Appears in the Seventh International World Wide Web Conference, Brisbane, Australia, Elsevier Science, pp. 95-105, 1998.

Inquirus, the NECI meta search engine

Steve Lawrence and C. Lee Giles
NEC Research Institute,
4 Independence Way, Princeton, NJ 08540, U.S.A.
lawrence@research.nj.nec.com and giles@research.nj.nec.com

Abstract

World Wide Web (WWW) search engines (e.g. AltaVista, Infoseek, HotBot, etc.) have a number of deficiencies including: periods of downtime, low coverage of the WWW, inconsistent and inefficient user interfaces, out of date databases, poor relevancy ranking and precision, and difficulties with spamming techniques. Meta search engines have been introduced which address some of these and other difficulties in searching the WWW. However, current meta search engines retain some of these difficulties and may also introduce their own problems (e.g. reduced relevance because one or more of the search engines returns results with poor relevance). We present Inquirus, the NECI meta search engine, which addresses many of the deficiencies in current techniques. Rather than working with the list of documents and summaries returned by search engines, as current meta search engines typically do, the Inquirus meta search engine works by downloading and analyzing the individual documents. The Inquirus meta search engine makes improvements over existing search engines in a number of areas, e.g.: more useful document summaries incorporating query term context, identification of both pages which no longer exist and pages which no longer contain the query terms, advanced detection of duplicate pages, improved document ranking using proximity information, dramatically improved precision for certain queries by using specific expressive forms, and quick jump links and highlighting when viewing the full documents.

Keywords

Information retrieval; Search engine; Meta search; Context-based search

1. Introduction

A number of useful and popular search engines attempt to maintain full text indexes of the World Wide Web (e.g. AltaVista (http://www.altavista.digital.com), Excite (http://www.excite.com), HotBot (http://www.hotbot.com), Infoseek (http://www.infoseek.com), Lycos (http://www.lycos.com), and Northern Light (http://www.nlsearch.com)). However, searching the Web can still be a slow and tedious process. Limitations of the search services have led to the introduction of meta search engines, e.g. MetaCrawler [5] and SavvySearch [1]. A meta search engine searches the Web by making requests to multiple search engines such as AltaVista or Infoseek. The primary advantages of current meta search engines are the ability to combine the results of multiple search engines, and the ability to provide a consistent user interface for searching these engines [6]. The results of Selberg and Etzioni [5] suggest that the major search engines index a relatively small amount of the Web. Combining the results of multiple engines can therefore return many documents that would otherwise not be found. This paper describes the motivation for and operation of the Inquirus meta search engine, which includes a number of improvements over existing search engines. Data from the Inquirus meta search engine is used to create statistics on the speed of the major search engines.

2. Motivation

The principle motivation behind the Inquirus meta search engine was the poor precision, limited coverage, limited availability, limited user interfaces, and out of date databases of the major Web search engines. Expanding on these points:

Poor precision. The diverse nature of the Web, and the focus of the Web search engines on handling relatively simple queries very quickly, leads to search results often having poor precision. Additionally, the practice of ``search engine spamming" has become popular, whereby users add possibly unrelated keywords to their pages in order to alter the ranking of their pages. Our experience indicates that the relevance of a particular page is often obvious only after waiting for the page to load and finding the query term(s) in the page.

Limited coverage. Our experience with using different search engines suggested that the coverage of the individual engines was relatively low, i.e. searching with a second engine would often return several documents which were not returned by the first engine. The results of Selberg and Etzioni [5] suggest that the coverage of any one engine is limited.

Limited availability. Due to search engine and/or network difficulties, we have observed that the engine which responds the quickest varies over time.

Limited user interfaces. It is possible to add a number of features which enhance the usability of the search engines.

Out of date databases. Centralized search engine databases are always out of date. There is a time lag between the time when new information is made available and the time that it is indexed.

3. Previous work

The idea of querying and collating results from multiple databases is not new. Companies like PLS (http://www.pls.com), Lexis-Nexis (http://www.lexis-nexis.com), DIALOG (http://www.dialog.com), and Verity (http://www.verity.com) have long since created systems which integrate the results of multiple heterogeneous databases [5]. Many Web meta search services exist such as the popular MetaCrawler and SavvySearch services [6,1].

