The Semanference System: Better Search Results through Better Queries

Anthony Scime and Colleen Powderly
State University of New York College at Brockport, Brockport, NY, USA

ascime@brockport.edu powderly@rochester.rr.com

Abstract

A method to create more effective Web search queries is to combine elements of a semantic approach with a template that requests specific details about the searcher's information need. Fundamental to this process is the use of semantics. Nouns, key phrases, and verbs are scored according to their frequency of use, then ranked as keywords and used to create the query. Key phrases and words in the query accurately represent the concepts of the text, generating search results that are significantly more accurate than those available using current methods.

Keywords: Information Retrieval, Text Processing, Query Refinement, Semantic Web Searching, Natural Language Processing

Introduction

Searching the Web has become a helter-skelter enterprise, which may or may not yield effective results. Most searches return a large number of results. The massive amounts of information available on the World Wide Web make quality results, a reasonable number of matches containing information relevant to the search terms, difficult to find using traditional search methods. In fact, Lucas, Schiano, and Crosett note that the explosion of information available on the World Wide Web is actually causing a narrowing of search spaces (i.e., a growing tendency among users to target searches to their specific knowledge domain) (Lucas, Schiano, and Crosett, 2001). Rather than narrowing the search space, a better method may be to narrow the focus of the search terms.

Search engines search the World Wide Web for information by matching keywords on the two halves of an inference network. Web search may be thought of as an inference network made from document keywords and concepts (Turtle and Croft, 1996). In the development of inference networks, information in a document is reviewed and keywords or phrases extracted. These keywords represent the content of the document and can be used to retrieve the information itself. The document is represented as a node on an inference network. It is connected to concept nodes, which represent the meaning of the document. These concepts are in turn simply expressed by keyword representation on the inference network. Identification of a keyword leads to concepts and on through the inference links to all the applicable documents.

The user expresses their need for information as a query within their understanding of the information source schema. In the process of developing the query, the user goes through an incremental breakdown of the information need. First, general categories of the need must be determined and then specific query keywords selected. This generates a query network in which the sum of the keywords and operators becomes the query sent to the search engine.

The query is then a matching of the keywords as query network end nodes or leaves to the keywords in the document leaves. The documents retrieved are those that are found by following the document links to the top of the network. In Figure 1 the query (I) is formatted through query concepts (q1 and q2) into keywords that match keywords in the document network and return documents d2 and d3.

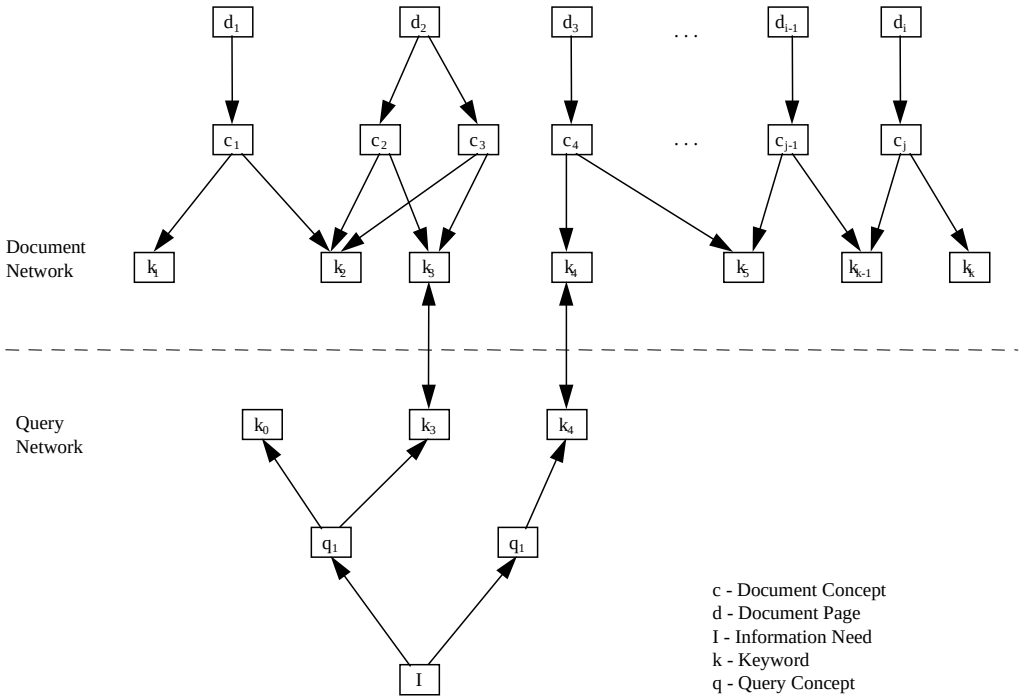

Figure 1 Page Retrieval Based on an Information Need

The indexing process used by engines searching for Web pages forms the document half of the inference network. Inefficiency occurs because indexing relies on ineffective input. This input is either directly created, via author definition through the use of metatags or other means of specifying keywords, or indirectly created, through automatic or manual analysis of content or location. Work to improve these problems is occurring in the research on search engines.

