widespread application of advanced information retrieval and knowledge representation methods.

1.2. Search Engines

At last count, there were at least 128 search engines (see ugweb.cs.ualberta.ca/~mentor02/search/search-all.html). Whereas a handful of search engines are functioning as true portals or entrypoints to the WWW, there is an increase in the number of category or context-specific search engines [7, 9]. This is because (a) no one generic search engine covers the entire WWW; and (b) context-specific search engines can provide more focused, precise searches by concentrating their indexing of relevant documents in specific domains [11, 21].

Search engines have automated the task of sifting for the tiny pearls of information scattered among the tons of useless data flooding in hyperspace. These modern-day descendants of the library's card catalog scan the Internet for data, organize it into a searchable database, and provide the user with a simple interface for making queries. To facilitate search, there exist decentralized indexes constructed by knowbots, robots, spiders, or humans that scan the WWW and construct indexes of useful keywords. These indexes can be useful in locating information by using browsers or other automatic tools. Yet, there is no high level query language for locating, filtering, and presenting WWW-held information. The situation today is analogous to that of a huge file system or a gigantic, mostly read-only, database with many useful indexes, but without a convenient facility for querying this information. One can retrieve information manually through

browsing and indexes or write special purpose programs to obtain specific pieces of information. In order to get a specific piece of information, one basically needs to know where it is located.

Arguably, the top 6 search engines are: AltaVista, Excite (Magellan and WebCrawler acquired by Excite), HotBot, Infoseek, Lycos, and Yahoo! Table 1 outlines some salient differences between these generic search engines. All are multi-domain and span the WWW. They are contrasted to single domain search engines, meta, and alternative search engines.

AltaVista is a pure multi-domain search engine that is, in reality, like the Yellow Pages for the WWW; it indexes the full-text of documents. Excite is an artificial intelligence adherent and employs concept-driven or ''fuzzy'' search. HotBot's slurp spider is the most powerful of all the Web ''creatures'' and can index the entire WWW in about a week. This translates into fewer out-of-date links. Infoseek is the most user-friendly, and possesses a clean, intuitive interface. Lycos is much like a bibliographic database service except that abstracts are generated by a program called a Web crawler, rather than a human indexer. Collected abstracts are full-text indexed and served from a computer center of file servers. Yahoo! is not really a search engine; it is a directory compiled by humans of the electronic subject world of the Internet. Others are focused on single domains [7] or are meta-search engines (go to Savvy Search; http://www.cs.colostate.edu/~dreiling/smartform.html) that harness the power of existing search engines. Still others employ user profiles and server logs to determine the most popular sites.

2. Problem

The field of information retrieval is fundamentally about indexing, searching, and retrieving information. Indexing is essentially classificatory and can involve abstraction (i.e., summarizing the most important parts) and extraction (i.e., identifying prespecified pieces of information) of document files and information requests. Searching involves queries that can involve structure (i.e., words or concepts). There are two basic types of hypertext queries: content queries which are based on the content of a single node of the hypertext; and structure-specifying queries which take advantage of the information conveyed in the hypertext organization itself. Retrieval involves enduser displays such as graphical user interfaces commonly, and can involve categorization, filtering/routing.

Most commonly employed indexing algorithms utilize an inverted list index, a tried and true methodology. Researchers develop new, theoretically sound algorithms, but few are employed and tested in the real world. Similarly, many search engines employ different hardware and software methodologies such as distribution of requests to multiple, parallel computers and ''flow control'' software which download often-used pages from sites to the user's hard drive [17]. Associated problems involve the size of the WWW and the rapid rate at which it is growing, the need for a blend of skills (i.e., librarian, scientist, technologist) to exhaustively search the Web, and the growing demand for fast, precise tools that can find exactly what the enduser wants—even though the

Table 1. Generic Search Engines.

