Semantic Enterprise Search (but no Web 2.0)

Ronald Winnemöller

***Abstract*—In this paper, we propose semantic enterprise search as promising technical methodology for improving on accessibility to institutional knowledge. We briefly discuss the nature of knowledge and ignorance in respect to web-based information retrieval before introducing our particular view on semantic search as tight fusion of search engine and semantic web technologies, based on semantic annotations and the concept of intra-institutionwise distributed extensibility while still maintaining free keyword search functionality. Consequently , our architecture implementation makes strong use of the Aperture and Lucene software frameworks but introduces the novel concept of "RDF documents". Because our prototype system is not yet complete, we are not able to provide performance statistics but instead we present a concise example scenario.**

***Index Terms*—RDF documents, semantic web, ignorance.**

I. Introduction

INTUITIVELY, it is the duty of Universities (and, to a certain degree, of technikons and other schools) to produce knowledge in research and teaching.

This, we might assume, is what they do very well.

Unfortunately, we may also find that keeping, consolidating and making accessible that knowledge, even when we restrict ourselves to electronically stored knowledge, is a field that is neglected in many cases – a fact that is acknowledged by several institutions as stated in the *Implementation of the Berlin Declaration on Open Access*, cf. [1] and the *Berlin Declaration on Open Access to Knowledge in the Sciences and Humanities* itself, cf. [2].

We can, for example, identity the following "knowledge leaks" as common for many institutions:

- The conventional, if not required publishing process often means that a researcher sums up his knowledge in an journal article or conference paper – which eventually gets published by some commercial company. It does not necessarily mean that this publication is kept (electronically) in the realm of the home university of the mentioned researcher and be available to other members – students or fellow researchers – of that institution. This issue is also reflected on in [1].
- Many institutions run web-accessible publication databases, like eprint archives, citation indexes and such – sometimes even at the department or team level. While these repositories usually provide for intra-repository search functionality, they cannot be searched using an institution-wise global methodology because of the well known "hidden web" problem (cf. e.g. [3], [4]).
- ELearning, content management and "Web 2.0" systems (such as departmental wikis, blogs, etc.) usually require authentication (and authorization) prior to accessing content. These knowledge sources cannot be accessed through global methods without special adaption to these requirements. Even then, automatic retrieval methods still face the above mentioned "hidden web" problem.
- Other types of knowledge material are transmitted electronically but are of transient nature, such as RSS-Feeds (on an organizational basis), mails, filestores, etc. We hesitate to call these types "documents" because of their temporary character - but nevertheless they might contain valuable knowledge nonetheless.
- Some publications are simply not available electronically, because they were published through a "pure-paper"[1] process.
- Certain documents may be accessible through the "visible" intraweb of an institution, but due to an ineffective implementation of the retrieval process they may still be inaccessible.

In this paper, we propose a *technical* methodology for improving on accessibility to institutional knowledge.

We will not, however, try to solve *social* institutional issues such as implementing open access publishing processes, etc[2].

The remainder of this paper is structured as follows:

In the next section, we will briefly discuss the term "knowledge". After this, we will put our work in an appropriate scientific context in section III, followed by a description of our own approach (section IV), including our proposed architecture and already implemented modules. Since we are still working on a concise evaluation procedure, we will instead preliminarily provide a realistic scenario in section IV-C as indication for the intended functionality. Subsequently, we conclude in section V

II. A Brief Elaboration on a Search-related Notion of "Knowledge"

In order to clarify what we are talking about, we need to discuss what we mean when using the term "knowledge".

Manuscript received February 2, 2009. Manuscript accepted for publication March 17, 2009.

The author is with University of Hamburg, Germany; email: ronald.winnemoeller@uni-hamburg.de

[1] As opposed to "paper-less".

[2] We will rather leave this issue to the Web2.0 community ...

