A WordNet-Based Interface to Internet Search Engines

Dan I. Moldovan and Rada Mihalcea
Department of Computer Science and Engineering
Southern Methodist University
Dallas, Texas, 75275-0122
{moldovan, rada}@seas.smu.edu

Abstract

A vast amount of information is available on the Internet, and naturally, many information gathering tools have been developed. Several search engines with different characteristics, such as AltaVista, Lycos, Infoseek, and others are available. However, the web information retrieval technology is still in its infancy, and there is need for considerable improvement. Some inherent difficulties are: (1) the web information is diverse and highly unstructured, (2) the size of information is large and it grows at an exponential rate, and (3) the current search engine technology is still rudimentary. While the first two issues are more profound and require long term solutions, it may be possible to develop software around the search engines to improve the quality of the information retrieved. In this paper we present a natural language interface system to a search engine and discuss some of the results obtained.

Introduction

A main problem with the current search engines is the large volume of documents extracted as a result of broad, general queries, and the luck of output produced to specific, narrow questions (Selberg and Etzioni 1995), (Zorn, Emanoil and Marshall 1996). A simple query can return anywhere from none to several million documents. This is called the precision or the relevance problem. In information retrieval, the *precision* is defined as the ratio between the number of relevant documents retrieved over the total number of documents retrieved.

The search engines identify documents by matching words or combination of words to document contents. AltaVista (Altavista) is one of the best known search engines that automatically tracks each word on each web page that it can find, down-loads the page and builds an index. Often, the boolean operators AND and OR are too week to identify relevant documents; that is, while many documents may be identified, the top ranked ones are not relevant to that query. On the other hand, the NEAR operator is too restrictive and excludes useful documents. Thus, short of using a more sophisticated natural language processing of documents, an area of improvement may be to design new retrieval operators that retain more relevant documents. A possible approach along this direction is still to use the existent search engines, which were developed with great efforts, and to post-process the set of documents produced to a query.

Another aspect that needs attention is to bridge the gap between the human questions and the simple query format that search engines take. When we want information, we think in terms of questions that are far more complex than simple words or combinations of words currently accepted by the search engines. One may ask:

`Q1: ``Who were the US Presidents of the last century?''` , or

`Q2: ``I want to know who was the 10th President of the United States, his religion and where was he borne''`.

This calls for a natural language interface that transforms sentences into queries with boolean operators currently accepted by the search engines. For many applications, such an interface does not have to perform a complex parsing and a deep semantic analysis of the input sentence. It may be sufficient to recognize the main concepts by performing a shallow linguistic processing. However, one thing that is highly beneficial is to search not only for the words that occur in the input sentence, but to create **similarity lists** with words from on-line dictionaries that have the same meaning as the input words. This can significantly broaden the web search. While it may seem counterproductive to our effort of filtering the documents, by broadening the search we increase the *recall*, defined as the ratio between the number of relevant documents extracted over the total number of relevant documents in the database.

In this paper we examine some of the benefits of using Natural Language Processing in conjunction with WordNet, an on-line lexical database developed at Princeton University (Miller 1995), to improve the quality of the results. Specifically, the paper describes a system that addresses two issues: (1) translates a natural language question or sentence into several simple queries, and (2) processes the documents fetched by the search engines to filter them and extracts the paragraphs that render relevant information. The first step is intended to increase the recall while the second one increases the precision.

System architecture

The system architecture is shown in Figure 1. An input query or sentence expressed in English is first presented to the lexical processing module that extracts the keywords from the sentence. The query formation module uses these keywords to form queries that are sent to one or more search engines. The documents fetched by the search engines are filtered with the help of some new operators.