4. The Inquirus meta search engine

One of the fundamental features of the Inquirus meta search engine is that it analyzes each document and displays local context around the query terms. The benefit of displaying the local context, rather than an abstract or query-insensitive summary of the document, is that the user may be able to more readily determine if the document answers his or her specific query. A user can therefore find documents of high relevance by quickly scanning the local context of the query terms. This technique is simple, but can be very effective, especially in the case of Web search where the database is very large, diverse, and poorly organized. Users indicate that the page summaries generated using local context allow them to assess the relevance of documents more easily and more rapidly. Recent work by Tombros (1997) agrees: Tombros considered the use of query biased summaries and performed a user study which showed that users working with the query biased summaries had a higher success rate. The query biased summaries allowed users to perform relevance judgments more accurately and more rapidly, and greatly reduced the need to refer to the full text of documents. The display of local context does not require the use of multiple search engines and can be very useful even if only one engine is used. However, as with other meta search engines, Inquirus makes parallel queries to multiple search engines. The major features of the Inquirus meta search engine include displaying the context of the query terms, advanced duplicate detection, progressive display of results, highlighting query terms in the pages when viewed, insertion of quick jump links for finding the query terms in large pages, dramatically improved precision for certain queries by using specific expressive forms, and improved relevancy ranking. A more complete list follows:

1. The engine downloads the actual pages corresponding to the hits and searches them for the query terms. The engine then provides the context in which the query terms appear rather than a summary of the page. This typically provides a much better indication of the relevance of a page than the summaries or abstracts used by other search engines, and it often helps to avoid looking at a page only to find that it does not contain the required information (click, wait, click, wait, click, wait, ...). The context can be particularly helpful whenever a search includes terms which may occur in a different context to that required. The amount of context is specified by the user in terms of the number of characters to display on either side of the query terms. Most non-alphanumeric characters are filtered from the context in order to produce more readable and informative results.

2. Results are returned progressively after each individual page is downloaded and analyzed, rather than after all pages are downloaded. The first result is typically displayed faster than the average time for a search engine to

respond (see Section 7). When multiple pages provide the information required, the architecture of the meta engine can be helpful because the fastest sites are the first ones to be analyzed and displayed.

3. When viewing the full pages corresponding to the hits, the pages are filtered to highlight the query terms, and links are inserted at the top of the page which jump to the first occurrence of each query term. Links at each occurrence of the query terms jump to the next occurrence of the respective term. Query term highlighting helps to identify the query terms and page relevance quickly.

4. Pages which are no longer available can be identified. These pages are listed at the end of the response. Some other meta search services also provide ``dead link" detection, however the feature is usually turned off by default and no results are returned until all pages are checked. For the Inquirus meta search engine however, the feature is intrinsic to the architecture of the engine which is able to produce results both incrementally and quickly.

5. Pages which no longer contain the search terms or that do not properly match the query can be identified. These pages are listed after pages which properly match the query. This can be very important - different engines use different relevance techniques, and if just one engine returns poor relevance results, this can lead to poor results from standard meta search techniques [3]. Search terms in meta tags are treated as if they were part of the main text.

6. More advanced detection of duplicate pages is done. Pages are considered duplicates if the relevant context strings are identical. This allows the detection of a duplicate if the page has a different header or footer (e.g. a mailing list message archived in several places).

7. Kirsch has presented a technique for relevance ranking with meta search techniques [2] wherein the underlying search engines are modified to return extra information such as the number of occurrences of each search term in the documents and the number of occurrences in the entire database. Such a technique is not required for the Inquirus meta search engine because the actual pages are downloaded and analyzed. It is therefore possible to apply a uniform ranking measure to documents returned by different engines. Currently, the engine displays pages in descending order of the number of query terms present in the document (if none of the first few pages contain all of the query terms then the engine initially displays results which contain the maximum number of query terms found in a page so far). After all pages have been downloaded the engine then relists the pages according to a simple relevance measure. This measure currently considers the number of query terms present in the document, the proximity between query terms, and term frequency (the usual inverse document frequency may also be useful [4]):