Our focus is on the other half of the inference network, the query. A typical query is a keyword or collection of keywords the user has cobbled together to express their information need. As such, it is a representation of the information need. Most of the research in this area involves the automatic personalization of searches, using agents that collect personal or group preferences for Web pages or Web page concepts. However, at times, a Web searcher may look for something they have never sought before. The keyword query then becomes inefficient as an expression of information need because of the searcher's inability to express the need in terms that match the search engine's half of the inference network.

Improvements can be made at the query end of a Web search by processing a textual description of the searcher's concept. This approach identifies semantic elements matching the searcher's concepts; these elements are then extracted from the textual description. Nouns, key phrases, and verbs are scored according to frequency of use, then ranked as keywords and used as the query half of the inference network. The leaves of this half of the inference network are therefore effectively narrowed to keywords specific to the information need. This process thus improves user queries to search engines.

Related Work

Since queries are constructed using words, a semantic approach is logical. A great deal of interest is focused on this problem, along with a great deal of effort. Similar approaches have been used to create the document half of inference networks so that better search engine indexes can be built. One set of solutions uses ontological creation, which constructs narrow "dictionaries" of related terms and concepts. The WebOntEx project attempts this. A parser/scanner is used to parse pages structurally, then to choose an arbitrary number of words at the beginning of the document as entity concept candidates. An ontology is built from them, which is specific to a particular domain. This process can be applied to any document from any domain; its result is based on realistic Web page structure (Han and Elmasri, 2001). Still, one big problem is that the information located in the first *n* words of any document may or may not be suitable for concept-designation; this can lead to inaccuracies.

Martinez-Trinidad, Beltran-Martinez, and Ruiz-Shulcloper use a better approach to concept identification. Their CLASITEX system identifies the main concepts of both English and Spanish documents by constructing trees of concepts for specific domains. While they successfully use phrase variants to identify concepts, they find greater success with long documents using this system than with short ones (Martinez-Trinidad, Beltran-Martinez, and Ruiz-Shulcloper, 2000). Given the limited number of lengthy documents currently found on the Web, this approach is of limited effectiveness.

A better approach identifies sentences in documents and assigns them tags based on their structure. Iatsko works with abstracts as examples of documents with relatively fixed structures, and delineates new ways of classifying them. He notes that differentiation between abstracts also comes from connections between sentences called "super phrasal units," and that these can be parallel or linear (Iatsko, 2001). He is on the right track in terms of his solution to the problem; he understands that text can be dealt with in terms of structure as well as content, and offers a workable partial solution. The problem is, however, that it is only a partial solution because it does not break down the documents with sufficient granularity to maximize information flexibility.

A system which combines some of the granularly ontological features of the WebOntEx project with structural parsing similar to Iatsko's comes from Laforest and Flory, who present a system for querying electronic documents. They define three types of structures: freetext documents which "only contain information and formatting instructions" and are very imprecise to work with; strongly structured documents, which "contain information and semantics guides" which are so rigidly structured that they are insufficiently atomic; and weakly structured documents, which "use DTDs containing optional tags. . . . Most tags delimitate paragraphs rather than data. A paragraph contains freetext which may include many data." Laforest and Flory choose weakly structured documents and tag paragraphs according to a predetermined domain. They call their system the Documents- and Rules-based User Interface for Databases (DRUID); it uses a database of weakly structured documents and an analyzer to extract data and place it in a classical database, along with queries of data in that database and/or of documents in their database (LaForest and Flory, 2001). The example they use involves medical records; medical records can reasonably be expected to contain "prognosis" and "prescription" indicators, which can be structured as paragraphs. This system clearly is useful only in a limited domain, one in which specific information is expected to be found in the documents being analyzed. Nonetheless, some points of Laforest's and Flory's system address the problem effectively. For instance, they suggest tagging by smaller units than full documents. They break documents into logical units or concepts. Their searches are then conducted against these concepts rather than the entire document. If implemented properly, this could lead to more effective searches than the current system of keywords.

Adelberg's NoDoSE project attempts to learn how to extract information from plain text files. Plain text files are particularly difficult to extract information from because they are not clearly structured, they are not clearly marked or tagged for structure, and text contains human errors. With NoDoSE, data is manu-

ally structured from documents. First, a data model is specified; next, documents are hierarchically decomposed into components of the model; and, finally, an output format is specified. Then NoDoSE maps the output structure and text via a tree in which each node/leaf holds a single data type. Despite this system's greater granularity than other systems, Adelberg notes that failures have occurred with documents because NoDoSE's parsing theories depend on constant markers (Adelberg, 1998).