Search Engine and URL	Concentration	Means	Salient features	Future direction
AltaVista www.altavista.digital.com	Syntax and semantic search capability in multiple languages	Scooter spider; updates WWW database everyday	Powerful, comprehensive, difficult to master; too many hits	Wants to partner; includes many other services: maps, stock quotes, peoplefinder
Excite www.excite.com	Concept-based search and subject matter-based topics	Index updated every 7–10 days	Powerful, but easy to master	True portal to WWW; includes many other services: maps, stock quotes, peoplefinder, chat links, sports stories
HotBot www.hotbot.com	Fast search capability of subject matter-based topics	Slurp spider; index updated every 7–10 days	Ease of use, speed	Strategy unclear; includes many other services: maps, stock quotes, peoplefinder
Infoseek www.infoseek.com	Search and directory of topics	Ultraseek and Ultrasmart	Ease of use	True portal to WWW; includes many other services: maps, stock quotes, peoplefinder
Lycos www.lycos.com	Search and directory of topics	Index updated weekly	Difficult; good for multimedia files, no newsgroup searching	True portal to WWW; includes many other services: maps, stock quotes, peoplefinder
Yahoo! www.yahoo.com	Subject matter based; indexing done mostly by humans	Hierarchical approach to topic organization.	Comprehensive in spanning the WWW	True portal to WWW; includes many other services: maps, stock quotes, peoplefinder

original query was ill-defined (i.e., ‘‘fuzzy search’’). At present, access to the WWW is based on navigationally oriented browsers. The end result is often a ‘‘lost-in-cyberspace’’ phenomenon because (a) there is no reliable road map for the WWW; (b) obtained information is heterogeneous and difficult to analyze; and (c) organization of documents conveys information which is not exploited. Currently available Web search tools suffer from some of the following drawbacks:

1. User partial knowledge is not fully exploited. For example, the user may know that a particular keyword is in the header or in the body of a document which would aid in its location and retrieval.

2. The restructuring ability of current tools is limited or nonexistent. The querying tool should permit *ad hoc* specification of the format in which the answer should be presented. One should be able to search for two chapters that have references to the same article or to view two references side by side when output is given.

3. The dynamic nature of Web documents are unaccounted for, which results in poor query result quality.

The core need of endusers is to quickly and easily help them find the exact piece of information they want (even though they might not be able to exactly describe their needs) without letting them drown in an information sea. When faced with the task of searching for something one can ask for recommendations for WWW sites from others or use Web indexes which are manually constructed and organized by category. Using this latter scheme, sites appear more quickly than can be indexed by hand and a search engine can rapidly scan an index of WWW pages for certain key words. A better solution involves the use of visualization methods.

3. Rationale

There are many reasons why information retrieval is a worthy topic of further investigation. One salient reason is the quest for the informational Holy Grail: realization of the grand vision for retrieval of all the scientific literature by anyone, anytime, anywhere, and via any digital device (almost). An associated rationale is the quest for consilience whereby bridging and integration of sub-fields will complement the reductionist scientific strategies of the past [20]. On a more earthly plane, cost and convenience also represent worthy reasons for pursuing ways to better index, search, and retrieve information.

EnviroDaemon [9] was built to ease the task of finding environment-related information. It automatically builds and updates a catalog of objects at pre-selected Internet sites. Users submit keywords and can restrict themselves to subsets of the indexed sites by choosing one of five search criteria (Pollution Prevention and Pollution

Control, Regulatory, International, ISO 14000, and Recycling and Materials Exchange), or the entire catalog can be searched. The results are returned rapidly and are embedded in several lines of text to provide context. If the text looks promising, the user can click on a hyperlink to access the full article. In contrast to generic search engines, EnviroDaemon is highly specific, focusing only on environment-related information. It employs Gatherers and Brokers of the Harvest system to gather, extract, organize, retrieve, and filter information. On top of Harvest is the Glimpse tool, which indexes the information and provides a query system for rapidly searching through all the gathered files. EnviroDaemon constantly updates its catalog of more than 35,000 environment-related objects from approximately 160 Internet servers. EnviroDaemon is a keyword based search tool which is now being extended by incorporating Hierarchical Information Search Tool (HIST) which will permit queries based on hierarchical structure. The rationale for this new extension is that search conditions incorporating both keyword and structure-based queries will help the user more precisely locate information.