$$R = c_1 N_p + \left(c_2 - \frac{\sum_{i=1}^{N_p - 1} \sum_{j=i+1}^{N_p} \min(d(i,j), c_2)}{\sum_{k=1}^{N_p - 1} (N_p - k)} \right) / \frac{c_2}{c_1} + \frac{N_t}{c_3} \qquad (1)$$

where N_p is the number of query terms that are present in the document (each term is counted only once), N_t is the total number of query terms in the document (each term is counted as many times as it appears), $d(i, j)$ is the minimum distance between the ith and jth of the query terms which are present in the document (currently in terms of the number of characters), c_1 is a constant which controls the overall magnitude of R, c_2 is a constant specifying the maximum distance between query terms which is considered useful, and c_3 is a constant specifying the importance of term frequency (currently $c_1 = 100$, $c_2 = 5000$, and $c_3 = 10\, c_1$). When there is only one query term we currently use the distance from the start of the page to the first occurrence of the term as an indicator of relevance. We have found that this ranking criterion can be particularly useful with Web searches. A query for multiple terms on the Web often returns documents which contain all terms, but the terms are far apart in the document and may be in unrelated sections of the page, e.g. in separate Usenet messages archived on a single Web page, or in separate bookmarks on a page containing a list of bookmarks.

8. The engine does not use the lowest common denominator in terms of the search syntax. The engine supports all common search formats, including boolean syntax. Queries are dynamically modified in order to match each individual engine's query syntax (other meta search engines also do this).

9. Inquirus uses a *specific expressive forms* search technique, which can dramatically improve precision for certain queries. The technique works by searching for specific ways of expressing the answer to a query.

5. Operation

Figures 1 and 2 show a sample response of the Inquirus meta search engine for the query `"image watermarking"`. The search form can be seen at the top, followed by links to the individual engine responses and a tip which may be query sensitive. Results which contain all of the query terms are then displayed as they are retrieved and analyzed (if none of the first few pages contain all of the query terms then the engine initially displays results which contain the maximum number of query terms found in a page so far). The bars to the left of the document titles indicate how close the query terms are in the documents (or how close they are to the start of the document for a single term) - longer bars indicate that the query terms are closer together. The engine which found the document (e.g. A = AltaVista), the age of the document (e.g. 1m = 1 month), the size of the document, and the URL follow the document title. After all pages have been retrieved, the engine then displays the top 20 pages ranked using term proximity information. The engine then displays those pages which contain fewer query terms, those pages which contain none of the query terms, those pages which contain duplicate context strings, and those pages which could not be downloaded. Links to the search engine pages which were used are then provided. Finally, the engine displays a summary box with information on the number of documents found from each individual engine, the number retrieved and processed, and the number of duplicates. Options for Inquirus include which set of search engines to use (e.g. Web search engines or Usenet search engines), the maximum number of hits, the amount of local context to display, etc. Figure 3 shows a sample of how the individual pages are processed when viewed. The links at the top jump to the first occurrence of the query terms in the document, and indicate the number of occurrences. The [Track Page] link activates tracking for this page - the user will be informed when and how the document changes.

Fig. 1. The first part of a sample response from the Inquirus meta search engine for the query `"image watermarking"`

Home Options Help Feedback **Inquirus**

Find: `"image watermarking"`

Web & Usenet | Web | Usenet | Journals | News | Tech | All

(Hide) Options:

Hits: 100 Context: 200 Cluster: No Tracking: No

Searching for: "**image watermarking**" using: HotBot Infoseek AltaVista Excite Lycos Northern Light Yahoo WebCrawler.

Tip: The query term links in the "Searching for" line lead to the Webster dictionary definitions.

Digital ***Image Watermarking*** A 1m 3k http://www.thomtech.com/tlab/projects/digh20.htm
...Digital ***Image Watermarking*** ... /... Digital **Image Watermarking** Digital **Image Watermarking** As publishers make digital images, audio, and video available on CDROM or online, unauthorized use is a g... /... Digital **Image Watermarking** Digital **Image Watermarking** As publishers make digital images, audio, and video available on CDROM or online, unauthorized use is a greater threat. **Image watermarking** - a process that encodes a signal or patter...

