



Technical Overview of Cognition's Semantic NLP™ (as Applied to Search)

Kathleen Dahlgren, Ph.D.

Growth of the Internet, the proliferation of email as the preferred method of information exchange, and the creation of huge stores of digitized text have opened the gateway to a deluge of information that is often difficult to navigate and search. Users are awash in data – unable to glean relevant information.

To address this need and pain-point, Cognition Technologies, Inc. has introduced Cognition's Semantic NLP™ (Natural Language Processing). This evolutionary software uses state-of-the-art *linguistic* technology to easily and precisely find on-target information on the Internet or in large libraries of digitized text. Users pose queries in plain English and Cognition's Semantic NLP interprets their meaning -- responding with *more precise results* than is possible with traditional search technologies (e.g. pattern matching, concept search, etc.). Cognition's Semantic NLP produces results which are both *highly relevant* to the user and very complete. This increased relevancy and completeness (precision and recall) is much higher than is possible with traditional Search technologies no matter how the user query is worded.

(Please refer to the glossary in the Appendix while reading this Technical Overview.)

I. Search Technology

Most search engines in use today are frustrating to use because they yield a large quantity of irrelevant information. Paradoxically, they also fail to retrieve significant amounts of relevant information. Current search technologies only work well when the user knows exactly how the information in the target documents is worded, and forms a search query with sufficiently fine granularity to yield a manageable amount of information. It is impossible to know how to word a query in advance (it would require the user to know the answer to the query as the query was constructed), so typically users spend a lot of time browsing irrelevant information, constructing ever more complex Boolean queries with only marginal success, or face the frustration of finding nothing at all.

Cognition's Semantic NLP is substantially more precise and exhaustive in its ability to search a dataset, as indicated by commonly used Precision/Recall tests. **Precision** is a measure of retrieval accuracy calculated by dividing the total number of relevant retrievals by the number of all retrievals generated by the search. **Recall** is a measure of the extent to which relevant material in the total document base is found. It is calculated by dividing the number of relevant retrievals by the total number of potentially relevant retrievals in the document base. One

comparative benchmark for measuring Precision and Recall is through a government-sponsored competition known as TREC, which is intended to test Search technologies. In 2006, the best performer in bioinformatics had 16% precision and 26% recall. Cognition's Semantic NLP has performed a number of internal head-to-head comparisons of search comparing Cognition's Semantic NLP with other well-known Search technologies on the same databases with the same queries. Document sets searchable by all the Search technologies were not available, so the head-to-head comparisons were performed on different document sets, but in each case Cognition's Semantic NLP and the competitor Search engine were searching in the identical documents with identical queries. Fifty queries considered likely to be entered by users were formed, searches performed, and result relevancy judged by members of the Cognition Technologies staff. Precision/Recall results were then compared.

Recall for Cognition's Semantic NLP and other Search engines was measured by taking all the relevant retrievals found as a baseline of 100% and comparing the Completeness of each Search engine to that baseline. There may have been other retrievals missed, but none was observed. Note that Cognition's Semantic NLP far out-performs the competitors in both precision and recall.

Search Engine	Document Base	Precision	Recall
Google	globalissues.com (blog)	12%	21%
<i>Cognition's Semantic NLP</i>	<i>globalissues.com (blog)</i>	91%	90%
dtSearch	Microsoft Anti-Trust Case emails	24%	19%
<i>Cognition's Semantic NLP</i>	<i>Microsoft Anti-Trust Case emails</i>	96%	95%
Autonomy	NewYorkLife.com (corporate Website)	1%	40%
<i>Cognition's Semantic NLP</i>	<i>NewYorkLife.com (corporate Website)</i>	92%	87%

II. Cognition's Semantic NLP Linguistic Technology

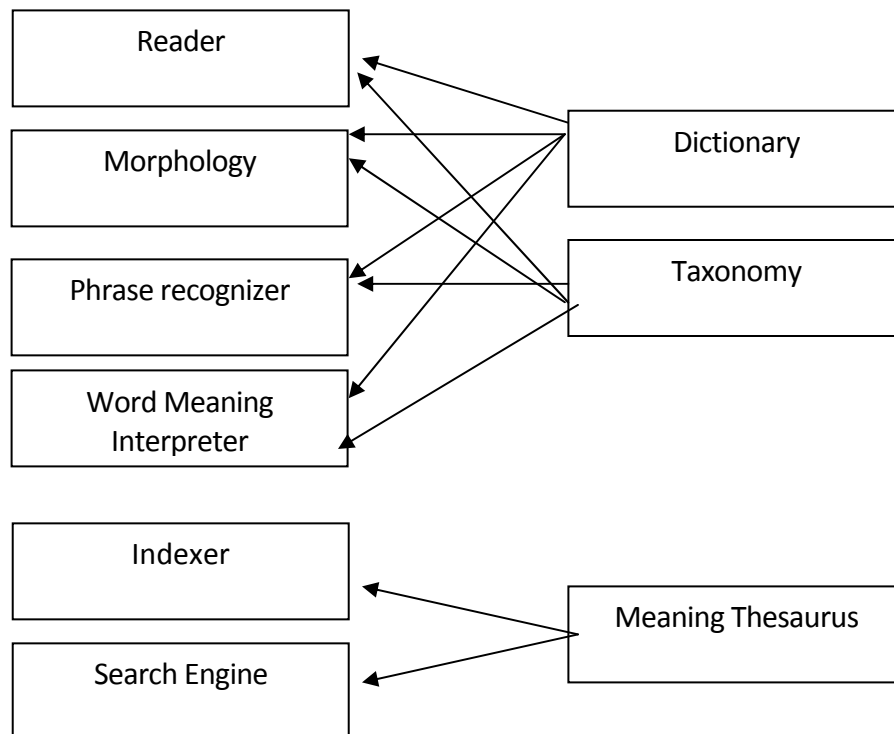
Cognition's Semantic NLP searches on meanings, not patterns, therefore, its results are very precise. The user poses queries in plain English, and Cognition's Semantic NLP determines what the words in the query mean in the context of the query. If you ask "How can I buy stock on the market?", Cognition's Semantic NLP determines that "stock" means "share" or "security". It searches only on that meaning of "stock" and doesn't retrieve information about stocking shelves, cattle or flowers. If you ask "How can I stock the shelves of my market?", it retrieves information about merchandising, and doesn't retrieve information about shares in companies. Cognition's Semantic NLP returns information with over 90% Relevancy, reducing the user's need to ponder large numbers of irrelevant retrievals found with other Search technologies.