4. Objectives

Many search engines have their distinctive features: multi-domain, single domain, meta-search engines which front-end other search engines. However, none of them make attempts to exploit the explicit underlying hypertext tags in a single domain. Our new search filtering tool, HIST, however, was able to exploit the full syntactic detail of any hypertext markup language and provide hierarchical query.

The essence of HIST is to permit endusers to specify in which parts of a document a keyword should appear: in a document title, in a section header, somewhere in a paragraph, or in a table. This allows more precise control over the search process, hence, results in much better selection of relevant material. Another salient feature of HIST is the idea of ‘‘fuzzy search.’’ The documents returned by HIST do not have to be exact match. Users have control over how similar the target document should be in the hierarchical query. Furthermore, at the speed that Web technology standard is proceeding, an information retrieval tool must be able to meet the challenges of the next generation of the document standard on the WWW. Since HIST is based on the document type definition (DTD) Model, it is suitable for the new emerging document exchange standard for the WWW—extensible markup language (XML).

This new tool—HIST, does not replace the existing EnviroDaemon search engine. The work presented in this paper enhances the EnviroDaemon for information searching on associated environment-related topics. For a general search on the Web, where the user has little knowledge about the nature of the document (in terms of document structure), we still encourage the use of EnviroDaemon. On the other hand, when the situation permits, our new tool will be extremely useful when the target document structure is partially or completely known. Figure 1 shows the user front-end for EnviroDaemon (http://cache.njit.edu:8080).

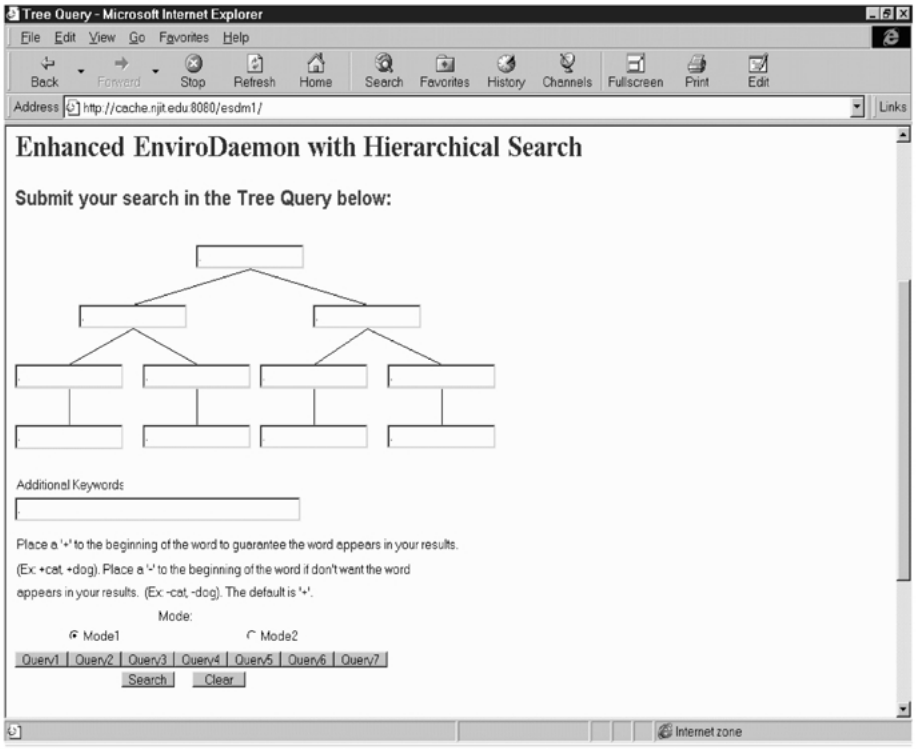

Figure 1. Front-end for EnviroDaemon search tool.

5. Methodology

5.1. System Architecture: An Overview

Figure 2 depicts the HIST system architecture. Our toolkit is composed of four modules: Extractor, Parser, Query Processor, and Tree Comparator. The Extractor retrieves the actual HTML pages from the Web to be used by the Tree Comparator.