Another semantically based approach is taken by Shian-Hua Lin and associates. Their system, the Automatic Classifier for Internet Resource Discovery (ACIRD), is designed to automatically classify documents. It begins with a training phase. Manually classified documents are used as a training set to teach the system classes. Then HTML tags on selected documents that contain potentially useful information (for example, the title of the document) are weighted to determine a degree of support given to a particular class or term within that class. These terms are analyzed, as are associations between terms. The results are used to form an inference model. The words in the inference model's leaves become keywords. Lin and company compared the results of their process with manual keywords developed by experts for the same documents, and concluded that their process was effective (Lin, et. al., 1998). This makes a positive example of using semantics and inference modeling to complement standard Internet search techniques.

Yi and Sundaresan took a somewhat different approach. They identified "pair[s] of inter-related phrases such as (book, author) . . . [and] (acronym, expansion) relations." They searched the Web for pairs of acronyms and their expansions, known as (A,E) pairs, and distilled formation rules from the acronym-expansion linkages. These acronym formation rules consist of listed replacement rules and "intermediates," the words which are found in the expansion, but are not used in the acronym. Their results showed a significant increase in the number of (A,E) pairs identified (Yi and Sundaresan, 1999). However, as with other processes described by other researchers, this is limited to HTML-tagged documents. While working with such pairs should not be discounted as a potential tool for mining Web documents, it must be broadened beyond the strict controls placed on this experiment before it can be seen as an effective solution.

Any study of semantically based approaches to Web queries must include a discussion of Natural Language Processing (NLP) techniques. A great deal of work has been done using NLP. Losee describes the history of NLP usage within the context of Web queries. He advocates the use of a structural approach to queries, particularly phrases and POS tagging. He notes the "brief and topical . . . nature" of queries devised using current methods (775), as well as the need for greater input and specificity of noun-based tags. However, he finds POS tagging of long phrases ineffective because of a corresponding loss of linguistic nuance (Losee, 2001). Losee's finding underscores the need for additional structural techniques to be applied to Web queries.

Christiansen and Chater deal with the connectionist approach to NLP. They note that incremental training of the dataset is necessary until a critical mass of data is built; past this point a system can become successfully self-training. They also achieve limited success using simple phrase identification (Christiansen and Chater, 1999).

Perez-Carballo and Strzalkowski have progressed substantially with NLP approaches to queries. In fact, they conclude that "topic expansion appears to lead to a genuine, sustainable advance in IR effectiveness" (157). They work with phrase identification, particularly the identification of important concepts in a given domain. Additionally, they recognize the importance of term weighting and scoring. Their "streams architecture," when applied to text to improve query results, consistently yielded significantly improved results when long Web queries were used. As a result of this finding, they applied their architecture to long text sequences; this improved their results in excess of 40%. Next, they experimented with several query expansion methods, including an interactive one. They report that, with maximally allowable responses of 1000, their queries on several occasions drew fewer than 1000 responses (Perez-Carballo and

Strzalkowski, 2000). Nonetheless, this number of responses is still far too large for usability by all but the hardiest researchers.

And therein lies the rub. Many Web users who create queries are unwilling or unable to wade through hundreds of responses. They examine the first page or two of results, which often yield articles wide of their intended mark, and give up on the Web as a useful source of information. This state of affairs must change if the World Wide Web's power is to be effectively harnessed. Users must learn effective query techniques, just as indexing and searching techniques must be improved, before the Web's full educational potential can be tapped. Notably, much of the above work, which has been applied to documents from the Web, can also be applied to non-Web documents. Techniques for extracting meaning from non-Web documents can be used to define the user's search intent by reducing their expression of need to some effective search keywords. These keywords will yield results superior to those traditionally used.

The Semanference System

One solution to the problem of more effectively searching the Web is to ask the searcher to write a narrative expressing their information need. This narrative follows a template that requests specific information about the statement of information need. After semantically processing this narrative, nouns, key phrases and verbs are scored according to their frequency of use, then ranked and used to create keywords for queries. Because the key phrases and words in the query accurately represent the concepts of the narrative, search results are significantly more accurate than those achieved by direct keyword selection. Obviously, the use of semantics is fundamental to this process. But a semantic approach, like any other, must use as much granular information as possible; consequently, specific information must be requested from the user. The granular information must be refined by a number of semantic and grammatical processes to produce meaningful keywords for the search queries (Figure 2).

Illuminating the Information Need

As noted above, a common problem with the user's request is lack of specificity, so that unusually large numbers of responses are returned. For example, if a user wants to investigate the progress of the "patients' bill of rights" through the U. S. Senate over the past two years, a search using the Google search engine, for example, may be initiated with the following terms: U.S., Senate, Republicans, Democratic, majority, HMO, patients, bill, rights. The returned documents number 1840, which is unwieldy. Add "lawsuits" and it reduces to 793, which is still not manageable. Add "treatments" and "opinions," and the number becomes a skimmable list of 130 items. However, if terms are added which specify the topic further, like "vote, Daschle, Lott, Bush," then the list of returned articles becomes an entirely manageable 14. Obviously, specificity is key to an efficient search.

Specificity needs can be met by questioning the searcher in detail. A list of questions is asked at the beginning of the query formulation process.