Simultaneously, Cognition’s Semantic NLP overcomes the problem of information underload, i.e. not finding anything at all because of differences in wording. It finds information regardless of the way a concept is worded in the target documents. If you ask “Fatal fumes in the workplace?”, Cognition’s Semantic NLP finds documents that talk about “gas”, “vapor”, “steam”, etc. It is important to note that some of these words have ambiguous meanings (e.g. “fume” can mean a vapor or it can mean an aromatic wine), but Cognition’s Semantic NLP doesn’t retrieve irrelevant information triggered by those words used in a different meaning than the query. For example, when searching on “fume”, it retrieves to “gas” meaning vapor, but not “gas” meaning gasoline. The result is that Cognition’s Semantic NLP retrieves 5 to 7 times more relevant information than other Search technologies, as measured in head-to-head comparisons with other Search engines, while maintaining over 90% Relevancy.

Another source of greater Completeness is Cognition’s Semantic NLP taxonomy, which enables Cognition’s Semantic NLP to search on specific information when queried on more general information. As an example, if the user searches on “money”, Cognition’s Semantic NLP will find information about “dollar”, “pound” and “yen”, etc. Cognition’s Semantic NLP taxonomy covers 506,000 concepts, and is thus very complete. The customer doesn’t have to build a taxonomy from scratch, as with some technologies.

Cognition’s Semantic NLP Architecture

The components of linguistic processing in Cognition include a reader, a phrase parser, a morphological component, a word meaning interpreter, a dictionary, a taxonomy and a meaning thesaurus. The dictionary and taxonomy are used by the phrase parser, morphology and word meaning interpreter. The meaning thesaurus is used to find alternate wordings during search.



1. **The reader.** The reader reads the text or query, locates the words, and looks them up in the dictionary. This component guarantees against false hits when one word is part of another word, as in “part” and “party”, or “loss” and “floss”.

2. **The morphological component.** The morphological component isolates word stems from prefixes and suffixes, enabling Cognition’s Semantic NLP to recognize many millions of word forms (actually, an indefinite number). Some words take various forms according to morphological rules. Some of these are enumerated below:
 - a. Nouns with irregular plural morphology such as “mouse-mice”, “tooth-teeth”.
 - b. Nouns with regular changes in the plural such as “baby-babies”.
 - c. Regular verb inflections such as “raze – razed – razing” and “ship – shipped – shipping”.
 - d. Verbs with irregular past tense forms such as “catch – caught – catching”.
 - e. Regular derived forms:

inter-	communicate	intercommunicate
-tion	communicate	communication
inter- + -tion	communicate	intercommunication
-ize	actual	actualize
re-	marry	remarry

3. **The Phrase Recognizer.** The phrase recognizer combines words into phrases for more accurate interpretation. There are several different types of phrases it handles.
 - a. Names. The phrase parser recognizes that certain patterns are personal names, company names or places, whether or not those names are represented in the lexicon. The result is the ability to map from one form to another, including regular short forms. Examples are:
 - Mr. Savi Samdi - short form Mr. Samdi , Samdi is a human male
 - Lake Su – short form Su, is a lake
 - The XYZ Corporation – short form XYZ Corp. or XYZ, is a company
 - b. Dates. The phrase recognizer sees all date variations, and maps one to the other, as in
 - December 1, 1992
 - 12/1/92
 - 12-1-92
 - Dec. 1, '92.

- c. **Compounds.** The phrase recognizer interprets lexical phrases such as “movie set”, “heart attack”, “net revenue”, etc. Such phrases can consist of many words. There are currently over 191,000 compound phrases in the lexicon.
- d. **Acronyms.** The phrase recognizer maps the long form of acronyms to the short form, noting that “Securities and Exchange Commission” is the long form of “SEC”.
- e. **Idioms.** The phrase recognizer pieces together idioms, including morphological variants of them (which are linked to other word meanings in the meaning thesaurus). Some examples are:
 - kick the bucket – kicked the bucket (“die, sense1”)
 - let the cat out of the bag – letting the cat out of the bag (“disclose, sense 2”)

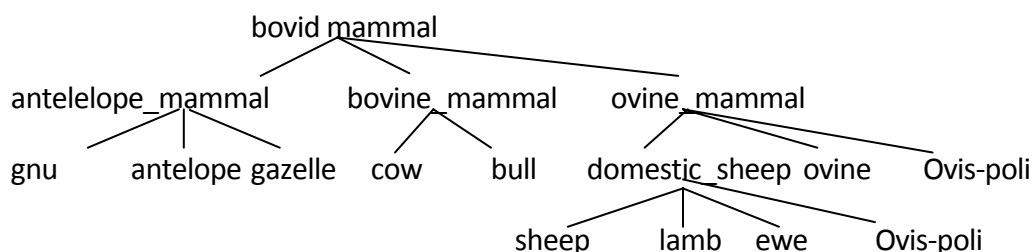
4. Word Meaning Interpreter. The word meaning interpreter uses context and structure to determine the meanings of words in context. Several databases are consulted to determine word meanings. For example, the word “check” in “check-up” is interpreted as part of phrase with “up” with a specific meaning, while “check” in “pay with a check” is interpreted as an individual word meaning a promissory note because of the context with “pay”. The lexicon contains 17,000 ambiguous words.