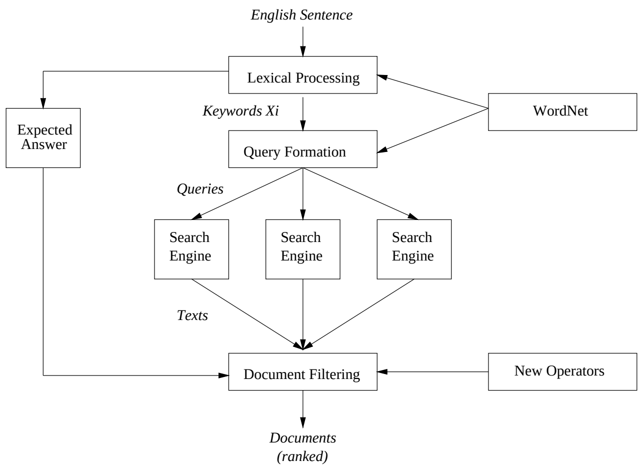

Figure 1: System organization

Lexical processing

This module has been adopted from an information extraction system developed by us for the MUC competition (Moldovan et al. 1993). First the sentence boundaries are located. It then places part of speech tags on words by using a version of Brill's tagger (Brill 1992) in conjunction with WordNet. It also contains a phrase parser that segments each sentence into constituent noun and verb phrases and recognizes the head words. After the elimination of conjunctions, prepositions, pronouns and modal verbs we are left with some keywords x_i that represent the important concepts of the input sentence.

WordNet

The WordNet lexical dictionary developed at Princeton University (Miller 1995) is used for part of speech tagging and word sense disambiguation. WordNet 1.6 contains 126,520 English words grouped into 91,595 synonym sets, called synsets. Words and synsets are connected by 391,885 lexico-semantic relations, making WordNet a useful resource for natural language processing.

For example, WordNet lists the concept {`cat, true cat`} with the gloss: (`feline mammal usually having thick soft fur and being unable to roar; domestic cats; wildcats`). From our world experience, we realize that this definition does not cover all we know about cats. WordNet provides additional information about the concept {`cat, true cat`} by its hypernymes and their glosses and also by the glosses of other concepts that use {`cat, true cat`} as a defining concept. Examples of these are:
{`mouser`}, gloss: (`a cat proficient at mousing`)
{`meow, mew,miaou, miaow`}, gloss: (`the sound made by a cat`)
{`caterwaul`}, gloss: (`the yowling sound made by a cat in heat`)
{`pur`}, gloss: (`a low vibrating sound typical of a contented cat`).

In WordNet the concept {`cat, true cat`} is related to 215 other concepts (10 from its gloss, 38 from the glosses of its hypernyms, 25 concepts that use it in their glosses as a defining concept plus 142 concepts with which the concept interacts in these 25 glosses). This information is semantically rich enough to presume that WordNet can act as a powerful knowledge base for information extraction.

Word-sense disambiguation

Our idea for word sense disambiguation is to use WordNet to determine the possible association between words. In WordNet each concept has a gloss, an explanation in English of the meaning of that concept. Given a pair of words, the algorithm identifies the most likely combinations of their senses by measuring the conceptual distance between them. The conceptual distance is determined by counting the number of common words that are semantically associated with the senses of the two words in the pair. (Li, Szpakovicz and Matwin 1995) and others have demonstrated that it is possible to use machine readable dictionaries instead of large corpora to gather statistics for the senses of noun-verb pairs. We have tested our disambiguation method against the semantic annotations from SemCor, and the results showed that in 80% of the cases the correct result as indicated by SemCor was in the top four choices of the ranked list of possible pairs of the two words senses (for polycemous words many combinations are possible). These results are described in (Mihalcea and Moldovan 1998).