- What is the topic?
- Who is affected by it?
- Who benefits from it?
- Who can change it?
- What is its purpose?

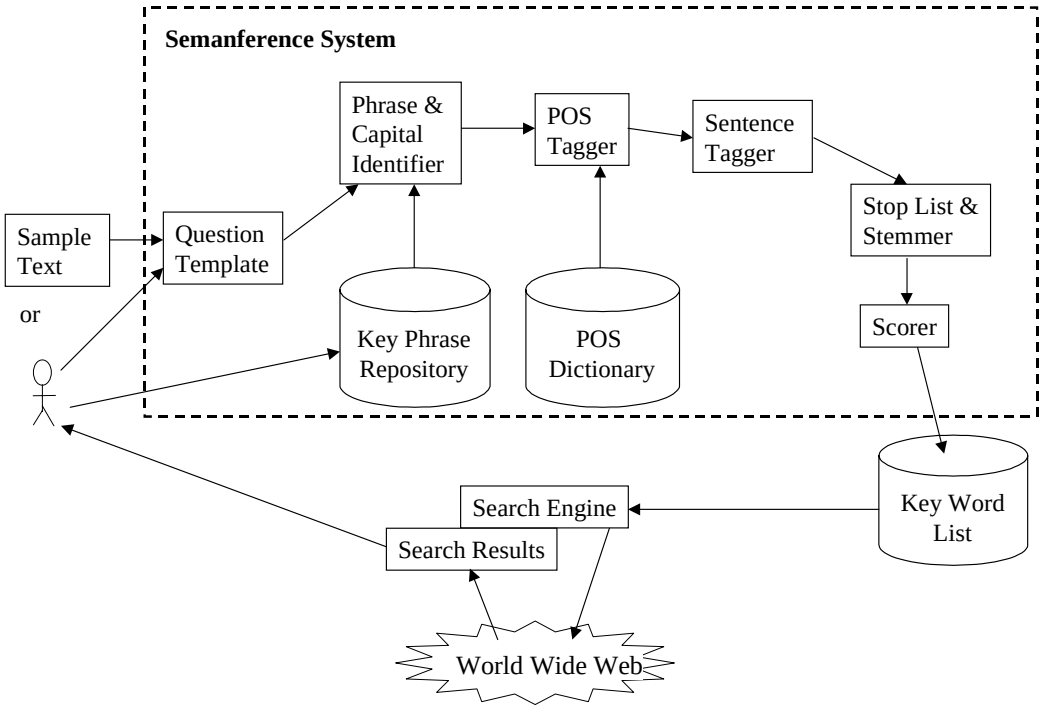

Figure 2 The Semanference System

- What phrases are used about it?
- Where does it take place?
- Who introduced it?
- Who wants this to occur?
- Who opposes it?
- What actions (verbs) are associated with it?
- What makes it necessary or worth researching?
- When was it begun?
- What holidays are associated with it?
- How current should the results be? Six months? A year? All results?

One important point is that the questions must be answered using complete sentences; in paragraph form answer as many questions as possible. This is necessary because basic English syntax is used as a tool by the Semanference System. Additionally, it provides the searcher with practice in topic analysis and written communication, so that larger educational goals are addressed. An alternative approach to answering questions is for the searcher to find and input a document, or portion of a document, to use as a sample text about which they want to know more.

Recognizing Key Phrases and Proper Names

Some phrases in English contain words which, when put together, yield changed meanings. These phrases must be recognized in their own right in order to accurately grasp the meaning of the text in which they occur. For example, the phrase "Senate majority" carries meaning beyond the words "Senate" and "majority." Similarly, "prescription medication" means more than either "prescription" or "medication." The first step in the Semanference System is to compare the input text to a database of such key phrases, identify them, and weld them into single semantic units which remain intact throughout processing. The next step is similar: all proper names are identified by capitalization, except for the first word of a sentence. Clustered capitalizations (for example, "Hillary Clinton" or "Empire State Building") are welded into single semantic units. Additionally, anagrams like "HMO" are identified. Thus, all proper names, key phrases, and anagrams act as single nouns as they continue processing. All numbers except dates are ignored; dates are tagged as proper nouns. Additionally, the System prepares hyphenated words for processing by removing their hyphens. Hyphenated words fall into two categories, place names like "Minneapolis-St. Paul" or multiword modifiers like "employer-sponsored." In both cases, removing the hyphens creates single units, which are processed according to their parts of speech. Finally, all apostrophes are removed from words and phrases. Their function, indicating possession, is irrelevant to the concept matches the System makes.

Part of Speech Tagging

The next step in the Semanference System is to tag all words in the statement by their parts of speech. In a common sentence, nouns are recognized as names for persons, places, or things. Verbs denote agency for nouns in the sentence; that is, they show the action, state of being, or change in state of being for the nouns in the sentence. Adjectives, which modify nouns, and adverbs, which modify verbs, adjectives, and other adverbs, can be identified according to these functions. A dictionary is used to identify these parts of speech, as well as articles, prepositions, interjections, and conjunctions.