5. Word Sense Selection Database. This database encodes trigger information and statistical occurrence metrics used to contribute to contextual word sense selection. There are over 4 million such meaning contexts.

6. Dictionary. In the dictionary, each meaning of each word is defined, and given morphological, syntactic, taxonomic and semantic features. This information enables the software to select word meanings, recognize various forms of a given word, and parse sentences.

Cognition’s Semantic NLP lexicon is quite broad, including 506,000 forms, over 536,000 concepts, and millions of word forms. It has entries for almost every common word of English, and tens of thousands of proper names, phrases and acronyms. In combination with the morphology component, it recognizes millions of word forms. It has vocabulary in many domains including law, health and medicine, biology, genomics, finance, terrorism, recreation, household, human resources, encyclopedia articles, nuclear energy, software technical notes, newspaper articles, government regulations, telecommunications, human factors engineering, and military.

7. Taxonomy. Cognition’s Semantic NLP taxonomy classifies all objects and events in an inheritance hierarchy. An abbreviated piece of the taxonomy is shown below:



There are approximately 7,000 unique non-terminal nodes and 536,000 unique leaves or word senses.

- 8. Meaning Thesaurus.** The meaning thesaurus maps meanings to each other, forming meaning classes. Unlike ordinary thesauri, the mapping is from word meaning to word meaning (or phrase). The meanings within a grouping are judged to be loosely synonymous. The synonymy is "loose" in the sense that related meanings may have different parts of speech, but they evoke the same ideas in the mind of the reader. For example, the following concepts are in one thesaural grouping:

bank9, column2, file3, line3, queue1, rank4, row1, tier1, alignment1, align1

The digits indicate the specific senses or meanings of the words. In this example, bank9 means "a set of similar things", "column2" means "a line of similar things", "file3" means "a lined-up group of things", and so on.

The word "support" illustrates that a given word in different meanings may belong to a number of different meaning classes or thesaural groups, as shown below:

- support1, abet1, assist1, bail3, sponsor1, benefit1, benefit2, assistance1
- support2, attest1, back7, affirm3, establish2, prove1
- support3, bear3, carry9, fortify1, shore2
- support4, helpdesk1, hotline1, service3, serve3, help2
- ...

Phrase relations are also indicated in the concept thesaurus:

- "kick the bucket" – die1 – expire4
- "SEC" – "Securities and Exchange commission"

There are currently over 76,000 concept thesaural groups.

This database is a primary source of Cognition's Semantic NLP unique combination of Relevancy and Completeness. Cognition's Semantic NLP not only knows all the different ways of saying things for full Completeness, it also knows which senses of the words should be counted as equivalents for high Relevancy. In fact, Cognition Semantic NLP's word meaning interpreter has 94% Relevancy. It is a commonplace in search to say that high recall comes at the expense of high precision. Since Cognition's Semantic NLP disambiguates words, and maps meaning to meaning in the thesaurus, the thesaurus improves recall without lowering precision.

- 9. Synographs.** Synographs are alternate spellings for entries in the dictionary, such as "cookie" and "cooky". This database allows for recognition of alternate spellings and common misspellings. There are currently over 12,000 synograph entries.

III. Lexicography Tools

The semantic databases have been developed with over 100 person years of lexicography work. A number of tools have been developed to assist lexicographers in the development of the semantic databases.

1. Lexicon Tool

Entries in the dictionary include single words, multiple-word phrases, and the nodes of the ontology. Each entry has one or more senses (meanings) with syntactic and semantic features.

Each word sense has several components, as follows:

a. plain English definition

This is what is displayed to the user in the search interface if the user chooses to view the search concepts.

b. ontological attachment(s)

Every sense is attached to one or more nodes in the ontology. For example, the first sense of "dog" is attached to "pet_node" and "canine_mammal".

c. syntactic features

There are currently about 1,250 unique syntactic features. Each sense has 2 or more features associated with it. The syntactic features include main category features (noun, verb, etc.), morphological features for classifying the different forms of a word (e.g. "wind", "winding", "wound"), and subcategory features (such as intransitive for verbs). For example, the first sense of "dog" has the category feature "noun", a morphological feature indicating how to pluralize it, and no subcategory features.

d. semantic features

Each sense may have semantic features, such as "domain" features (used to prefer a sense in a particular domain) and selectional restrictions (for use with a parser). Selectional features help guide word sense disambiguation. For example, the verb "charge" in the meaning "indict" requires a sentient object and a crime as the oblique object, whereas "charge" in the meaning "electrify" requires an electrical device as object and a form of energy as oblique object.

The tool runs as a server and guarantees that no word is edited by more than one lexicographer at a time, for integrity of the database. It also makes changes to words available to all lexicographers immediately after they have been saved into the database.

2. Meaning Thesaurus Tool

The concept thesaurus tool enables the lexicographer to look up words in the lexicon, view the meanings, select a meaning, and view all of the concept groups that the meaning is a member of. The lexicographer may add to the concept groups, delete from them, and create new ones. This can be done in parallel with more than one concept group at a time.

The tool runs as a server and guarantees that no concept group is edited by more than one lexicographer at a time, for integrity of the database. It also makes upgrades available to all lexicographers immediately after they have been saved into the database.