An example

The system operation is presented below with the help of an example. Suppose one wants to find the answer to the question: `''How much tax an average salary person pays in the United States?''` This question is ambiguous since the word `tax` is too broad for such a specific question. With the help of WordNet, the system asks back the user to select from the hyponyms of concept `tax` (concepts that are more specific and subsumed by concept `tax`). The choices are: `single tax, income tax, capital gains tax, capital levy, inheritance tax, estate tax, death tax, death duty, direct tax, indirect tax, capitation, rate, stamp tax, stamp duty, surtax, supertax, pavage, special assessment, duty, tariff, excise, excise tax, gasoline tax.`

The user clicks on `income tax` and the new question becomes: `''How much income tax an average salary person pays in the United States?''` The linguistic processing module identified the following keywords:

x_1 =(`income tax`), pos = noun, sense #1/1
x_2 =(`average`), pos = adjective, sense #4/5
x_3 =(`salary`), pos = noun, sense #1/1
x_4 =(`the United States`), pos = noun, sense #1/2
x_5 =(`person`), pos = noun, sense #1/3
x_6 =(`pays`), pos = verb, sense #1/7

In the notation above "pos" means part of speech, and the sense number indicates the actual WordNet sense that resulted from the disambiguation out of all possible senses in WordNet. For instance adjective `average` has 5 senses and the system picked sense #4. Note that the senses of words in WordNet are ranked in the order of their utilization frequency in a large corpora.

Query formulation

The two main functions performed by this module are: 1) the construction of similarity lists using WordNet, and 2) the actual query formation.

Once we know the sense of each word in the input sentence, it is relatively easy to use the rich semantic information contained in WordNet to identify many other words that are semantically similar to a given input word. By doing this we increase the chance of finding more answers to input queries. WordNet can provide semantic similarity between concepts at various levels. Here are three levels that may be considered in descending order of interest.

Level 1. Words are semantically similar if they belong to the same synset.

Level 2. Words that express concepts linked by semantic relations; i.e. hyponymy/ hypernymy, meronymy/ holonymy and entailment.

Level 3. Words that belong to sibling concepts; namely concepts that are subsumed by the same concept.

Let's denote with x_i the words of a question or sentence, and with W_i the similarity lists provided by WordNet for each word x_i. In our example considering only Level 1 similarity, and only the first four keywords, WordNet provides:

W_1 = {`income tax`}
W_2 = {`average, intermediate, medium, middle`}
W_3 = {`salary, wage, pay, earnings, remuneration`}
W_4 = {`United States, United States of America, America, US, U.S., USA, U.S.A.`}

These lists are used to formulate queries for the search engine. As we will see, the operators available today for the search engines are not adequate to provide the desired answers in most of the cases. Table 1 shows some queries and the number of documents provided by AltaVista, considered to be one of the search engines with the most powerful set of operators available today.

	Query	Number of documents
1	W_1 AND W_2 AND W_3 AND W_4	15,464
2	W_1 AND (W_2 NEAR W_3) AND W_4	3,217
3	W_1 NEAR (W_2 NEAR W_3) AND W_4	803
4	W_1 NEAR W_2 NEAR W_3 NEAR W_4	0
5	W_1 AND {`average` W_3} AND W_4	1752
6	W_1 AND {`average` W_3} NEAR W_4	1 (no)
7	W_1 NEAR {`average` W_3} NEAR W_4	0

Table 1: Queries with various combinations of operators

The ranking provided by the Alta Vista is of no use for us here. None of the leading documents in any category provides the desired information. The only document fetched by Query 6 is equally irrelevant:

....Instead, their plans would shift more of the total tax burden on to labor, taxing capital once under a business tax, and taxing wages and salaries twice under both the income tax and the payroll tax. Middle-class Americans have to pay more under such a system, and wealthy people much less....

....The average taxpayer must work 86 days to pay all federal taxes, and must work 36 days just to pay his or her federal income tax. The average American must work 2 hours and 49 minutes every working day to pay all their taxes.....

An analysis of the results in table above indicates that there is a gap in the volume of documents retrieved with the Alta Vista operators. For instance using only the AND operator (Query 1) 15,464 documents were obtained, but the NEAR operator (Query 4) produced no output. This operator seems to be too restrictive, while it fails to identify the right answer. Various combinations of AND and NEAR operators were tried, as indicated by the table above with no great results.

The conclusion so far is that the documents containing the answers, if any, must be among the large number of documents provided by the AND operators. However, the search engines failed to rank them in the top of the list. Thus, we sought to find new operators that filtered out many of the irrelevant texts.