Related Technologies H 3m 1k http://image.hp.com/htdocs_present/iwhpoverview/tsld018.htm
... Related Technologies Related Technologies Image Content-Based Retrieval Research activities at HP Laboratories Early experiments on the WWW **Image Watermarking** Visible and Invisible systems For security, attribution, inventory tracking Smart Cards, Security and E-commerce Secure, Internet-based Commerce HP Imagine Card, Praes...

Watermark Example 1 A 1y 1k http://www.thomtech.com/mmedia/becker/wmark.htm
... Watermark Example 1 Multimedia Lab Digital **Image Watermarking** This site is still under construction..... Original **Image Watermarke**d Image Here's the watermark Click here to see another example ... /... Watermark Example 1 Multimedia Lab Digital **Image Watermarking** This site is still under construction..... Original **Image Watermarke**d Image Here's the watermark Click here to see another example ...

PC WEEK: A lasting way for artists to leave their mark I 11m 8k
http://www8.zdnet.com/pcweek/reviews/1209/09mark.html
...ir mark Digimarc's watermark technology embeds ``invisible'' digital information in computer-generated images By Herb Bethoney Using Digimarc Corp.'s PictureMarc **image watermarking** technology, illustrators and photographers will be able to copyright their work with a persistent "watermark" that is virtually imperceptible until read by Digimarc's software reader. Although the ...

Master Thesis Project: Analysis of ***Image Watermarking*** Methods H 5m 2k
http://www.it.isy.liu.se/students/master/marking.html.en
...Master Thesis Project: Analysis of ***Image Watermarking*** Methods... /... Master Thesis Project: Analysis of **Image Watermarking** Methods Link ping University , Department of Electrical Engineering , Information Theory Division , Student Information Master Thesis Project in Information Theory Ana... /... Link ping University , Department of Electrical Engineering , Information Theory Division , udent Information Master Thesis Project in Information Theory Analysis of **Image Watermarking** Methods Svensk version Examiner: Niclas Wiberg, nicwi@isy.liu.se Location: Link ping University, Department of Electrical Engineering Other...

[... section deleted ...]

Fig. 2. The second part of a sample response from the Inquirus meta search engine for the query `"image watermarking"`

Ranked pages:

R1 978 Digital ***Image Watermarking*** A 1m 3k http://www.thomtech.com/tlab/projects/digh20.htm
...Digital ***Image Watermarking*** ... /... Digital **Image Watermarking** Digital **Image Watermarking** As publishers make digital images, audio, and video available on CDROM or online, unauthorized use is a g... /... Digital **Image W ermarking** Digital **Image Watermarking** As publishers make digital images, audio, and video available on CDROM or online, unauthorized use is a greater threat. **Image watermarking** - a process that encodes a signal or patter...

R2 976 Steganography - ***Image watermarking*** I 0d 4k
http://www.cl.cam.ac.uk/users/fapp2/steganography/**image_watermarking**/index.html
...Steganography - ***Image watermarking***...
...http://www.cl.cam.ac.uk/users/fapp2/steganography/image_watermarking/index.html... /... Steganography - **Image watermarking Image Watermarking** Weakness of Existing Schemes 2Mosaic S ome companies have proposed an automatic system for...

R3 957 Abstract... A 2m 17k http://www.comm.toronto.edu/~deepa/abstracts.html
... Abstract... A Robust Digital **Image Watermarking** Scheme using Wavelet-Based Fusion D. Kundur and D. Hatzinakos 1997 IEEE International Conference on Image Processing Abstract We present an approach for still **image watermarking** in whi... /...Digital **Image Watermarking** Scheme using Wavelet-Based Fusion D. Kundur and D. Hatzinakos 1997 IEEE nternational Conference on Image Processing Abstract We present an approach for still **image watermarking** in which the watermark embedding process employs multiresolution fusion techniques and incorporates a model of the human visual system (HVS). The original unmarked image is required to extract the ...

[... section deleted ...]

No search terms were found in these documents:

0 USENET: sci.research.postdoc : PhD Scholarship L - 2k http://mineral.umd.edu/usenet/sci.research.postdoc/0042.html

USENET: sci.research.postdoc : PhD Scholarship in Operational Research - Imperial College PhD Scholarship

[... section deleted ...]