The Parser translates the HTML pages into hierarchical tree structures based on the HTML DTD. The Tree Comparator contains various approximate tree and string matching programs while the Query Processor handles queries and invokes the Tree Comparator when necessary. The toolkit interacts with web browsers, the EnviroDaemon search engine, and Web servers.

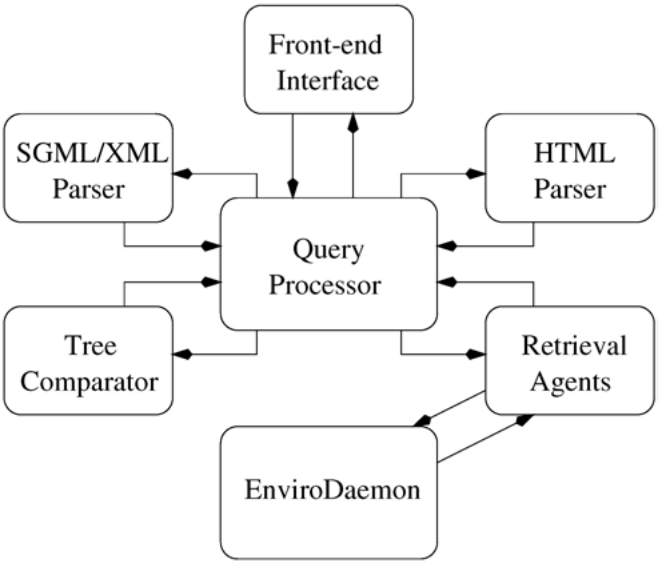

Figure 2. HIST system architecture.

6. Approach

Our approach is not meant to replace the existing WWW search facilities. In this sense, the work presented in this paper complements the tools that are available for Web searching. For ''global'' searches where the user has little knowledge about the information they want (in terms of possible locations, keywords *etc*.) and wants to query the entire WWW, we still advocate querying the entire WWW or a specific domain therein using a more domain-specific search engine. On the other hand, there are circumstances where the user has partial knowledge of the information required and could benefit from our approach. As many previous approaches have suggested, to analyze documents from the WWW, the first step is to derive a scheme that describes the HTML page. While specially designed schema provide a structural way of analyzing the hypertext document, much of the semantics associated with the document is lost after the

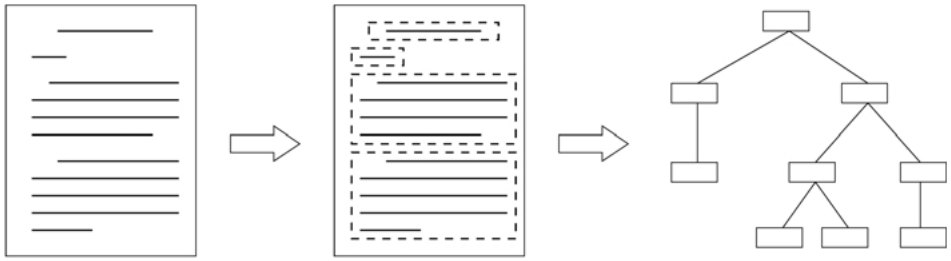

Figure 3. Convert hypertext document to a labeled tree using DTD.

