TECH_08_A2/2-2008-0092

MODELLALAPÚ SZEMANTIKUS KERESŐ RENDSZER KIDOLGOZÁSA

IDŐKÖZI SZAKMAI BESZÁMOLÓ

1. SZAKASZ

A SZEMANTIKUS KERESŐRENDSZEREK ÁTTEKINTÉSE

1.2. melléklet

Alkalmazott Logikai Laboratórium
Szegedi Tudományegyetem

Verziókövetés

dátum	Változtatás	szerző
2009. 1. 20.	első változat	Varasdi Károly
2009. 3. 15.	általános jellemzés, ontológiát használó keresők áttekintése	Simonyi András
2009. 12. 19.	a két változat összefésülése, egységesítése	Simonyi András

1 The Problem

For the purposes of the present survey, the task of a semantic search engine is to take a query specifying a piece of semantic content as input, and to select those 'matching' items from a class of natural language text documents, which contain semantic content identical or sufficiently similar to the query.

2 Solution Paradigms

We can distinguish two basic solution paradigms:

- *Static* semantic search engines, where a semantic representation of text documents is generated, independently (and, typically, in advance) of the query.
- *Dynamic* engines, where the semantic analysis of documents is based on the query.

The two basic approaches can be combined into *hybrid* engines employing both query-independent and query-dependent semantic analysis methods: e.g. an engine may perform shallow semantic parsing in a preprocessing phase, and query-dependent deep semantic analysis in the query-execution phase.

2.1 Static Semantic Search

The defining characteristic of static semantic search architectures is that the generation of the documents' semantic representations is *independent* of the query. Typically, a static search engine generates the semantic representations during a preprocessing phase, and search is performed via an *index* assembled from the semantic representations. The main components of a static semantic search engine are the following (cf. [AE06]):

- **Annotator.** The annotator component generates semantic annotations for the documents. A semantic annotation consists of two components: (i) (partial) semantic representation of the content of the document (ii) a 'locational' mapping from components of the semantic representation to parts of the text.
- **Query UI.** Enables the user to specify her query.
- **Query representation generator.** This component generates a semantic representation from the query which can be compared with the document annotations.

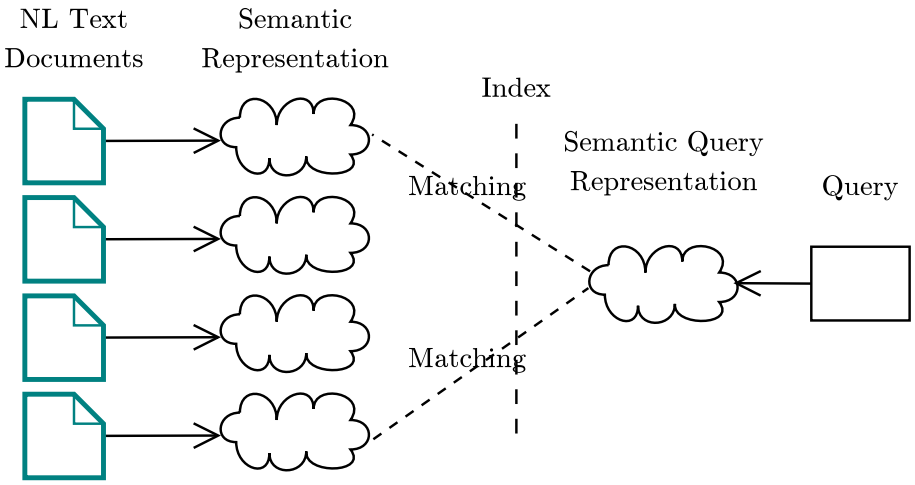

Figure 1: Representations in a static semantic search engine

- **Retriever.** The retriever module compares the semantic representation of the query with the semantic representations of the documents, and selects those documents that have parts 'matching' the query. In the majority of the cases, the output of the retriever is an ordered set, i.e. the module provides a ranking of the results.
- **Result presentation module.**

2.2 Dynamic Semantic Search

In contrast to a static semantic search architecture, a dynamic engine performs semantic analysis on the basis of the query, and the purpose of the analysis is solely to decide whether the document in question matches the actual query (and, in certain cases, to calculate a similarity score between the document and the query, which can be used for ranking the results).