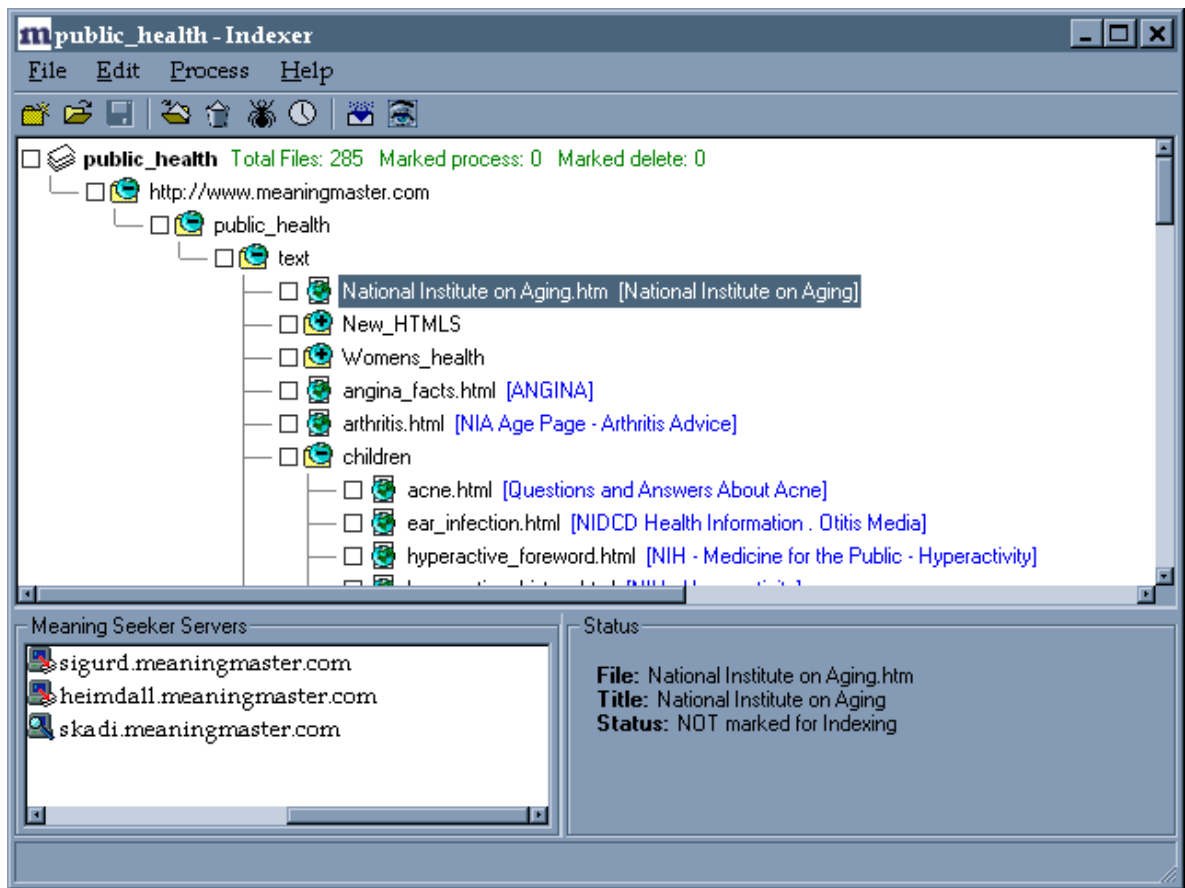
IV. Customer Tools

A number of tools have been developed to enable customers to index their documents, customize their specific jargon such as product names, and search their documents. Cognition's Semantic NLP employs client-server communication for optimal ease of use and efficiency on large document bases. A stand-alone, non-client-server version of Cognition's Semantic NLP is also available. At the core of the system is the Cognition's Semantic NLP Server, which can be configured for indexing, searching, or both.

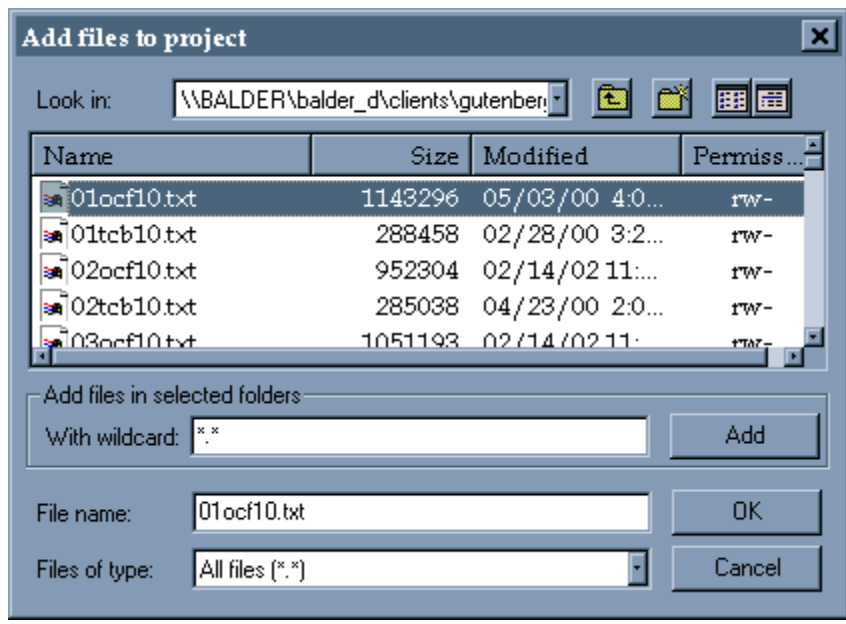
1. Indexing Tool

The Cognition Indexer GUI is the primary interface used to create and index a Cognition's Semantic NLP Project. A Cognition's Semantic NLP Project is simply a list of documents (in any of the formats Cognition's Semantic NLP handles), together with a set of parameters to be used when they are indexed and searched. It is straightforward and easy to use, though as with any software, good results will depend on training (or reference to the user's guide) and practice.

The following screenshot shows the main window of the Indexer GUI.



To get started, the user employs the Cognition Indexer GUI to create a list of the documents that the user wants to search. The following screenshot shows the Indexer window used for selecting files for indexing from a file system.



The Indexer is then used to send an indexing request to the Cognition's Semantic NLP Service. The documents to be indexed can be in HTML, plain text, MS Word, WordPerfect, RTF, or PDF formats, and can be included from the local network or from the Web via the Cognition Spider.

Following its initial creation, an index may be updated as often as needed. Updating the index does not entail re-indexing the entire document set, but only those documents for which a change is indicated. The Cognition Indexer GUI can automatically detect when documents have changed, and mark them for re-indexing. Documents can also be added to or removed from the document set by the user. In addition, a daemon may be invoked to automatically detect changes and update the index using the command-line version of the Cognition Indexer.