Question	AND x_i	NEAR x_i	AND w_i	NEAR w_i	PARAGRAPH w_i	SENTENCE w_i
Where is the name ABBA coming from?	1381 no	0	1964 no	0	1 yes	1 yes
What is the frequency used for electronic products in the United States?	18687 no	0	44530 no	2 1 yes	1 yes	0
Who suggested for the first time that the Earth moves around the Sun?	45387 no	0	85785 no	0	26 3 yes	0
What is the meaning of the harp symbol on the Guinness merchandise?	28 no	0	97 no	4 2 yes	6 2 yes	0
How much tax an average salary person pays in the United States?	8819 no	0	15464 no	0	1 yes	0

Table 2: Sumary of results

Storage of documents

The program sends a query to AltaVista and takes the URL addresses of the first 1000 documents that match the query (this is a current limitation of AltaVista). The documents are down-loaded and processed. The HTML tags are removed and the documents loaded on the disk as text files.

New operators

Our approach to filtering documents is to first search the Internet using weak operators (AND, OR) and then to further search this large number of documents using more powerful operators. For this second phase, we propose the following additional operators:

PARAGRAPH n (... similarity lists ...)
The PARAGRAPH operator searches like an AND operator for the words in the similarity lists with the constraint that the words belong only to some n paragraphs. The rationale is that most likely the information requested is found in a few paragraphs rather than being dispersed over an entire document. A similar idea can be found in (Callan 1994).

SENTENCE n (... similarity lists ...)
The SENTENCE operator searches like an AND operator for the words in the similarity lists with the constraint that the words belong to a sentence. The answers to many queries are found in single, sometimes complex sentences.

SEQUENCE ($W_1 x W_2 x .... W_n$)
where x is a numeric variable that indicates the distance between the words in the W lists for which the search is done. The SEQUENCE operator imposes a more flexible NEAR search, but it requires that the sequence of the words in the similarity list be maintained as specified. Of course, combinations of these operators are possible.

Using the PARAGRAPH operator for the example above, the system found a relevant answer:

In 1910, American workers paid no income tax. In 1995, a worker earning an average wage of \$26,000 pays about 24% (about \$6,000) in income taxes. The average American worker's pay has risen greatly since 1910. Then, the average worker earned about \$600 per year. Today, the figure is \$26,000.

Evaluation of Results

The system has been tested on five randomly selected questions. Table 2 summarizes our results. The AND x_i column indicates the number of documents extracted by AltaVista when only AND operator was applied to input words x_i. Interestingly, no relevant answers were found in the top ten documents in any of these searches. Using only the NEAR operator applied to input words x_i produced no result in any case.

By replacing the words x_i with their similarity lists derived from WordNet, the number of documents retrieved was increased, as expected. Also, no relevant documents were found in the top ten ranked documents in any of the searches. However, by applying the NEAR operator to words in the similarity lists relevant answers were found for two questions.

The next column contains the number of documents extracted when the new operator PARAGRAPH 2 (meaning two consecutive paragraphs) was applied to words from the similarity lists. The results were encouraging. The number of documents retrieved was small and correct answers were found in all cases. The number near "yes" indicates how many documents contained the desired answer in each case. The last column shows that the operator SENTENCE was two restrictive, producing corect answer in only one out of five cases.

Conclusions

This paper has introduced the idea of using WordNet to extend the search based on semantic similarity. The example clearly shows that without this it was not possible to find an answer. Then, we have introduced some new operators that fill the gap between the operators currently used by the search engines.

The broad use of natural language queries in information retrieval is still beyond the capabilities of current natural language technology. Machine readable dictionaries, such as WordNet, prove to be useful tools to web search. However, their use for the Internet has been limited so far (Allen 1997), (Hearst, Karger and Pedersen 1995), (Katz 1997).