3 The Role of Ontologies

Ontologies may play one or more of the following important roles in a semantic search engine:

- **Semantic representation.** As formal languages with a well defined semantics, terms and statements of an ontology language can be used to represent (features of) the content of natural language texts, and also to represent the query.
- **Measuring similarity.** Knowledge bases stored in the form of ontologies can be used for measuring the similarity (relevance) of semantic representations.

- **General knowledge representation.** The grammatical and semantic analysis of natural language texts typically requires a certain amount of background knowledge, and the query UI may also rely on background information. In these cases, the used knowledge can be stored (partly) in ontologies.

3.1 The Use of Ontology Languages for Representing Semantic Content

The majority of the ontology-based semantic representation schemas utilise only certain fragments/constructions of full-blown ontology languages such as variants of OWL. As a consequence, existing solutions vary greatly with respect to their expressive power. Typical choices include (in increasing order of expressivity):

- **Using ontology terms as a set of labels for 'tagging'.** If textual units larger than words are tagged/annotated (e.g. sentences, paragraphs etc.), then the typical use is thematic tagging: it can be indicated that a certain paragraph is about cars etc.

  If the tagged/annotated units are words, then the tags may indicate the sense of a word in the given context, i.e. it can be used for recording the result of disambiguation, and/or for indicating the categories named entities belong to.

- **Atomic sentences.** In this case atomic sentences of the ontology language are used for representing content. Some solutions utilize only monadic predicates, while others also involve dyadic ones.

  A meaning representation based only on atomic statements with dyadic predicates can be surprisingly powerful if co-reference is indicated by using the same individual constant across atomic sentences. For instance, the content of the sentence

  (1) John said Tom gave Mary a pen.

  can be (partially) described by the following atomic sentences in the language of an eventuality-oriented Davidsonian ontology:

  (2) Saying(s) Giving(g) Pen(p) hasTheme(s,g) hasActor(g,Tom) hasActor(s,John) hasObject (g,p) hasBeneficient(g,Mary)

- **Molecular sentences.** In addition to atomic sentences, Boolean constructions are also used, making it possible to represent negation, disjunction etc.

- **Quantification.** In addition to molecular sentences, quantifier constructs of the ontology language are also used in the semantic representations.

4 Semantic Search Engines

In this section we give an overview of some of the leading semantic search engines. As the present survey focuses on ontology-based solutions, the first two subsections about Lexxe and Setrure are simply compilations of quotations from their respective websites.

4.1 Semantic Search Engines not (known to be) Relying on Ontologies

Lexxe Lexxe is a third generation Internet meta search engine featuring Natural Language Processing technologies. It is fully automatic without human editing involved. Most of its answers come from unstructured texts and webpages on the Internet.

Two types of search:

1. Key Word Query Processing: key word queries will be parsed (cut into phrases)
2. Natural Language Query Processing: one can ask a question starting with a modal verb (e.g. must, may, can, etc) or an auxilary verb (e.g. is, have, did, etc). A question word like “who”, “which”, “what”, “when”, “where”, “why” and “how” must be placed in the beginning of each query.

Searching

1. Lexxe retrieves search engine results
2. then take the unstructured text results for a comprehensive analysis, which includes
   - Clustering: a way to point to the users a series of important words and phrases that are extremely closely related to the documents found.
   - Meaning Understanding: Lexxe is able to recognize those words that suggest the meaning of a person or profession relevant to the query.
   - Phrase Recognition and Short Answer Extraction: the parsing of (result) sentences to locate the short answer to the query.

Setrue Setrue Semantic Patent Search Engine offers:

- Semantic natural language query capabilities
- Query auto-complete suggestions
- Detailed USPTO (United States Patent and Trademark Office) class directory structure enabling a one click segmentation of search results to a specific uspto patent class

- Clustering of search results by uspto classes, years, assignees
- Similar patents segmentation
- Dynamic weighting of search terms

Setrue's semantic parser:

- focuses on the sentence level, and scales up to the document level
- yields the semantic parsing tree representing the sentence.

Search Engine	*Ontology*	*Ontology Size*
Hakia	OntoSem	≥8500 classes/relations
Cognition	Cognition Semantic Map	≥7500 classes/relations
Powerset	Freebase,?	N/A
Squirrel	PROTON	∼200,000 entities
UpTake	UpTake travel ontology	'thousands of concepts'
GoWeb	Gene Ont., MeSH Ont.	≥24,500 terms, ∼18,000 categor.