2. Sample Scripts

Once the index has been created, the user employs a Web browser (e.g. Internet Explorer or Netscape Communicator) to Search it. The Cognition's Semantic NLP package includes sample Web pages for this purpose. These sample pages may be used without modification, or they may be customized by the user to obtain the desired appearance and functionality. Sample scripts for ASP, Python, Java, PHP and Perl are available.

3. Automatic Dictionary expansion

Cognition's Semantic NLP dictionary can be expanded to include large numbers of customer vocabulary words automatically. Any file of terms and words used in the customer's business along with the categories of the terms can be merged automatically with Cognition's Semantic NLP dictionary. Recently, Cognition expanded its vocabulary in medicine and molecular biology by over 130,000 words using semi-automated techniques.

4. Customer Dictionary expansion

The customer may add words in ontological classes if desired. The customer may force Cognition's Semantic NLP into the desired meaning of a word, and the customer may force Cognition's Semantic NLP to consider a word a last name, or not a last name, as desired.

V. Product Features of Cognition's Semantic NLP

- 1. Relevance ranking.** Cognition's Semantic NLP returns a list of retrievals containing documents in which the query concepts were found together in a sentence. Those with exact word matches to the query terms come first. The next group is documents in which there were exact matches to some query terms in the body of the document, but other query terms only matched conceptually. Which terms match exactly is indicated. The last group is documents in which there were conceptual matches to the query terms. Within each group, documents are listed according to the number of sentences in which all query terms were found.

2. **Spelling correction.** In the Search interface, unrecognized words are noted and the user is given a list of alternative spellings to select from. The user may also leave the word as is. Cognition's Semantic NLP does search on words that are unrecognized if the user so desires.
3. **Specific retrievals highlighted.** When the user clicks on the "Highlighted Text" link corresponding to a retrieved document, Cognition's Semantic NLP highlights the relevant section. Additional relevant retrievals within a document are indicated by a pointing-hand figure at the end of a highlighted section. If there is no pointing-hand figure at the end of a highlighted section, it tells the user that there are no additional relevant results. In other words, it behaves like a clipping service, which is most useful with larger documents.
4. **Specific words highlighted.** When the user clicks on the "Highlighted Text" link corresponding to a retrieved document, Cognition's Semantic NLP also color-codes the specific words which matched the query within the relevant highlight section. As the user hovers the cursor over a matched word, the corresponding query term is indicated. Sometimes a given word may correspond to more than one query term. In this case the word is highlighted according to the first query term matched, but the hover-over text indicates all matched terms.
5. **Linguistic Boolean Search.** Cognition's Semantic NLP searches can be formed using fully-recursive Boolean expressions with AND, OR, WITH and AND NOT operators. The expressions connected with the Boolean operators are interpreted for meaning. See the help function at <http://medline.cognition.com> or <http://wikipedia.cognition.com> for complete instructions and examples of use for linguistic Booleans.
6. **Fuzzy Search.** Cognition's Semantic NLP searches can be formed using wildcards and fuzzy operators (*e.g.*, `"/Liebowicz/n"` matches names that sound like "Liebowicz", and `"<dr*g>i"` matches words that start with "dr" and end with "g", regardless of case) such that proper names and other words can be matched approximately. See the help function at <http://medline.cognition.com> or <http://wikipedia.cognition.com> for complete instructions and examples of use for fuzzy search.
7. **Review Tool.** As part its ASP solution, Cognition's Semantic NLP has a review tool for reviewing and categorizing document sets. Features include the ability to create project categories and classify documents into those categories, to limit searches within categories, to browse document sets without querying and to export archives of categorized documents. When combined with a relational database, the features provided allow the user to create and modify database tables, display document-related data from tables as part of query results and restrict searches based on table column values. Also included are scripts that allow the user to take notes and have the notes indexed and available to be searched in tandem with the related document set.
8. **Formats.** Cognition's Semantic NLP indexes documents in HTML, XML, OCR'd text and plain ASCII text. Documents in Word, Power Point RTF, or WordPerfect are converted to HTML before being indexed. Some engineering may be required for XML. Documents in PDF are

converted to plain text before being indexed. The user may view retrievals in documents converted to HTML or plain text with the specific retrieved sections highlighted, but may also choose to view the original file without highlighting.

- 9. Customer and meta-tags.** Cognition's Semantic NLP can search in customer-specified tags, if desired, with some engineering assistance from Cognition Technologies.
- 10. Indexing local directories.** The Cognition Indexer interface permits the user to select individual files or whole directories for indexing.
- 11. Spider.** Cognition's Indexer GUI includes an interface for creating a list of Web files to index. The user enters a URL from which to start, a desired depth to crawl, and parameters to include or exclude particular URLs.
- 12. Authentication.** The spider can be directed to use passwords or cookies to enter sites that require authentication, so that the user can index these sites.
- 13. Languages.** Cognition's Semantic NLP searches in any language handled by Unicode.
- 14. Search and retrieval pages.** Customizable search and retrieval ASP pages are provided for the user. The user can make the search and retrieval look any way desired.
- 15. Partial updating.** The user may select any number of files to re-index, rather than having to re-index an entire document base when individual documents are added or changed.
- 16. Automatic updating.** A console-level indexing command is provided so that system administrators can automatically update new files or changed files on a regular basis, initiated by the computer clock.
- 17. Load balancing.** The Cognition's Semantic NLP indexing interface automatically distributes document indexing across as many servers as the administrator selects.
- 18. Brokering.** Administrative tools enable the system administrator to manually control indexing and searching load, and membership access to document bases. The tools send queries to servers in response to load and allocate databases to specified servers. Customer-specific criteria that may involve user parameters such as subscription membership.
- 19. Categorization.** The interface queries users for categories and saves whole retrieval lists or individual files into the categories. New categories can be created on the fly. Subsequent searches can be restricted to categories.
- 20. User defined ontology.** Users can add ontological classes by editing a standard file. In this way users can define search into classes unknown to Cognition Semantic NLP, such as company widget names or phrases. For example, Sony could add a category video-recorder with specific video-record names as category members. Using this new category, and user

could ask “video-recorder” and retrieve to specific video recorder names mentioned in indexed documents.