However, because English uses words in multiple contexts and their meanings are context-dependent, a potential interpretive problem exists. For example, the English word "prompt" can be used in multiple ways. It can be an adjective meaning "punctual," as in "Be prompt." It can also be a noun synonymous with the word "hint," as in "The teacher gave the essay prompt to the student." Furthermore, it can be a verb meaning "to remind," as in "We prompt our students about assignment due dates." In addition to root word uses, "prompt" can be combined with the suffix "ly" to create the adverb "promptly." As a verb it can also take on past and present participial forms, as in "prompted" and "prompting," as well as indicate person and number, as in the third person singular expression "she prompts."

Therefore, in order to accurately parse meaning from a document containing the word "prompt," a method of determining part of speech must be devised. One relatively simple method is to introduce the word "prompt" to a series of if-then rules to determine its part of speech. For example, if it is preceded by an article, as in "the prompt," then its part of speech is "noun," even if additional words intervene, as in "the final prompt." Secondly, if it is preceded by a form of the verb "to be," as in "You must be prompt" or "He is prompt," then "prompt" is an adjective. Next, if in a given sentence the word "prompt" follows a noun or a pronoun but has *no article intervening* between the noun and the word itself, then "prompt" is a verb, as in "They often prompt each other in math class." With an "s" ending, the word becomes "prompts," which can be either a noun or a verb. This case necessitates two additional rules. First, if the word is preceded by an article and ends with the letter "s," then "prompts" is a noun, as in "Only a few prompts were given to the actor." And, secondly, if no article precedes "prompts," then it is a verb, as in "She sometimes prompts the children to be quiet." Additional usage rules for "prompt" are straightforward: if "ly" is attached to it, it is an adverb, and if "ed" or "ing" are attached to "prompt," then it is a

IF	Example	THEN POS testword is a
article testword	*the prompt*	noun
article (otherword) testword	*the final prompt*	noun
(otherword) tobe testword	*You must be prompt*	adjective
(otherword) tobe testword	*He is prompt*	adjective
pronoun (otherword) testword	*They often prompt*	verb
(otherword) article (otherword) testword + s	*Only a few prompts*	noun
(otherword) (otherword) testword+s	*She sometimes prompts*	verb
testword + ly	*promptly*	adverb
testword + ed	*prompted*	verb
testword + ing	*prompting*	verb

Table 1 Generalized Rules for Part of Speech Determination

verb. The rules (Table 1) for determining a word's part of speech (POS) are applied to each word in the narrative. The words are thus tagged for further processing.

Sentence Tagging

The Semanference System next tags each sentence according to its purpose; the set of tags used equates with traditional sentence purposes: declarative, imperative, and interrogative. Declarative sentences make statements, as in "Some students completed their assignments." Imperative sentences issue requests or commands, as in "Complete your assignment." Interrogative sentences ask questions, as in "Did all the students complete their assignments?" The parser can easily apply tags based on syntactical cues contained in a sentence. Basic sentence syntax in English is "subject-verb-object"; after POS tagging, the tag pattern is noun-verb-noun. Declarative and imperative sentences can be recognized using variations of this pattern. The parser can recognize the noun-verb-noun pattern as declarative, as in "Students completed assignments." It can also recognize the verb-noun pattern, which forms an imperative sentence, as in "Complete your assignment." Obviously, the question mark punctuating the end of an interrogative sentence cues the parser to its purpose. However, interrogative sentences are treated differently once they are identified: they are dropped from further processing because questions rarely add pertinent information to the text for a given topic. Rather, questions raise expectations that answers will follow them; the answers contain useful information structured in declarative sentences. Therefore, the information necessary to include in the end nodes of a query's inference network comes from the answering (declarative) sentence.

Just as a question mark cues the parser to the presence of a question, so quotation marks cue the parser to the presence of a quotation. Quotations offer information of secondary importance to a given topic. Usually, this is an interpretation of facts contained in nearby declarative sentences, as in a news story. On some occasions, a quotation forms the springboard from which an author launches their theory regarding the topic of the text, as in a review article or an article discussing theory in an academic journal. Because the content of such sentences is of secondary importance to the main content of the text, it will not significantly add precision to the search process; therefore, these sentences are eliminated from further processing.

Removing Stop List Entries and Stemming

The remaining text is compared to a stop list of articles ("a," "an," "the"), indefinite indicators (like "that" and "those"), prepositions, interjections, conjunctions, pronouns, auxiliary verbs (like "may" and "would"), and forms of the verb "to be" (like "is," "was," and "been"); words found on the list are deleted from processing. The remaining words are stemmed; that is, their number and tense indicators are removed.