Table 1: Ontology usage in a few semantic search engines

4.2 Ontogy-Based Semantic Search Engines

4.2.1 Domain-Independent Solutions

Hakia Hakia is a general-purpose static web search engine relying on the following technologies:

- the QDEX (Query Detection and Extraction) indexing method, which analyses the content of a web page, and generates a set of possible queries that can be asked to this content.
- the SemanticRank algorithm, which ranks the results of a query in relevancy order.

Both of these technologies are based on Ontological Semantics, an ontology-based theory of linguistic meaning described in [NR04].

QDEX produces queries on the content of web pages by generating semantic representations of the pages (TMRs, or Text Meaning Representations) in the non-standard ontology language of Hakia's OntoSem ontology. OntoSem TMRs are sets of atomic sentences with monadic and dyadic predicates, and the representation is along broadly Davidsonian lines, as OntoSem is an eventuality-oriented ontology. Although atomic in form, OntoSem TMRs can represent truth-functional

connectives by reification: e.g. the negation of a sentence can be represented by attaching a special 'negation modality' attribute to the eventuality representing the sentence's content [NR04, 297].

The language-independent OntoSem ontology used by Hakia contains more than 8500 concepts [Nir07]. Although OntoSem was originally developed and has been since maintained in a non-standard format, it can be (and partially has been) converted to OWL-DL [BGK06]. Hakia currently handles only web pages in English — the English lexicon linking words to ontology concepts covers approximately 100,000 English word senses [Hak], and more than a million English words [Mac08].

Cognition Cognition is a comprehensive NLP framework, which contains a domain-independent search engine module. Although the company plans to develop a general web search engine in the long run [Mac08], currently only vertical, narrow deployments of the engine are available. There are publicly accessible applications of Cognition to the following domains/document sets: Wikipedia, Medline abstracts, The Gospels (text and notes), and Fed and Supreme Court Decisions.

As sketched in their white paper [Dah07], Cognition's semantic search engine has a traditional static architecture, in which the semantic analysis of the document set to be searched is performed during a preprocessing phase. The analysis relies on an extensive English lexicon covering 536,000 word senses, and the semantics is given partly in terms of an ontology containing 7,500 nodes [Mac08] (this ontology might well be only a taxonomy, see [DA08, 2]). Cognition claims that their system covers '10 million semantic connections; over 4 million semantic contexts' [Mac08].

Although no details have been published about the form and expressivity of the semantic representations generated by Cognition's parser, references to a (not yet commercially available) 'lambda expression generator component' [Mac08, 3] seem to indicate that semantic representations might go beyond the atomic and molecular levels.

Powerset Powerset is a general-purpose semantic search solution, which—similarly to Cognition—aspires to become a comprehensive web search engine, but currently indexes only a relatively small document set, Wikipedia.

Although information about Powerset's architecture is rather scarce, it seems to be a static system, which annotates web documents with RDF triplets ('factz' in Powerset terminology) as a partial description of the document's content.

Powerset uses NLP technologies licensed from Xerox PARC [Hel07] to extract semantic content, and it is known that the RDF representation relies on ontologies [Joh07]. Unfortunately, no details about these ontologies are available except for the fact that Powerset in part utilises Freebase, a collaborative RDF-based fact-database collection, which is freely available and can be accessed via the Metaweb Query API, and MQL (Metaweb Query Language) (see [MQL]). Freebase, in

effect, can be considered as a collection of collaboratively built ontologies: the ontologies contain relatively few concepts and relations, as Freebase is focused on accumulating A-box data.

Recently Microsoft has acquired Powerset, and plans to use its semantic technology in its upcoming search engine internally named Kumo [Mon09].

Squirrel Squirrel is a general-purpose semantic search and browse tool developed in the EU sponsored SEKT (Semantically-Enabled Knowledge Technologies) project (2004–2007), which was trialled in a case-study on improving digital libraries.

As described in [DGD07], in this pilot project Squirrel supplied two functions that can be termed 'semantic search in text documents':

- it provided thematic search in the digital library,
- certain phrases of the documents (referring to 'named entities') were annotated with (links to) concepts/instances in an ontology, and Squirrel provided text search in this type of metadata (in addition to the full-text search in the documents themselves).

Squirrel has a traditional static architecture, in which the semantic data used during the search process is built in a preprocessing phase. The generation of a thematic taxonomy and the automatic classification of documents implements the semi-automatic clustering method described in [FMG05], while the generation of semantic annotations utilises the named entity recognition and information extraction capabilities of the GATE (General Architecture for Text Engineering) NLP platform [Cun02, BTMC04].