Pages with duplicate context strings to a page above:

D UCR/CMP - Sponsors E - 3k http://www.cmp.ucr.edu/sponsors/
UCR/CMP - Sponsors H - 3k http://138.23.124.101/sponsors/sponsors.html
...al airline of the Museum! Adobe Systems - imaging software Digimarc Corporation , the digital **image watermarking** company Micropolis, Inc. - file storage Pacific Lightwave - DS1 connectivity Pacific Coast...

[... section deleted ...]

These documents no longer exist:

Error 404 Not found Computer Software Vendors D http://guide.sbanetweb.com/softD.html

[... section deleted ...]

Engine	Response	Total	Retrieved	Processed	Duplicates
AltaVista	Yes	139	50	36	14
Excite	No	-	-	-	-
HotBot	Yes	84	25	23	15
Infoseek	Yes	72	62	39	9
Lycos	Yes	6	6	4	0
Northern Light	Yes	88	25	22	9
WebCrawler	Yes	4	4	4	1
Yahoo	Yes	0	0	0	0
Total		**393**	**172**	**128**	**48**
More documents were found but the maximum number of hits was reached.					

Fig. 3. Sample page view for the Inquirus meta search engine. Query terms are highlighted (only one occurs in this page) and the link at the top jumps directly to the first occurrence of the query term

Jump to: **image watermarking (1)** `http://www.comm.toronto.edu/~deepa/pub.html` [Track Page]

Tip: Click on the query term above to jump to the first occurence of the term.

[... section deleted ...]

D. Kundur and D. Hatzinakos, "A robust digital image watermarking scheme using wavelet-based fusion," *Proc. IEEE Int. Conf. on Image Processing*, Santa Barbara, California, October 1997.
Abstract(html), Postscript(396K), gzipped PS(154K).

D. Kundur, D. Hatzinakos and H. Leung, "A novel approach to robust blind classification of remote sensing imagery," *Proc. IEEE Int. Conf. on Image Processing*, Santa Barbara, California, October 1997.
Abstract(html), Postscript(433K), gzipped PS(111K).

[... section deleted ...]

The engine consists of two main logical parts: the meta search code and a parallel page retrieval daemon. Pseudocode for (a simplified version of) the search code is as follows:

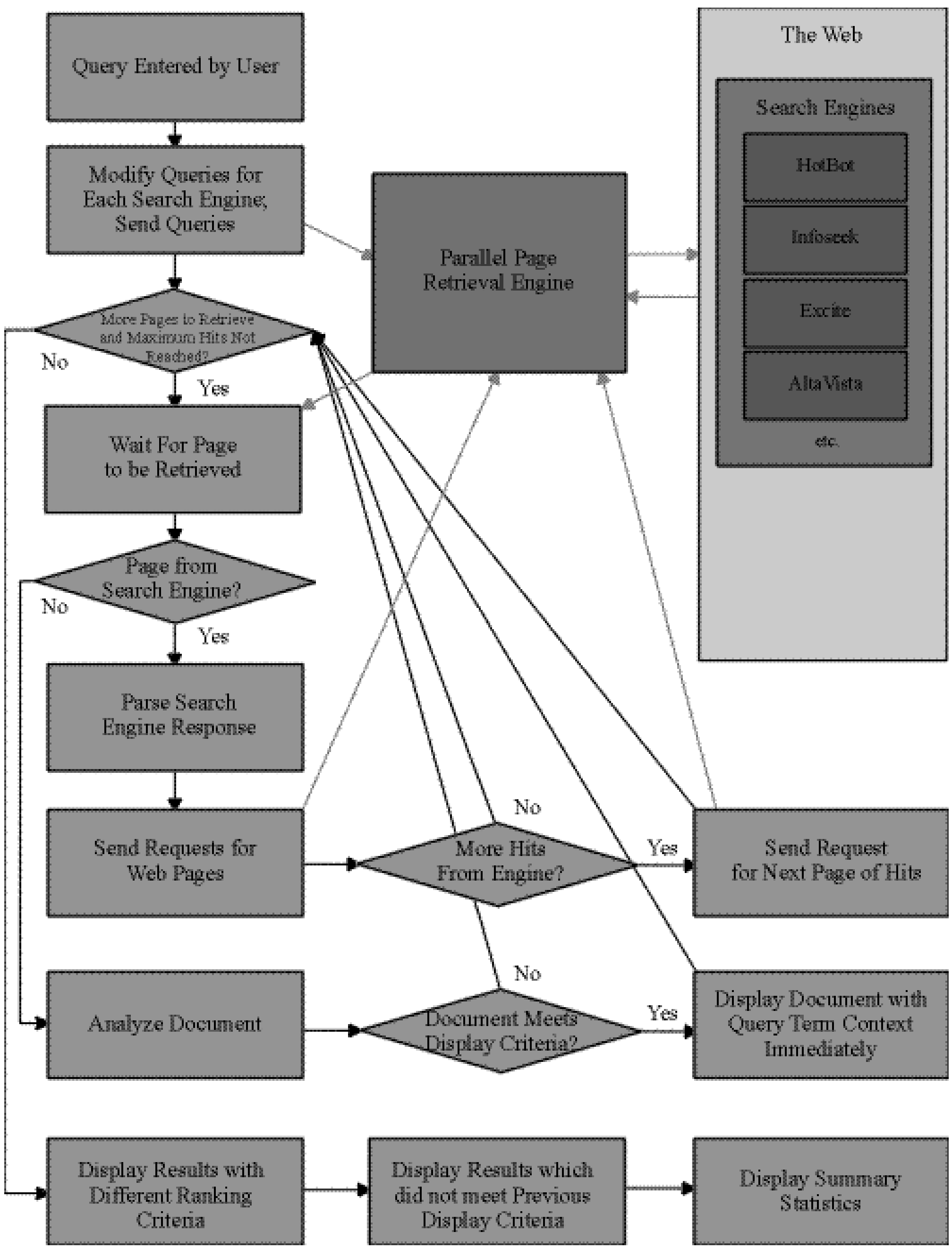
```
Process the request to check syntax and create regular expressions which are used to..
        ..match query terms
Send requests (modified appropriately) to all relevant search engines
Loop for each page retrieved until maximum number of results or all pages retrieved
        If page is from a search engine
                Parse search engine response extracting hits and any link for the next..
                        ..set of results
                Send requests for all of the hits
                Send request for the next set of results if applicable
        Else
                Check page for query terms and create context strings if found
                Print page information and context strings if all query terms are found..
                        ..and duplicate context strings have not been encountered before
        Endif
End loop

Re-rank pages using proximity and term frequency information
```

```
Print page information and context strings for pages which contained some but not all..
        ..query terms
Print page information for pages which contained no query terms
Print page information and context strings for pages which contain duplicate context strings
Print page information for pages which could not be downloaded
Print summary statistics
```

Figure 4 shows a simplified control flow diagram of the meta search engine. The page retrieval engine is relatively simple but does incorporate features such as queuing requests and balancing the load from multiple search processes, and delaying requests to the same site to prevent overloading a site. The page retrieval engine consists of a dispatch daemon and a number of client retrieval processes. The client processes simply retrieve the relevant pages, handling errors and timeouts, and return the pages directly to the appropriate search process.

Fig. 4. Simplified control flow of the meta search engine. Interactions with the page retrieval daemon are shown in gray.



6. Specific expressive forms

Accurate information retrieval is difficult due to the possibility of information being represented in many ways - requiring an optimal retrieval system to incorporate semantics and understand natural language. Research in information retrieval often considers techniques aimed at improving recall, e.g. word stemming and query expansion. It is possible for these techniques to decrease precision, especially in a database as diverse as the Web. The World Wide Web contains a lot of redundancy. Information is often contained multiple times and expressed in different forms across the Web. In the limit where all information is expressed in all possible ways, high precision information retrieval would be relatively simple - one would only need to search for one particular way of expressing the information. While such a goal will never be reached for all information, our experiments indicate that the Web is already sufficient for an approach based on searching for specific ways of expressing information to be effective for certain retrieval tasks.