There are several other posible ways of improving the web search not discussed in this paper. One such a possibility is to index words by their WordNet senses. This of course implies some on-line word-sense disambiguation of documents which may be possible in not too distant future. Semantic indexing has the potential of improving the ranking of search results, as well as to allow information extraction of objects and their relationships (Pustejovsky et al. 1997).

Another way to improve the web search is to use compound nouns or collocations. In WordNet there are thousands of groups of words such as blue color worker, stock market, etc , that point to their respective concept. Each compound noun is better indexed as one term. This reduces the storage space for the search engine and may increase the precision.

References

AltaVista. *Digital Equipment Corporation.* AltaVista Home Page http://www.altavista.digital.com.

Allen, B.P. 1997. WordWeb - Using the Lexicon for WWW. *Inference Corporation* http://www.inference.com.

Brill, E. 1992. A simple rule-based part of speech tagger. In Proceedings of the 3rd Conference on Applied Natural Language Processing, Trento, Italy.

Burke, R.; Hammond, K. and Kozlovsky J. 1995. Knowledge-based Information Retrieval from Semi-Structured Text. In Proceedings of the American Association for Artifical Intelligence Conference, Fall Symmposium, "AI Applications in Knowledge Navigation & Retrieval", 15-19, Cambridge, MA.

Callan, J.P. 1994. Passage-Level Evidence in Document Retrieval. In Proceedings of the 17th Annual International ACM SIGIR, Conference on Research and Development in Information Retrieval, 302-310, Dublin, Ireland.

Hearst, M.A.; Karger D.R. and Pedersen, J.O. 1995. Scatter/Gather as a Tool for the Navigation of Retrieval Results. In Proceedings of the American Association for Artifical Intelligence Conference, Fall Symposium "AI Applications in Knowledge Navigation & Retrieval", 65-71, Cambridge, MA.

Katz, B. 1997. From Sentence Processing to Information Access on the World Wide Web, In Proceedings of the American Association for Artifical Intelligence Conference, Spring Symposium, "NLP for WWW", 77-86, Stanford University, CA.

Leong, H.; Kapur, S. and de Vel, O. 1997. Text Summarisation for Knowledge Filtering Agents in Distributed Heterogeneous Environments. In Proceedings of the American Association for Artifical Intelligence Conference, Spring Symposium, "NLP for WWW", 87-94, Stanford University, CA.

Li, X.; Szpakowicz, S. and Matwin, S. 1995. A WordNet-based Algorithm for Word Sense Disambiguation, In Proceedings of the 14th International Joint Conference on Artificial Intelligence, 1368-1374, Montreal.

Mihalcea, R. and Moldovan, D.I. 1998. Measuring the Conceptual Distance between Words, Technical Report, Department of Computer Science and Engineering, Southern Methodist University, Dallas, TX.

Miller, G.A. 1995. WordNet: A Lexical Database. Communication of the ACM, 38(11):39-41.

Moldovan, D. et al. 1993. USC: Description of the SNAP System Used for MUC-5. In Proceedings of the 5th Message Understanding Conference, Baltimore, MD.

Pustejovsky, J.; Boguraev B., Verhagen, M.; Buitelaar, P. and Johnston, M.1997. Semantic Indexing and Typed Hyperlinking. In Proceedings of the American Association for Artifical Intelligence Conference, Spring Symposium, "NLP for WWW", 120-128. Stanford University, CA.

Selberg, E. and Etzioni, O. 1995. Multi-Service Search and Comparison Using the MetaCrawler. In Proceedings of the 4th International World Wide Web Conference, 195-208, Boston, MA.

Velez, B.; Weiss, R.; Sheldon, M.A. and Gifford, D.K. 1997. Fast and Effective Query Refinement. In Proceedings of the 20th ACM Conference on Research and Development in Information Retrieval SIGIR, 6-15, Philapdelphia, PA.

Zorn, P.; Emanoil, M. and Marshall, L. 1996. Advanced Searching: Tricks of the Trade. *Online. The Magazine of Online Information Systems* 20(3).