The pilot-system was based on the PROTON lightweight general purpose ontology, which was also developed in the SEKT project, and the PROTON-based knowledge-base used for semantic annotation and extraction contained more than 200,000 entities [DGD07, 4].

Avatar Avatar is a general-purpose semantic search engine developed in IBM's Almaden research lab.

As described in [KKR+06] and [KRVZ06], Avatar has a static architecture, in which a series of IE modules annotate the documents with concepts and relationships that are important in the application context, and atomic statements resulting from the information extraction/annotation process are stored in a database. Although publications about Avatar do not call the used collections of relevant concepts and relationships ontologies, they can be considered small domain ontologies containing only T-box statements.

Avatar accepts a list of keywords as a query, and transforms this list into a series of plausible formal 'interpretations', which are, in effect, representations of the query's hypothesised content in Avatar's database query language.

Currently the most important Avatar-based search solution is OmniFind Personal E-mail search, which is a freely downloadable plug-in enabling semantic e-mail search in Microsoft Outlook and Lotus Notes.

4.2.2 Domain-Specific (Vertical) Solutions

Convera Convera is a vertical semantic search engine provider, whose products include search engines for enterprise content management platforms, information retrieval solutions for publishers (used by e.g. Wiley Publications), and online vertical search engines. The latter include SearchMining.net for searching the mining industry domain, MaritimeAnswers.com for shipping industry, and SearchMedica.com for medical search.

Convera's search engine has a static architecture which prepares an index in a preprocessing phase, and it utilises NLP techniques, ontologies and taxonomies for extracting concepts from text documents [Han07]. No specific information appears to be available about the use of ontologies in Convera's current product line. In RetrievalWare 8, one of Convera's earlier enterprise search products, concepts (synsets in Convera's terminology) were organized into a semantic network similar to Wordnet, and were also linked to nodes in a taxonomy [BHJ+05].

UpTake UpTake is a travel search engine indexing more then 5000 sites about tourist destinations, accommodation etc. in the U.S., which also stores information about traveller's opinions and reports.

Based on a static architecture, UpTake extracts information into the A-box of a travel ontology using NLP methods and sentiment analysis. UpTake's ontology 'currently has thousands of concepts, relationships and rules', and the company plans to 'introduce machine-learning to automatically identify more relationships and propose more rules' in the long run [Alt08].

One of the most important features of UpTake that distinguishes it from other engines is that it is not 'stateless' but 'conversational': it focuses on multi-question queries, in which the user moves from general queries to more specific ones [Alt08].

Trovix Trovix is a job search engine, which uses job seekers' resumes to improve search results. As described in their search technology overview [Tro], Trovix's static architecture normalises input resumes into semantic representations comprising of concepts and relations that are components of a proprietary job-ontology. Natural language search queries are also converted to the language of Trovix's job-ontology, and the comparison of the query with the content of resumes and form-based data provided by the users is realized at the level of ontology-based semantic representations.

GoWeb GoWeb is a semantic web-search engine for life sciences. Following [DS08], the operation of GoWeb can be summed up in the following way:

Accepting a natural language query as input, GoWeb first executes a traditional keyword-based web-search using Yahoo's BOSS service. The result is a set of textual summaries ('snippets'), which serves as the input for semantic analysis and ranking. GoWeb uses special algorithms developed for the GoPubMed medical literature search service for identifying medical concepts of the Gene Ontology and the MeSH (Medical Subject Headings) ontology, and it also relies on the OpenCalais semantic annotator service for recognising names of persons and places. The user interface shows relevant fragments of the Gene and MeSH ontologies, and provides ways of sorting/filtering the result set using concepts from these ontologies.

References

[AE06] H. Abolhassani and K. S. Esmaili. A categorization scheme for semantic web search engines. *4th ACS/IEEE International Conference on Computer Systems and Applications (AICCSA-06)*, 2006.

[Alt08] UpTake under the hood—the Interview. *AltSearchEngines*, May 14, 2008. WWW document. http://www.altsearchengines.com/2008/05/14/uptake-under-the-hood-exclusive-interview/ (accessed March 27, 2008).

[BGK06] G. Barzdins, N. Gruzitis, and R. Kudins. Re-engineering OntoSem Ontology Towards OWL DL Compliance. In *Knowledge-Based Software Engineering: Proceedings of the Seventh Joint Conference on Knowledge-Based Software Engineering*. IOS Press, 2006.