Our proposed method is to transform queries in the form of a question, into specific forms for expressing the answer. For example, the query `What does NASDAQ stand for?` is transformed into the query `"NASDAQ stands for" "NASDAQ is an abbreviation" "NASDAQ means"`. Clearly the information may be contained in a different form to these three possibilities, however if the information does exist in one of these forms, then there is a high likelihood that finding these phrases will provide the answer to the query. The technique thus trades recall for precision. The Inquirus meta search engine currently uses the specific expressive forms (SEF) technique for a number of queries, e.g. `What [is|are] x?`, `What [causes|creates|produces] x?`, `What does x [stand for|mean]?`, `[Why|how] [is|are] (a|the) x y?`, etc. As an example of the transformations, `What does x [stand for|mean]?` is currently converted to `"x stands for" "x is an abbreviation" "x means"`, and `What [causes|creates|produces] x?` is currently transformed to `"x is caused" "x is created" "causes x" "produces x" "makes x" "creates x"`. Different search engines use different stop words (common words that are not indexed, e.g. "the") and relevance measures, and this tends to result in some engines returning many pages not containing the SEFs. We therefore filter out the offending phrases from the queries for the relevant engines. Figure 5 shows the response of the Inquirus meta search engine for the query `What does NASDAQ stand for?`. The answer to the query is contained in the local context displayed for 9 out of the first 10 pages. In contrast, the response of standard search engines often does not contain the answer to the query in any of the documents listed on the first page, even for engines which list support for natural language queries. It is reasonable to expect that the amount of easily accessible information will increase over time, and therefore that the viability of the specific expressive forms technique will improve over time. An extension which we have not currently implemented is to define an order over the various SEFs, e.g. `"x stands for"` may result in higher precision for the query `What does x stand for?` than the phrase `"x means"`. If none of the SEFs are found then the engine could fall back to a standard query.

Fig. 5. Inquirus meta search engine response for the query `What does NASDAQ stand for?`

Home Options Help Feedback **Inquirus**

Find: `what does NASDAQ stand for?`

[... section deleted ...]

Ref:...ex Search Previous Next Subject: Exchanges - The NASDAQ Last-Revised: 25 Oct 1996 From: billmanr@aol.com , jeffwben@aol.com , lott@invest-faq.com **NASDAQ is an abbreviation** for the National Association of Securities Dealers Automated Quotation system. It is also commonly, and confusingly, called the OTC market. Visit their home page: http://www.nasdaq.com...
Ref:... - Subject: Exchanges - The NASDAQ Last-Revised: 25 Oct 1996 From: billmanr@aol.com , jeffwben@aol.com , cml@cs.umd.edu **NASDAQ is an abbreviation** for the National Association of Securities Dealers Automated Quotation system. It is also commonly, and confusingly, called the OTC market. Visit their home page: http://www.nasdaq.com The NASD...
Ref:...on of Securities Dealers, Inc., the selfregulatory organization of the securities industry responsible for the operation and regulation of the NASDAQ stock market and overthecounter markets. **NASDAQ Stands for** the National Association of Securities Dealers Automated Quotation System. A nationwide computerized quotation system for current bid and asked quotations on over 5,500 over-the-counter stocks. ...

[... section deleted ...]

7. Efficiency

A simple analysis of page retrieval times leads to some interesting conclusions. Table 2 shows the median time for each of six major search engines to respond, along with the median time for the first of the six engines to respond when queries are made simultaneously to all engines, and the median time for the Inquirus search engine to display the first result. It can be seen that, on average, the parallel architecture of Inquirus allows it to find, download and analyze the first page faster than the standard search engines can produce a result, even though the standard engines do not download and analyze the pages. The Inquirus engine is surprisingly fast, with the only user comments regarding speed currently being that the engine is fast. These results are from 1,000 queries performed during September 1997, and we note that the relative speed of the search engines varies significantly over time, and depends on the location of the accessing site.

Table 2. Median time for a response from the search engines and the meta engine; the meta engine displays the first result faster than the average time taken by a standard search engine

Search engine	Median time for response (seconds)
AltaVista	0.9
Infoseek	1.3
HotBot	2.6
Excite	5.2
Lycos	2.8
Northern Light	7.5
All engines (average)	2.7
First of 6 search engines	0.8
First result from the Inquirus meta search engine	1.3

One potential drawback of the Inquirus search engine is that it uses significantly more bandwidth than other search engines. Although none of the current users have expressed concern, we expect that some Internet users may be concerned. The additional bandwidth requirements could limit the number of users which can simultaneously use a server based implementation, or present a disadvantage if Internet access is charged according to the volume of data transmitted. We note simply that the bandwidth requirements are not great compared to the increasing use of audio and video on the Web, and that, even if bandwidth requirements are important now, they will be less important in the future. Certainly, Inquirus is far more efficient than brute force search of the Web.