Scoring

The isolated words and phrases resulting from this process are in effect tokens for the subject matter of the statement of information need. Importance is determined by scoring words according to their frequency of use. Key phrases and proper names and phrases are automatically assigned a score of "2" for their first instance of use, and an extra point for each additional instance. Additionally, the answer to the template question, "What is the topic?" and the titles of a sample text identify the overall topic of the narrative. These topic words closely reflect the narrative content; therefore each non-stop-listed topic word is automatically assigned an initial score of "3," with an additional point added for each use instance. All other words are weighted by counting their frequency of use. Additionally, the status of the keywords is retained. A keyword is assigned its status according to its highest status order of precedence (Table 2). An algorithm is applied to the results so that they are submitted to the selected search engine in an orderly fashion. Specifically, title words having a score greater than "3," those that also appear in the text, and key phrases with a score of "2" or higher, proper nouns with a score greater than "2," and other words with scores of "4" or more are used. A searcher's Web query posed with these words will then elicit responses that accurately reflect the user's intent.

Title Word	(TW)
Key Phrase	(KP)
Proper Name	(PN)
Other word	(OW)

Table 2 Key Word Order of Precedence

SEMANFERENCE AT WORK: A NEWS STORY

An example of this process demonstrates its utility. We use the alternate method of providing an information-need; part of the text of a news article was used as the sample text. Figure 3 displays the top two paragraphs of an Associated Press news story displayed at www.netscape.com June 26, 2001. For demonstration we apply the Semanference System to these paragraphs, which we refer to here as the Sample Text. This sample text answers many of the questions from the Question Template (Table 3).

Senators Reject HMO Suit Immunity

Senate Republicans failed Tuesday to win employers full immunity from workers' health care lawsuits as the Democratic majority protected core elements of its patients' rights bill.

The vote was 56-43 against an amendment by Sen. Phil Gramm, R-Texas. It would have granted employers full protection from lawsuits filed by workers or family members covered by employer-sponsored health insurance.

Figure 3 A News Story

What is the topic?	Senators Reject (HMO) Suit Immunity.
Who is affected by it?	Workers and Employers are affected.
Who benefits from it?	Employers benefit if they are protected from lawsuits.
Who can change it?	The Senate can change it.
What is its purpose?	The purpose is to protect workers.
What phrases are used about it?	Patient rights bill, Senate Republicans, Democratic Majority
Who introduced it?	Sen. Phil Gramm introduced a bill.
Who wants this to occur?	Senate Republicans want to pass Gramm's bill.
Who opposes it?	The Democratic Majority opposes the bill.
What actions (verbs) are associated with it?	Vote, pass, reject, win, lose

Table 3 Question Template applied to Sample Text

Key Phrase and Capital Identification

The text is compared with the phrases stored in the Key Phrases database. Phrases matching those in the database are bracketed so that they are treated as a single semantic unit by the System. The process identifies the numerical phrase "56-43"; this phrase is ignored. The next step in the process is identification of proper names. Excepting the first word of each sentence, all capitalized words are parenthesized. All adjoining, or clustered, capitalized words or anagrams are parenthesized as a single unit. Key phrases are bracketed and capitalizations are parenthesized.

Part of Speech Tagging

Once this step is completed, the Sample Text is compared to the POS dictionary, and the rules are applied assigning parts of speech to each word/phrase in the text. Figure 4 shows processing to this point; key phrases and capitalizations are labeled nouns.

N V N N N
Senators Reject (HMO) Suit Immunity

N V N Pre V Adj N Pre N N N Pre Art N
[Senate Republicans] failed (Tuesday) to win full immunity from workers [health care] lawsuits as the [Democratic majority]

V N N Pre Pro N
protected core elements of its [patients rights bill].

Art N V IGN Pre Art N Pre N Pro V V V N Adj N
The vote was [56-43] against an amendment by (Sen. Phil Gramm, R-Texas). It would have granted employers full protection

Pre N V Pre N Con N V Pre N N N
from lawsuits filed by workers or [family members] covered by employer sponsored [health insurance].

Figure 4 Key Phrases, Capitalization, and Part of Speech Tagged

```
    N      V     N     N     N
Senator Reject (HMO) Suit Immunity
          N             V        N       V  Adj     N          N          N          N               N
[Senate Republicans] fail (Tuesday)   win full immunity  worker [health care] lawsuit  [Democratic majority]
    V     N      N              N
protect core element   [patients' rights bill].

 N   INSIG        N                    N                      V      N      Adj      N
vote  [56-43]   amendment  (Sen. Phil Gramm, R-Texas).   grant employer full protection

  N     V      N             N           V         N         N           N
lawsuit file  worker  [family members] cover  employer sponsor [health insurance].
```

Figure 5 Stop List and Stemming Applied

Sentence Tagging

The sentences in the Sample Text are all declarative and so are all retained for further processing.

Stop List and Stemming

Once sentences are tagged, the stop list words are removed. All remaining single words are stemmed. Figure 5 shows processing to this point. Note that verbs like "filed" and "sponsored," after losing their tense indicators, retain their POS tags as verbs.