[BHJ+05] O. Bayer, S. Höhfeld, F. Josbächer, N. Kimm, I. Kradepohl, M. Kwiatkowski, C. Puschmann, M. Sabbagh, N. Werner, and U. Vollmer. Evaluation of an Ontology-based Knowledge-Management-System. A Case Study of Convera RetrievalWare 8.0. *Information Services and Use*, 25(3), 2005.

[BTMC04] K. Bontcheva, V. Tablan, D. Maynard, and H. Cunningham. Evolving GATE to Meet New Challenges in Language Engineering. *Natural Language Engineering*, 10(3/4):349–373, 2004.

[Cun02] H. Cunningham. GATE, a General Architecture for Text Engineering. *Computers and the Humanities*, 36:223–254, 2002.

[DA08] K. Dahlgren and D. Albro. Cognition technology resources overview: Semantic map, system architecture and tools. Technical report, Cognition Technologies, Inc., 2008. http://www.cognition.com/pdfs/Cognition_Technology_Detail.pdf.

[Dah07] K. Dahlgren. Technical overview of Cognition's semantic NLP (as applied to search). Technical report, Cognition Technologies, Inc., 2007. http://www.cognition.com/pdfs/Cognition_Semantic_NLP_for_Search_Overview.pdf.

[DGD07] A. Duke, T. Glover, and J. Davies. Squirrel: An advanced semantic search and browse facility. In *The Semantic Web: Research and Application*, Lecture Notes in Computer Science. Springer, 2007.

[DS08] H. Dietze and M. Schroeder. GoWeb: A semantic search engine for the life science web. In A. Burger, A. Paschke, A. Romano, and A. Splendiani, editors, *Proceedings of the Intl. Workshop Semantic Web Applications and Tools for the Life Sciences SWAT4LS*. Edinburgh, 2008.

[FMG05] B. Fortuna, D. Mladenic, and M. Grobelnik. Semi-automatic construction of topic ontology. In *Proceedings of the ECML/PKDD Workshop on Knowledge Discovery for Ontologies*, 2005.

[Hak] Hakia Lab. WWW document. http://labs.hakia.com/hakia-lab-onto.html (accessed March 26, 2009).

[Han07] P. J. Hane. FAST buys Convera's RetrievalWare. *Taxonomy Watch*, 2007. WWW document. http://taxonomy2watch.blogspot.com/2007/04/fast-buys-converas-retrievalware.html (accessed March 27, 2009).

[Hel07] M. Helft. In a search refinement, a chance to rival Google. *The New York Times*, February 9, 2007.

[Joh07] M. Johnson. Powerset Blog: "politicians who died in office", June 8, 2007. WWW document. http://www.powerset.com/blog/articles/2007/6/8/politicians-who-died-in-office (accessed March 27, 2009).

[KKR+06] E. Kandogan, R. Krishnamurthy, S. Raghavan, S. Vaithyanathan, and H. Zhu. Avatar semantic search: a database approach to information retrieval. In *Proceedings of the 2006 ACM SIGMOD international conference on Management of data*, pages 790–792. ACM New York, 2006.

[KRVZ06] R. Krishnamurthy, S. Raghavan, S. Vaithyanathan, and H. Zhu. Using structured queries for keyword information retrieval. Technical report, IBM, 2006.

[Mac08] R. MacManus. Cognition announces 'world's largest semantic map'. *ReadWriteWeb*, September 16, 2008. WWW document. http://www.readwriteweb.com/archives/cognition_semantic_map.php (accessed March 26, 2009).

[Mon09] E. Montalbano. Microsoft testing Kumo search engine internally. *NetworkWorld*, March 3, 2009. WWW document. http://www.networkworld.com/news/2009/030309-microsoft-testing-kumo-search-engine.html (accessed March 27, 2009).

[MQL] MQL reference guide. WWW document. http://mql.freebaseapps.com/ (accessed March 27, 2009).

[Nir07] S. Nirenburg. Homer, the author of the Iliad and the computational-linguistic turn. In *Words and Intelligence II*. Springer, 2007.

[NR04] S. Nirenburg and V. Raskin. *Ontological Semantics*. The MIT Press, 2004.

[Tro] Trovix—search technology. WWW document. http://www.trovix.com/about/technology.jsp (accessed March 27, 2008).