8. Conclusions

The Inquirus meta search engine demonstrates that real-time analysis of documents returned from Web search engines is feasible. In fact, calling the Web search engines and downloading Web pages in parallel allows the Inquirus meta search engine to, on average, display the first result quicker than using a standard search engine. User feedback indicates that the display of real-time local context around query terms, and the highlighting of query terms in the documents when viewed, significantly improves the efficiency of searching the Web.

Acknowledgments

We would like to thank Eric Baum, Kurt Bollacker, Adam Grove, Bill Horne, Bob Krovetz, John Oliensis, Maximilian Ott, Steve Omohundro, James Philbin, Majd Sakr, and Lance Williams for useful comments and suggestions.

References

[1] Dreilinger, D. and Howe, A., An information gathering agent for querying Web search engines, Technical Report CS-96-111, Computer Science Department, Colorado State University, 1996.
[2] Kirsch, S.T., Document retrieval over networks wherein ranking and relevance scores are computed at the client for multiple database documents, United States Patent #5,659,732, 1997.
[3] Notess, G.R. Internet "onesearch" with the mega search engines, *Online* 20(6): 36-39, 1996.
[4] Salton, G., *Automatic Text Processing: The Transformation, Analysis and Retrieval of Information by Computer*. Addison-Wesley, Reading, MA, 1989.
[5] Selberg, E. and Etzioni, O., Multi-service search and comparison using the MetaCrawler, in: *Proc. of the 1995 World Wide Web Conference*, 1995.
[6] Selberg, E. and Etzioni, O., The MetaCrawler architecture for resource aggregation on the Web, *IEEE Expert*, January-February: 11-14, 1997, http://www.cs.washington.edu/homes/speed/papers/ieee/ieee-metacrawler.ps
[7] Tombros, A., Reflecting user information needs through query biased summaries, Ph.D. thesis, Department of Computer Science, M.Sc. in Advanced Information Systems, University of Glasgow, 1997.

Vitae

Steve Lawrence graduated *summa cum laude* in 1993 from the Queensland University of Technology, Australia, obtaining highest honors for B. Sc. and B. Eng. degrees. He received his Ph.D. degree from the University of Queensland, Australia in 1997. His awards include a university medal, a QUT award for excellence, ATERB and APA priority scholarships, QEC and Telecom Australia Engineering prizes, and three prizes in successive years of the Australian Mathematics Competition. He is presently working as a Scientist at the NEC Research Institute in Princeton, NJ. His research interests include machine learning, neural networks, and information retrieval.

C. Lee Giles is a Senior Research Scientist in Computer Science at NEC Research Institute, Princeton, N.J and an adjunct faculty at the Institute for Advanced Computer Studies at the University of Maryland, College Park. His research interests are: neural networks, machine learning and AI; hybrid systems and integrating neural networks with intelligent systems and intelligent agents; AI and machine learning applications in finance, communications, World Wide Web and computer systems and software; and optics in computing and processing. He has served or is currently serving on the editorial boards of IEEE Transactions on Knowledge and Data Engineering, IEEE Transactions on Neural Networks, Journal of Parallel and Distributed Computing, Neural Networks, Neural Computation, Optical Computing and Processing, Applied Optics, and Academic Press. In 1994, he coedited a special issue for IEEE Transactions on Neural Networks on ``Dynamic Recurrent Neural Networks''. He is a Fellow of the IEEE and a member of AAAI, ACM, INNS and the OSA. Previously, he was a Program Manager at the Air Force Office of Scientific Research in Washington, D.C. where he initiated and managed research programs in Neural Networks and in Optics in Computing and Processing. Before that he was a research scientist at the Naval Research Laboratory, Washington, D.C. and an Assistant Professor of Electrical and Computer Engineering at Clarkson University. During part of his graduate education he was a research engineer at Ford Motor Scientific Research Laboratory. His degrees include a Ph.D. in Optical Sciences from the University of Arizona, an M.S. from the University of Michigan and a B.S. and B.A. from respectively University of Tennessee and Rhodes College.