



Metadata Searches of Unstructured Textual Content

A [Plugged In Software](#) White Paper
David Wood, CTO
26 September 2002

Abstract

This White Paper discusses search techniques for large collections of textual content. It specifically addresses the advantages and disadvantages of current full-text search methods and suggests ways to supplement those searches with the selective use of metadata. Techniques for combined metadata and full-text searches are given.

Current Search Methods for Textual Content

Every Internet user can relate to the concept of a full-text search. Full text search engines routinely enable people to find content of interest on the World Wide Web and other content sources such as collected USENET news groups.

Most large stores of textual data today are indexed by centralized search engines using full-text algorithms, such as Internet search engines Google, Alta Vista and Lycos. In the context of textual data, this generally involves collecting copies of documents and running them through a tokenizing parser to note word counts and occasionally more advanced concepts such as the nearness of associated words.

There are two important problems with full-text searches: They return only word inclusions (or exclusions) and they have no semantic understanding of the content. This results in lengthy result sets of (roughly) increasingly irrelevant content and problems with polymorphism. Polymorphism is the occurrence of a word with more than one meaning. Try searching the World Wide Web for the term "Sun" and the search engine will probably not be able to separate results for Sun Microsystems from the Chicago Sun-Times from electromagnetic studies of the star in the center of the Solar System.

Full-text indexing provides full coverage of textual content, but has inherent limits to the dimensionality of subsequent searches.

To address the problems inherent in full-text approaches, many universities and commercial search companies have developed statistical approaches to searching text. These techniques take many forms, but generally roam over a document collection to determine the statistical similarities between two documents. This information can be used to determine similar documents and even find words that are alike others ("concept matches").

Both full-text and statistical approaches require document collections to be indexed prior to searching. This often engenders a significant time delay before collections may be searched. Centralized full-text systems must update a central index. Statistical systems must generate a taxonomy, which serves as the basis for indexing new content. If the taxonomy is changed, the entire document collection must be reindexed.



Metadata and Metadata Searches

Metadata is literally data about data. In the context of textual data (which are generally collected into documents), metadata means information relating to the documents but generally held externally to them. Information held in an electronic file system, for example, might include a file's creation date and time, its owner and restrictions on who may view or modify it. That information is held outside of the file itself. Some systems, notably word processors such as Microsoft Word, hold metadata within a document format but display it separately. This is the case with Microsoft Office Properties.

In the case of a word-processing document or an electronic mail message, metadata might include the author, the recipients, the subject, keywords, concepts addressed, people named, dates or places mentioned, etc. Note that this description of metadata goes well beyond the sort of metadata held in Microsoft Office Properties or the metadata fields of existing document management systems.

The interesting property of metadata is that we know what a metadata element means. We know that an author of a document is a person and that the field value is a person's name. We know that the number of pages in a document is an integer and that it may be added to other page counts to produce a total. In short, we have semantic information (meaning) as well as syntactic information (structure or arrangement) about the data.

This is not to infer that we may trust all sources of metadata. The Microsoft Office Properties mentioned earlier are notorious for holding little useful information and may in fact hold bogus information. However, automated metadata extraction techniques becoming commonplace provide mechanisms for discovering and extracting useful metadata such as names of people and places, citations and statistically-derived concepts and summaries. The statistical concept extraction systems mentioned above can provide one form of useful metadata, as may linguistic, pattern or even manual systems.

Metadata searches against very large data stores are effective if good-quality metadata is automatically extracted. In fact, the creation of large amounts of metadata is a way to provide structure to otherwise unstructured content. This "structure" is generally held externally to the original content.

A metadata search consists of a search using metadata values and/or metadata types (e.g. personal names or dates). We may extend this concept by allowing navigation of a metadata space; by placing the metadata into a directed graph data structure, we can determine which metadata elements are related to others and to what degree.

Metadata searches are necessarily bounded by tighter constraints than full-text searches. In practice this results in highly relevant search results and provides additional search criteria (i.e., related resources). Metadata extraction, however, cannot provide full coverage of textual content.

Conceptual Examples of Metadata and Full-Text Searches

Consider a typical search on the World Wide Web. Suppose we want to search for papers by James Gosling of Sun Microsystems. Our search term might be (James Gosling Sun Microsystems). We might use a search engine capable of matching phrases and modify our search term to be ("James Gosling" "Sun Microsystems"). As of this writing, Google returned 9,040 documents for this search. James is well documented on the Web. Unfortunately, the results were a jumble of interviews, conference proceedings, articles, advertisements, etc. Try adding the term "paper". This resulted in only 2,320 documents. The results included some red herrings such as a news story about a Sun Microsystems program for software developers which happened to mention James Gosling and included the word "paper". That story (and several other red herrings) were listed in the top ten search results. This is a common feature of full-text searches and is due to the lack of semantic understanding.