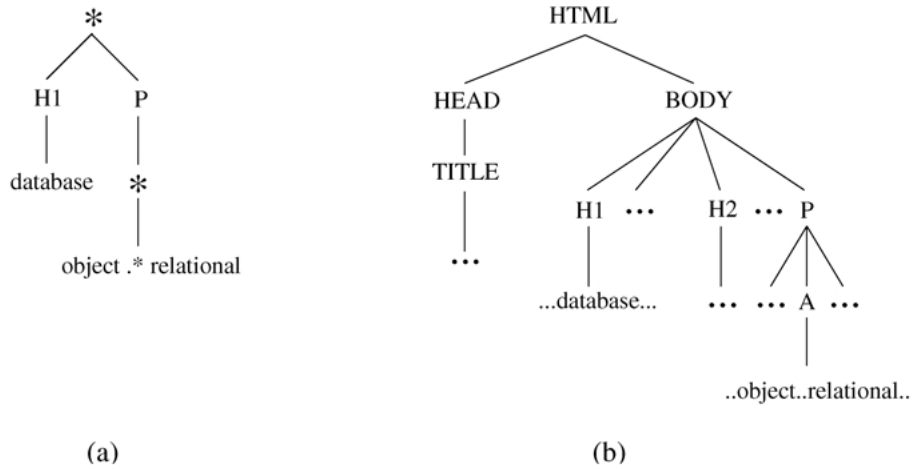

Figure 4. (a) A hierarchical query; (b) An HTML tree.

transformation process. Our approach differs from others in that we do not design a special schema, but, instead work with the DTD associated with hypertext documents.

With DTD in hand, we can parse each hypertext document using its own DTD, capitalizing on the fact that the DTD provides not only the grammatical information, but also semantic information. A document parsed tree structure of a typical article type document is illustrated in Figure 3. For HTML documents, we can parse it using HTML DTD 4.0. For article type SGML/XML documents, we can parse with article DTD.

6.1. Hierarchical Information Search

Consider the hierarchical query in Figure 4(a). This query is to find the HTML pages containing the word ''database'' in an H1 header followed by a paragraph consisting of ''object''' followed by ''relational.'' The ''*'' notation in the internal node of the query is a ''variable length don't care'' (VLDC) symbol, which represents an unspecified portion of a document [22] as described below. The query may be issued when an individual intends to locate some HTML pages available on the Internet while conducting a database-related research. Here the user places an emphasis on ''database'' and is interested in only those HTML pages having the word in an H1 header, rather than in any other place of a document.

To process such a hierarchical query using EnviroDaemon, we take the conjunction of the keywords appearing in the query and invoke our EnviroDaemon search engine to find the HTML pages containing these keywords. The EnviroDaemon search engine returns a collection of candidate uniform resource locators (URLs), ranked based on their relevance to the keywords. Duplicate URLs are deleted and a document corresponding to each matching URL is then retrieved using a libwww-Perl module, with time-outs set to 20 seconds to account for a busy network or failed connections. Each retrieved HTML

Table 2. Precision of HIST.

Search engine	Without HIST	With HIST
AltaVista	6/100	6/6
Excite	9/100	9/9
Infoseek	3/100	3/3
Lycos	4/100	4/4

document is then transformed into a hierarchical tree structure based on DTD described earlier. The transformed tree structures then became candidate HTML trees that will be compared with hierarchical query trees. Figure 4(b) shows an example tree for an HTML page. The tree is rooted, labeled and ordered (i.e., each node has a label and the order of siblings is important). An internal node represents an HTML tag and a leaf contains the associated text.

Our toolkit compares the query tree with each candidate HTML tree using the previously developed approximate tree matching [22] in conjunction with regular expression matching on leaves when it is required. The URL of qualified pages is then returned. In comparing the query with an HTML tree, a VLDC can be matched, at no cost, with a path or portion of a path in the tree. The tree matching algorithm calculates the minimum edit distance between the query and the tree after implicitly computing an optimal substitution for the VLDCs in the query, allowing zero or more cuttings at nodes from the tree [19]. Cutting at a node ''n'' means removing the subtree rooted at ''n.''

Given two trees T_1 and T_2, the algorithm runs in time $Q(|T_1| \times |T_2| \times \min\{\text{depth}(T_1), \text{leaves}(T_1)\} \times \min\{\text{depth}(T_2), \text{leaves}(T_2)\}$. Thus, for example, in matching the query in Figure 4(a) and the HTML tree in Figure 4(b), the ''*'' at the root in Figure 4(a) would be matched with (or instantiated into) the nodes HTML and BODY in Figure 4(b), and the ''*'' underneath P in Figure 4(a) would be matched with the node A (i.e., the Anchor tag) in Figure 4(b). The nodes $H1$, ''database'' and ''object.*relational'' in Figure 4(a) would be matched with their corresponding nodes in Figure 4(b). All the other nodes in Figure 4(b) are cut.