21. User control of names. Users can force preference for name or non-name interpretations of words like Bush and Stone, which can either be names or common words.

VI. Comparison with Other Search Engines

The low Relevancy/Completeness performance of most Search engines is due to the use of pattern-matching technology. Pattern-matching matches strings-of-letters in a query to strings-of-letters within the document set, ignoring context and meaning. If you search on “check” meaning a “promissory note”, a pattern-matching search engine only searches for instances of letter-pattern “check”, regardless of the meaning of the word in context. Your results include “check” meaning “see whether”, “postponement”, and “hold back”, as well as “promissory note”.

Pattern-matching technology results in *information overload* because of word ambiguity, so that the best this technology can offer is typically no better than 33% Relevancy. It is important to note that word ambiguity is most prominent in the most frequently used words of a language. So the problem is concentrated among the very words which occur most often in queries and documents.

Also, most Search engines don’t require that your search terms be near each other in a document, so if you search on “pay with a check”, they will retrieve to documents in which “pay” is found in the last sentence, and “check” in the first sentence. Some engines don’t even respect word boundaries, retrieving texts that contain one of your Search terms as part of another word. If you search on “loss”, they will return documents with “gloss” and “floss”.

Pattern-matching technologies also miss relevant information. They only find material with the exact words of the search. If you search on “profit” meaning “net revenue”, they don’t find documents containing “net revenue”, or “income”. If you search on “SEC”, they don’t find documents containing “Securities and Exchange Commission”. Conversely, if you search on “Securities and Exchange Commission”, they don’t find documents containing “SEC”. If you search on vehicle, they only find that word, not types of vehicles such as “car”, “truck” and “plane”. This is why such technology has vastly inferior Completeness to linguistic technology employed by Cognition. Typically, pattern-matching technologies retrieve no more than 20 % of the information Cognition’s Semantic NLP retrieves.

Some Search engines have added statistical information along with pattern-matching search, but the end result is still the same – too much irrelevant information and too little relevant information.

Verity, Yahoo.

These engines employ pattern-matching technology. They produce over-retrieval because they do not disambiguate words and they lack phrases. If you ask "How do I check on my stocks?", they retrieve to "Check your stock in the cattle yard". If you ask about "math", they retrieve to "aftermath". Cognition’s Semantic NLP has little over-retrieval because it disambiguates words and has over 100,000 phrases. For the query "How do I check on my stocks", Cognition’s

Semantic NLP interprets "check" to mean "review" or "update", NOT "note for payment" and "stocks" to mean shares in companies, NOT bovine animals. Pattern-matchers under-retrieve due to lack of synonyms. If you ask about "pay raises", they won't find "salary increases". Cognition's Semantic NLP has little or no under-retrieval because of its synonyms and taxonomy. If you ask about "pay raises", it figures out the meanings of "pay" and "raise", and then retrieves to "salary increases", "wage hikes", etc. Pattern-matchers under-retrieve due to lack of taxonomy. If you ask about "vehicles", they do not retrieve "car", "ship" or "plane". Cognition's Semantic NLP has taxonomy, and thus can reason from the general to the particular.

Google

Google also employs pattern-matching technology, but with a statistical boost. It never retrieves to your information unless the query is worded the same way as the target document. However, it tracks popularity and places the most popular Websites first. These tend to be the sites other users want to look at, so it seems more on target than Search engines without the popularity measure. Cognition Technologies conducted a comparison of Cognition's Semantic NLP vs. Google on the www.globalissues.org site, a world political site. There were 50 queries in the test.

Examples of Google search issues that are resolved by Cognition.

Search query	Google	Cognition Semantic NLP
treasonous behavior	0 retrievals Google doesn't know synonyms	2 relevant 0 irrelevant
casualties of natural disasters	0 relevant 107 irrelevant	7 relevant 2 irrelevant
tidal wave	No reply Google doesn't know that a tsunami is a tidal wave	8 relevant 0 irrelevant
heating up the globe and biodiversity	0 relevant 4 irrelevant	1 relevant 0 irrelevant
turmoil in the Middle East and economic downturns	No reply	4 relevant 0 irrelevant

Autonomy

Autonomy also employs a pattern-matching search engine with some statistical enhancements using Bayesian inference. In testing Autonomy on the New York Life Site, Cognition's Semantic NLP did not see the effects of the statistical reasoning. If working, such technology would recognize topics of documents based upon the probability of occurrence of individual words. These probabilities are calculated by associating a hand-assigned document topic with the words that occur in the document. For example, a text that has been assigned the topic baseball will have large numbers of occurrences of the words "ball", "bat", "hit", "field", "strike", etc. Thus upon searching for "baseball" in documents that have not been assigned topics by hand, documents with high numbers of those associated words will be recognized as being about "baseball". This type of technology only increases retrieval for those topics that have been identified in advance as of interest to many users, and for which additional processing of trial documents has been formed. Thus it only works well in a small number of cases, and to get even these cases to work is labor-intensive.

In practice, as Cognition tested it, Autonomy has two problems. It doesn't disambiguate words and at the same time it adds many search terms with statistics, so its retrieval Relevancy is only 0.5%. Secondly, it has poor Completeness because it has no paraphrasing or thesaurus. It misses 60% of the relevant material that Cognition's Semantic NLP finds on the same site (www.nylife.com).