If a system existed which knew that James Gosling was a person and that by "paper" we meant an academic publication, we would stand a better chance. We would even be able to differentiate in our search whether James was the author, editor or contributor of the papers in question or whether a paper by him was cited in a paper by others. This would be a metadata search.

Plugged In Software has developed a commercial metadata management system, Tucana Intelligent Connections™. Tucana allows metadata to be extracted from very large disparate data stores and searched. Tucana represents part of a new breed of intelligent search technology which will form the basis for the Internet's future search infrastructure. Tucana implements part of the World Wide Web Consortium's Semantic Web vision.

Combining Search Techniques

A perfect metadata system would know that "James" was the first name of James Gosling. But what about Henry James? Mistakes are easy to make. Many Chinese people reverse the order of their names when living in Western society. In the name "Mu Lan", for instance, which is the family name? It may be impossible to know. In a real-life metadata system we may know that "Mu Lan" is a person, but not know how to separate the name and assign further semantic meaning to the parts. Thus, there are times when a full-text approach may still be beneficial. We may want to search for authors of papers whose name includes the word "Mu". To do that, we require a hybrid system.

A hybrid system could use statistical methods in combination with linguistic, pattern and other methods to extract meaningful metadata. Storing this metadata in a database implementing a directed graph data structure would allow relationships between the information to be determined. A full-text engine could be used to index both textual content and metadata. Queries could be performed on both the metadata and the full-text index to process the query.

Perhaps we want to construct a search which asks for documents authored by James Gosling that deal conceptually with computer programming languages and which include the phrase "garbage collection". A search engine capable of processing this query would require a statistical algorithm to make conceptual mappings, a metadata system to determine that the person James Gosling is an author and a full text engine to find all instances of "garbage collection".

A Hybrid Approach Using Tucana Intelligent Connections

Plugged In Software's Tucana Intelligent Connections implements such a hybrid system. Tucana Intelligent Connections is comprised of two components; the Tucana Metadata Extractor and the Tucana Knowledge Store. The Tucana Metadata Extractor includes metadata extraction tools which implement a variety of algorithms and may be readily extended. The Tucana Metadata Extractor can automatically extract metadata, build a taxonomy of concepts and store its results in the Tucana Knowledge Store. Very large document collections may be indexed in this way and automatically kept up to date. The Tucana Knowledge Store is a distributed database for metadata built on a directed graph model. It includes a query engine capable of handling combined metadata and full-text searches. Full-text searches are passed to an external full-text engine (often, but not limited to, Lucene) for processing.

Queries to the Tucana Knowledge Store are made via the Tucana Query Language. There is, as yet, no international standard for metadata database query languages, although several proposals have been made to the World Wide Web Consortium. Plugged In tracks these developing standards. Information on the Tucana Query Language may be found in the TKS Tutorial or the TKS user documentation, both of which are available on Plugged In's Web site.



Conclusion

The combination of metadata and full-text searches can reduce the size of potential result sets and yield highly accurate results. This helps make the case for automated extraction of metadata and the expansion of metadata to include a wider range of features than traditionally encompassed in the term. The Tucana platform implements these concepts to provide a rich search environment for very large content collection.

Resources

- 1.# Plugged In Software's Home Page: <http://www.pisoftware.com> #
2. Plugged In Software's Products: <http://www.pisoftware.com/products>
3. Tucana FAQ: http://www.pisoftware.com/products/tucana_faq.html
4. Tucana Knowledge Store Tutorial: <http://www.pisoftware.com/products/TKS/docs/user/tutorial.html>
5. Tucana Integration Guide: http://www.pisoftware.com/downloads/tucana_integration_guide_US_letter.pdf
6. Understanding Tucana: http://www.pisoftware.com/products/understand_tks.html
7. Resource Description Framework. World Wide Web Consortium. <http://www.w3.org/RDF/>
8. Semantic Web. World Wide Web Consortium. <http://www.w3.org/2001/sw/>
9. This White Paper: <http://staff.pisoftware.com/dwood/publications/MetadataWhitePaper.html>