6.2. Experimental Results

In order to test the effectiveness of HIST in the environmental domain, we performed an experiment on hierarchical search query. Retrieval effectiveness is in terms of relevant

Table 3. Approximate retrieval with distance.

Search engine	Dist. 0	Dist. 1	Dist. 2	Dist. 3	Dist. 4	''Error''
AltaVista	6%	45%	12%	13%	0%	24%
Excite	9%	39%	33%	9%	0%	10%
Infoseek	3%	43%	34%	8%	0%	12%
Lycos	4%	31%	37%	9%	1%	18%

items retrieved. Recall refers to the percentage of relevant items that are retrieved and precision refers to the percentage of items retrieved in a search that are relevant. The query we evaluated is Figure 4(a). The query has been tested with four index servers (AltaVista, Excite, Infoseek, and Lycos). There were several thousand URLs returned by each search engine. To restrict the test set to a manageable number, we only processed the first one hundred URLs returned by each search engine and proceeded with HIST. The precision with and without HIST is summarized in Table 2. The results indicate that most URLs returned by search engines do not have the hierarchical structure specified in the query, and HIST was able to eliminate all of them.

The approximate search results are summarized in Table 3. HTML document distance with respect to query structure, ranging from 0 to 4 was found; where distance 0 indicates an exact match. The last column indicates the percentage of URLs that we were unable to retrieve due to a parsing error, and other network problems not uncommon on the Web.

7. Future Work

A number of new features will be added in the near future so as to enhance the depth and breadth of this context-specific, precise search engine. Addition of URLs related to sustainable green manufacturing, disassembly and demanufacturing is appropriate because of the evolution of the pollution control field towards pollution prevention, life cycle analysis, and industrial ecology. In terms of functionality, we intend to incorporate associative memory or natural language features similar to AltaVista. In addition, we will incorporate a website change detection tool so that the system can update the index dynamically by considering the changed portion only.

8. Conclusion

In this paper, we described the EnviroDaemon with Hierarchical Information Search on the WWW. The system makes several contributions. EnviroDaemon is context-specific and one of two environmentally directed search engines on the WWW (the other being the proprietary www.envirosources.com). It permits a more detailed search than that using the existing search engines. Additionally, it allows several kinds of approximate search at different levels: 0, approximate search on the structural level, based on ‘‘distance’’ between structures; 1, approximate search on the string level; 2, variable length don’t care on the structural level; and 3, any combination of the above.

It provides a SQL-like query language that allows users to flexibly combine a variety of constructs and has proven to be useful for the Web environment where the languages employed are HTML, SGML, and XML. EnviroDaemon with HIST promises to save users from drowning in an ‘‘information sea’’ and is generic enough to be useful for the WWW and intranets.

Acknowledgments

This work was performed while the authors were with New Jersey Institute of Technology in Newark, NJ. Partial support was provided through grants from the State of New Jersey, the U.S. Environmental Protection Agency, and the U.S. Department of Defense.

Notes

1. See Inxight's work at http://www.inxight.com
2. At last count there were some 30 million hosts (terminals), 2.2 million WWW servers, and approximately 1.6 million domains (personal computers with servers). Go to www.isoc.org/zakon/Internet/History/HIT.html, Hobbes' Internet Timeline for more information.
3. SGML is Standard General Markup Language, which defines formatting in a text document. HTML, a subset of SGML, is the foundation document format for the WWW.
4. libwww-Perl is a library of Perl packages/modules which provides a simple and consistent programming interface to the WWW. See http://www.ics.uci.edu/pub/websoft/libwww-perl/.
5. In order to evaluate any information retrieval system, performance is presented by calculating the recall $((100a)/a+c)$ and precision $((100a)/a+b)$ ratios.