Scoring

The remaining individual words in the Sample Text are counted and assigned a score based on their frequency of use. For example, "full," "worker," and "lawsuit" are each found twice, and so are assigned a score of "2." Bracketed phrases are automatically assigned a score of "2." Words in the title score higher. For example, "immunity" appears in the title, thus earning a score of "3," plus an additional point for its use in the text, bringing it to a score of "4." Once scores are assigned, the words and phrases are ranked with highest scores first. Figure 6 shows these results.

Keyword	Score	POS	Status	Keyword	Score	POS	Status
immunity	4	N	TW	Democratic majority	2	N	KP
Senator	3	N	TW	patients rights bill	2	N	KP
reject	3	V	TW	health insurance	2	N	KP
HMO	3	N	TW	family members	2	Adj	KP
suit	3	N	TW	full	2	N	OW
employer	3	N	OW	worker	2	N	OW
Senate Republicans	2	N	KP	lawsuit	2	N	OW
health care	2	N	KP				

Figure 6 Ranked Keywords

Searching the Web

The algorithm is applied to these results so that they are in an orderly fashion. In this case, title words having a score greater than "3", and key phrases with a score of "2" are used. The following list is submitted to Yahoo!: "immunity," "Senate Republicans," "health care," "Democratic majority," "patients rights bill," "health insurance," and "family members." The Web search done November 29, 2001 yielded only one result, a Web page that contained the Associated Press news story (Figure 7).

Had a searcher developed their own search keywords for a search on the patient bill of rights and the Senate Republican position many more results would have been found. A Yahoo! search using the phrase "patients rights bill" returned 2690 results, an unusable high number. A second search added "health care" to the first phrase; it returned 1620 results. In neither case was the target news article found among the first 100 results. To narrow the results further, the phrase "Senate Republicans" was combined with the other two for a third search. It yielded 92 results, with the target content at the 19th position.

Although the results of the search are obviously too narrow for practical use, they prove the utility of the Semanference System. A particular news story is found after several months have passed, using only words and phrases processed from the original story to match to the newly found story.

Future Work

Often the major subject of a narrative is mentioned many times, but typically not with a repetition of the noun. This is of course the purpose of the pronoun. In the future it is hoped that the rules used to tag words with their part of speech will be improved, particularly the identification of pronouns with the corresponding noun or noun phrase. Additional points can then be scored by those important words.

Currently verbs are tagged. However, they do not satisfy the algorithm for selecting the keywords from the scored words. This is because most search engine document keywords are nouns. But, as ontological approaches to Web indexing become pervasive, documents will be associated by verbs and verb phrases. The Semanference System will be able to provide necessary search guidance.

Because of the complexity of the English language, the complete range of usage for some words cannot be captured in a set of rules like those just enumerated. For example, consider the use of "prompts" in the following sentence: "Few prompts were given to the actor." Despite the fact that an article does not precede it, it is a noun in this sentence. Given the rules above, the parser may inaccurately identify it as a verb. But because, later in the Semanference System, the word "prompts" will be reduced to its root "prompt" and then weighted according to its number of occurrences, this error will have minimal impact on search results.

Conclusion

The Semanference System is designed to improve the query processes used to request documents from the World Wide Web. It processes a text of the user's choice so that its words and phrases are weighted according to their frequency of use in the text. The ranked results of this process are words and phrases which, when used as key words and phrases with a search engine, yield results superior in specificity and textual accuracy to those of current methods. In other words, they yield a document list that better matches the user's information needs.