VIII. Cognition's Semantic NLP Specifications

Deployment:

In order to deploy Cognition, the user installs the Cognition's Semantic NLP program and the Cognition's Semantic NLP Web component for Search. After installation is complete, the Cognition's Semantic NLP service (for Windows) or daemon (for Linux) is started automatically. This Cognition's Semantic NLP Server functions for indexing, search or both, as set by the user in the server administration interface.

The first step is to index the target documents, which can be either on the local network or on the Web. If the documents are local, the user selects which documents to index from the file system. If the documents are on the Web, the user employs the Cognition Spider to obtain a list of documents. Once the project has been created, the documents can be indexed.

The second step is the Search itself. At least one Cognition's Semantic NLP Server must be running with Search enabled. The user creates a script page (ASP and Perl are both supported) as in the sample provided with the program, and then visits that page in a browser (such as IE or Netscape). From that page, the user can search.

Platforms (operating systems):

Windows (32-bit) (Windows 2000, Windows XP, Windows 2003)
Unix, Solaris, RedHat, Linux, FreeBSD, Centos

RAM: 2 GB preferred, 1 GB minimum

Processor: Minimum 3.0 GHz preferred. Multi-core Hyper-threaded preferred.

Disk: SCSI/SAS drives preferred. Cognition's Semantic NLP requires 140 Mb of disk space plus space for concept indices, which range from one to two times the size of the original texts.

Search Interface: Any HTML browser such as Thunderbird or Internet Explorer, accessing a script which sends requests to the Cognition's Semantic NLP Server. ASP, Perl, Python, and PHP4 are all supported.

API: A C++ API to the Cognition's Semantic NLP system is available. An API to the ASP Component is provided in the documentation that comes with the software.

Speed: Cognition's Semantic NLP builds Concept Indices at about 1 hour per GB, depending on the machine, the configuration, and the text itself. It can handle an unlimited number of queries simultaneously using queuing technology and a sufficient number of computers.

Vocabulary: Cognition's Semantic NLP knows most English areas of interest or domains, except company-specific terminology, such as product names (which can be learned).

Technical Support: Support is available from 9am to 6pm Pacific time. Extended support with a 3 hour minimum response time can be arranged. With a service contract, Cognition's Semantic NLP will regularly monitor performance and make system adjustments to further enhance Relevancy and provide optimum quality control.

IX. Conclusion

Cognition's Semantic NLP, when applied to Search technology, returns the most relevant and complete results in the industry by employing linguistic techniques and huge semantic databases. Pattern matching and statistically-based technologies lack the knowledge of language that enables Cognition's Semantic NLP to outperform them so dramatically. Because of its vast knowledge of English, little or no customization is required. Search functions like those users are accustomed to are included, such as Boolean search (with conceptual Booleans) and fuzzy search. It comes with an easy-to-use indexer, and sample scripts for browser searching. Cognition's Semantic NLP has an API for embedding it in other software platforms. After a short time using Cognition Semantic NLP's patented technology, users are loathe to go back to pattern-matchers.

Appendix Glossary

ambiguous word - an ambiguous word has more than one meaning. "strike" is ambiguous because it can mean "to hit", "to ignite", "to walk out of a job", "a good pitch in baseball", or other meanings. An unambiguous word has only one meaning, as in "deer".

Bayesian - In search, a technique for classifying documents uses the statistical theorem of Bayes. In brief, this theorem explains what the probability of one word is, given another. So if you've seen "bat" in a document, the probability of seeing "ball" is higher than the probability of seeing "rock".

concept thesaurus - A traditional thesaurus lists all the synonyms of words in meaningful groupings, e.g.: "strike hit beat" as one group, "strike walkout protest" as another group. A concept thesaurus lists MEANINGS of words in groups, so if the first meaning of "strike" means "to hit or beat", then one thesaural group is "strike1 hit3 beat2" (assuming that the third meaning of "hit" and second meaning of "beat" mean the same thing). If the second meaning of "strike" means "walkout or protest", then another, independent, thesaural group would be "strike2 walkout1 protest2". Thus a concept thesaurus maps meaning to meaning, while a traditional thesaurus maps word to word, with no way of deciding which instance of a given word should be considered synonymous with which other words.

compound - a word that is formed from two or more identifiable words, e.g. "blackbird," "cookbook"

concept - a word meaning (see word meaning)

computational - the property of being an action carried out by a computer.

derived form - a word that is derived through morphological rules, such as "rerun" or "derivation".

domain - a subject area of human activity which has a sub-vocabulary of its own, such as law or medicine. Also "domain" refers to a basic part of a URL address which breaks down into three pieces, "site.domain.suffix" and in "www.cognition.com".

fine granularity query - a fine grained query is one which is looking for detailed information. For example, a fine granularity query would be "What is the cure for leaf wilt on fuschias?". In contrast a coarse granularity query (or general query) would be "How can I buy a car in Los Angeles?"

GUI - graphical user interface, such as the basic interface for Windows.

idiom - a fixed distinctive expression whose meaning cannot be deduced from the combined meanings of its actual words, such as "kick the bucket".

incremental indexing - the ability to add a new document index to an existing document base index. This ability is an advance over many available indexers, which require users to completely re-index an entire document base when new documents are added to it.

index - a computational representation of the location of words or concepts in a document base. This can include details such as word and sentence positions, or not.

latent semantic indexing - A statistical technique for classifying documents based on counting co-occurrences of words. Documents that share many words are considered to have similar content and are classed together.

linguistic algorithms - rules of language applied as computer algorithms. For example, given a word that pluralizes like "bat", add an "-s" to make it plural, or remove an "-s" to find the base form or stem.

linguistics - the study of human language including elements, structure, rules and history.

morphological feature - a property of a word that indicates how it changes form in specific syntactic situations. For example, the fact that the past tense of "catch" is "caught" is expressed in the dictionary with a morphological feature (see morphology).

morphology - the rules of word formation. Such rules dictate that the past tense of "cook" is "cooked", while the past tense of "catch" is "caught", and that the plural of "bat" is "bats", while the plural of "mouse" is "mice". Morphology also controls the derivation of words from other words, such as the change from "derive" to "derivation", or from "run" to "re-run".