References

1. S. Abiteboul and V. Vianu, ''Queries and computation on the Web.'' In *Proceedings of International Conference on Database Theory*, Delphi, Greece, pp. 262–275, January 1997.
2. P. Buneman, ''Semistructured data.'' In *Proceedings of the 16th ACM Symposium on Principles of Database Systems*. Tucson, AZ, pp. 117–121, May 1997.
3. R. Chandrasekar and B. Srinivas, ''Gleaning information from the web: Using syntax to filter out irrelevant information.'' In *Proceedings of AAAI Spring Symposium on Natural Language Processing from the WWW*. Stanford: California, March 1997.
4. S. S. Chawathe, A. Rajaraman, H. Garcia-Molina, and J. Widom, ''Change detection in hierarchically structured information.'' In *Proceedings of the ACM SIGMOD International Conference on Management of Data* Tucson, AZ, pp. 560–563, May 1997.
5. R. Cooley, B. Mobasher, and J. Srivastava, ''Web mining: Information and pattern discovery on the World Wide Web.'' In *Proceedings of the 9th IEEE International Conference on Tools with Artificial Intelligence* pp. 558–567, 1997.
6. M. Fernandez, D. Florescu, J. Kang, A. Levy, and D. Suciu, ''STRUDEL: A web-site management system.'' In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, Tucson, AZ, pp. 549–552, May 1997.
7. A. Glossbrenner and E. Glossbrenner, *Search Engines For the World Wide Web*. Peachpit Press: Berkeley, CA, 1998.
8. M. Z. Hasan, A. O. Mendelzon, and D. Vista, ''Applying database visualization to the World Wide Web.'' In *ACM SIGMOD Record* 25 pp. 45–49, 1996.
9. M. J. Healey, J. T. Lewis, and G. Samtani, ''Using EnviroDaemon to search the Internet for environmental information and building custom search engines,'' *Environmental Quality Management*, Spring, pp. 103–109, 1998.
10. Z. Lacroix, A. Sahuguet, R. Chandrasekar, and B. Srinivas, ''A novel approach to query the web,'' In *Proceedings of ER97 Workshop on Conceptual Modeling for Multimedia Information Seeking*, Los Angeles, California, November 1997.
11. S. Lawrence and C. L. Giles, ''Searching the World Wide Web,'' *Science* 280(3), pp. 98–100, 1998.

12. M. Lesk, ''Going digital,'' *Scientific American* pp. 58–60, 1997.
13. A. Y. Levy, A. Rajaraman, and J. J. Ordille, ''Querying heterogeneous information sources using source descriptions,'' In *Proceedings of the 22nd International Conference on Very Large Data Bases*. Bombay, India, pp. 54–65, September 1996.
14. K. Mahalingam and M. N. Huhns, ''A tool for organizing web information''. *IEEE Computer* pp. 80–83, 1997.
15. A. O. Mendelzon, G.A. Mihaila, and T. Milo. ''Querying the World Wide Web.'' *Journal of Digital Libraries* 1(1) pp. 54–67, 1997.
16. B. Schatz, ''Information retrieval in digital libraries: Bringing search to the Net.'' *Science* 275, pp. 327–333, 1997.
17. S. Thurm, and G. Anders. ''Inktomi IPO sparks another Internet frenzy.'' *Wall Street Journal* 1998.
18. J. T. L. Wang, K. Zhang, and D. Shasha, ''Pattern matching and pattern discovery in scientific, program, and document databases.'' In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, San Jose, CA, pp. 487, May 1995.
19. J. T. L. Wang, K. Zhang, K. Jeong, and D. Shasha, ''A system for approximate tree matching,'' *IEEE Transactions on Knowledge and Data Engineering* 6(4), pp. 559–571, 1994.
20. E. O. Wilson, *Consilience*. New York: Alfred A. Knopf, 1998.
21. T. E. Weber, ''Web's vastness foils even best search engines,'' *Wall Street Journal*. 1998.
22. K. Zhang, D. Shasha, and J. T. L. Wang, ''Approximate tree matching in the presence of variable length don't cares,'' *Journal of Algorithms* 16(1), pp. 33–66, 1994.