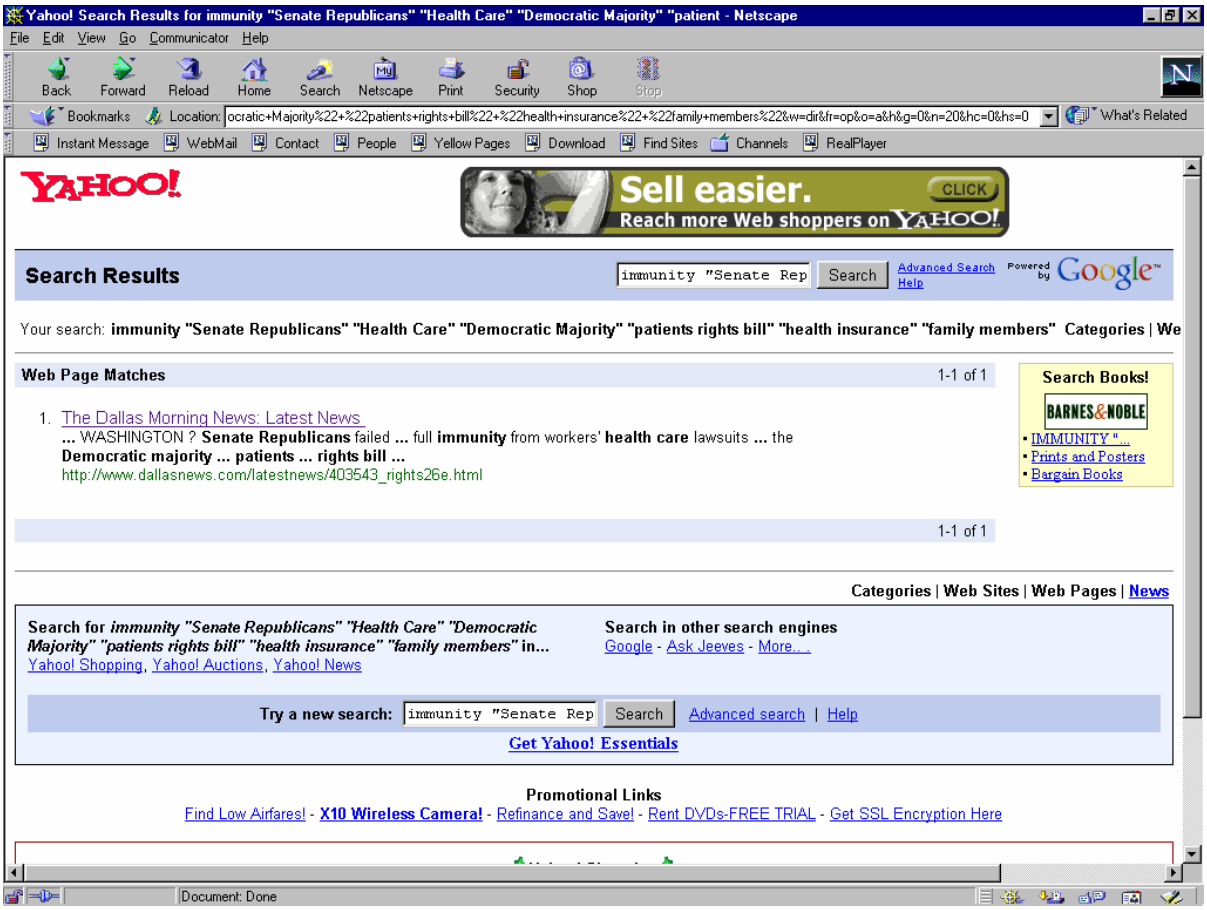

Figure 7 Results of Yahoo! Search

References

Adelberg, B. (1998). NoDoSE—A Tool For Semi-Automatically Extracting Structured and Semistructurted Data from Text documents. *Proceedings of ACM SIGMOD International Conference of Management of Data*, 283-94.

Christiansen, Morten and Chater, Nick(1999). Connectionist Natural Language Processing: The State of the Art. *Cognitive Science*, 23(4), 417-37.

Han, H. and Elmasri, R. (2001). Analysing Semi-structured Web Pages For Ontological Information Extraction. *Proceedings of the International Conference on Internet Computing (IC'2001)*, Las Vegas, Nevada, June 2001, 21-27.

Iatsko, V. (2001). Linguistic Aspects of Summarization. *Philologie im Netz* 18, 33-46.

LaForest, F., and Flory, A. (2001). Using Weakly Structured Documents to Fill a Classical Database. *Journal of Database Management* 12 (2), 3-13.

Lin, S., Shih, C., Chen, M., Ho, J., Ko, M., and Huang, Y. (1998). Extracting Classification Knowledge of Internet Documents with Mining Term Associations: a Semantic Approach. *Proceedings of the 21st Annual Internation Conference on Research and Development in Information Retrieval*, 241-49.

Losee, Robert M. (2001). Natural Language Processing in Support of Decision Making: Phrases and Part-of-Speech Tagging. *Information Processing and Management* 37, 769-87.

Lucas, W., Schiano, W., and Crosett, K. (2001). The Present and Future of Internet Search. *Communications of the Association for Information Systems* 5(8).

Martinez-Trinidad, J. F., Beltran-Martinez, B., and Ruiz-Shulcloper, J. (2000). A Tool to Discover the Main Themes in a Spanish or English Document. *Expert Systems with Applications* 19, 319-27.

Perez-Carballo, Jose and Strzalkowski, Tomek (2000). Natural Language Information Retrieval: Progress Report. *Information Processing and Management* 36, 155-78.

Turtle, Howard R., and Croft, W. Bruce (1996). Uncertainty in Information Retrieval Systems; in Uncertainty Management in Information Systems From Needs to Solutions, ed. by Motro, A. and Smets, P.; Boston; Kluwer Academic Publishers.

Yi, J., and Sundaresan, N. (1999). Mining the Web for Acronyms Using the Duality of Patterns and Relations. *WIDM99*, 48-52.

Biographies

Anthony Scime is currently an Assistant Professor of Computer Science at the State University of New York College at Brockport. His interests include the World Wide Web as an information system for the creation, discovery, storage, and dissemination of knowledge. He has over twenty years of academic, industry and government experience.

Colleen Powderly has degrees in English and Computer Science. Her interests include computational linguistics, particularly the application of algorithms to English syntax and semantics. She has worked as a teacher, writer, editor, and electronic publisher. She is also a poet.