overload - retrieval of many documents that are not relevant in response to a query; poor relevancy; poor precision

parsing - combining the words in a sentence into a structure which elucidates or shows the syntactic role of each word in the sentence. For example, in "John loves pizza", the parser creates a structure showing that "John" is the subject and "loves pizza" is the predicate, and that within the predicate, "love" is the verb, and "pizza" is the direct object of the verb.

pattern-match (or string pattern-matching) - In search, deciding to retrieve a document based on an exact match of strings in the query and document. For a query "what is the form of the law?", a pattern-matching search engine would match documents containing phrases like "...conforms to the law...", "...inform him that the lawn...", "...the laws of nature dictate that water forms crystals at a temperature of ..."

phrase - a meaningful sequence of words. "bok choy" is a phrase, while "the and" is not.

phrase parser - a computer program that reads in text and determines which sub sequences of sentences are phrases

precision/recall - This is the same thing as relevancy/completeness. Precision is the percentage of a set of retrievals that is relevant. If there are ten retrievals, and 9 of them are relevant to the query, then precision is 90%. Recall is the percentage of the relevant documents in the target database that is retrieved in response to a query. If there are 10 documents that exist in the target database that would be relevant to a query, and the software retrieves 9 relevant documents, then recall is 90%.

reasoning from the general to the particular - reasoning in a taxonomic tree from higher to lower nodes. For example, reasoning in the taxonomy example under the definition of "taxonomy", from "vehicle" to "car", or "vehicle" to "boat" (see taxonomy).

relevant - In search, document content that addresses or is similar to the content of a query; apt; on target

relevancy/completeness - This is the same thing as precision/recall. Relevancy is the percentage of a set of retrievals that is relevant. If there are ten retrievals in response to a query, and 9 of them are relevant, then relevancy is 90%. Completeness is the percentage of the relevant documents in the target database that is retrieved in response to a query. If there are 10 documents that exist in the target database that would be relevant to a query, and the software retrieves 9 relevant documents, then completeness is 90%.

semantics - In linguistics, the rules for determining meaning of words, sentences and discourses. The meanings of individual words are typically listed in a lexicon or dictionary in a computational linguistic system.

sense - An individual meaning of an ambiguous word. "strike" meaning "to hit or beat" is one sense of "strike".

spider - a computer program that searches the internet by connecting from one URL to the next via the links in it.

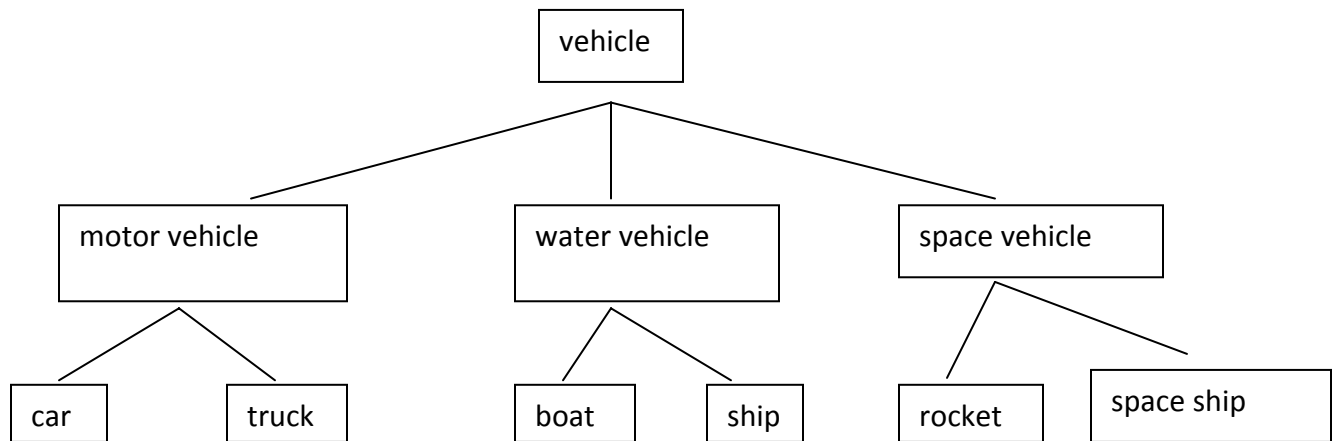
string - a sequence or pattern of letters. A string can be a word or not. "XXX " is a string, "cat" is a string.

synograph - an alternate spelling of a word, such as "center" or "centre".

syntactic feature - a property of a word that indicates how it functions in grammar. For example, the fact that a word is a noun is a syntactic feature. The fact that a verb requires a direct object is a syntactic feature. A verb which requires an object, used without one, forms ungrammatical sentences, as in "The girl wants."

underload - retrieval of very few, if any relevant documents in response to a query; poor completeness; poor recall

taxonomy - a hierarchy of ISA relationships between concepts that form a tree, with the top of the tree the most general concept. For example, a car is a motor vehicle in the vehicle taxonomy:



thesaural enhancer - A type of search engine that uses standard, non-conceptual thesaural groups to enhance query terms. If the query contains the word "strike", the thesaural enhancer adds terms from all the thesaural groups "strike" is a member of. So it would search on "hit, beat", but also "walkout, protest", "ignite", etc. Note that many of the synonyms are ambiguous here. The thesaural enhancer doesn't distinguish meanings, and doesn't know which meaning of "hit" or "beat" is relevant.

word meaning - an individual sense of a word. A sense is a description or mental picture of the objects referred to by the word sense. For example, the word "bank" can either mean "a place where money is stored", or it can mean "the side of a river". The meaning of the sense "where money is stored" is a description of a typical bank with tellers, counters, little windows to the tellers, a guard, a vault, etc.

word stem - the base form of a word with no morphological rules applied. "bat" is a base form, "bats" is not. "run" is a base form, "ran" is not and "re-run" is not.