Unfortunately, even a superficial definition of the nature of "knowledge" is far beyond the scope of a paper like this one but we would like to prevent some common misunderstandings:

1) We don't discuss open philosophical issues here (especially we will not argue about truth conditions, belief or justification). instead, we see "knowledge" in very narrow technical terms as *intentionally stored or transmitted information*, usually – but not necessarily – contained in electronically represented documents within a specific organization
2) Despite what we just claimed, "knowledge" does not equal "document content" but rather the information contained therein. This leads to the thought that even though "documents" usually are the unit of retrieval (cf. [5]), it does not necessarily mean that documents are *the* basic unit of a storing and indexing process. For instance, there might be transient (e.g. streamed RSS) forms of data that can not be encompassed by the document metaphor. On the other hand, single documents may contain many differently motivated informational items.
3) In the same way as in the previous point, knowledge may be distributed across several source documents (or streams), linguistic or ontological information does not need to be symbolic but can be spread across "meaning aspects" (as in [6]) or across concepts (as in e.g. neuronal networks, cf. [7]).
4) Knowledge by our means is seldom static or monotonous, especially in the context of textual data streams.

Another important question is what we try to mend when "searching for knowledge", i.e. what it means *not* to know (something):

Smithson constructed a taxonomy of ignorance (cf. [8]), based on the notions of *error* (bias, inaccuracy and confusion) and *irrelevance* ("mistaking some criterion as support for an argument when it has no bearing on its truth or falsehood"), based on his observations about the role of informational errors. His hypothesis is that not only knowledge is a socially constructed artifact, represented and communicated through symbolic frameworks, but that this is true for ignorance, too. He stated that in order to understand how and why people seek information, and how knowledge is linked with behavior, knowledge about both (perceived) relevance and irrelevance, objective knowledge and ignorance is fundamental.

We conclude from Smithsons findings, that in order to create an ultimately successful service for institutional information retrieval, at first the semantics of knowledge (and its absence) should be made explicit and clear.

So, where is that "semantics"? Where do we state ignorance and where should we create knowledge? When looking at web documents, for example, we see that they usually reflect an intentional act to represent and communicate *knowledge*. On the other hand, one common act to represent and communicate *ignorance* (as defined by Smithson) are search queries. Trivially, search engines use the first acts to solve issues of the latter ones - but there is one major problem: the documents are created prior to the queries; in terms of software development: they exist at *compile time* (when the search engine index is compiled). Queries, on the other hand, come into life at *run time*. Thinking about this, it does seem awkward that an answer[3] should exist prior to its question!

It is therefore our goal to resolve this curious situation.

III. Related Work

In this paper, we will concentrate on "informational" rather than on so-called "navigational" or "transactional" searches, i.e. searching for information rather than trying to reach a particular web page or perform some web-based action (this restriction is based on Broders "taxonomy of search", cf. [9]).

We also see an apparent solution to the issues explained so far in the "Semantic Search" research field[4]. There is a handful of related and distinct approaches, some of which we will introduce here:

Perpaolo Basile et al. attempt to enhance conventional syntax-based search through multi-level document representations (cf. [10]), the levels themselves being syntactic (keywords) and semantic (WordNet[5] synsets and named entities from pre-built lists). Their SENSE system is based on the GATE[6] architecture and uses distinct indexes for storing information from these levels but combines all levels in querying the index database. Unfortunately, their approach does not seem to outperform standard keyword level search in terms of recall and precision. Apart from that, their use of distinct indexes seems to give away any chances of creating synergies between the individual analytic methods.

An apparently more successful approach was presented by Fausto Giunchiglia et al. (cf. [13]); their prototype system also uses WordNet data in a GATE architecture but rather creates a "conceptual index" from natural language phrases linked to Wordnet concepts. The evaluation of their system showed a better precision/recall-performance (on a 29506-document corpus) than an Apache Lucene based search engine[7]. While the system allows for complex and precise queries, imprecise searches seem not to be covered - and even though query assistance is provided, user acceptance can be questioned.

Integrating predefined ontologies instead of simple WordNet concepts, Joan Codina et al. developed a multimodal search engine (cf. [14]), employing keyword and metadata search and also a semantic search approach in which they lemmatized, POS-tagged and parsed input documents. These informations were exploited to recognize ontology concepts, which in turn were used to annotate the indexed documents. Semantic

[3] "Answer" being a socially constructed artifact just as well

[4] Please note that we distinguish between "Semantic Searching the Web" and "Searching the Semantic Web" – a brief argumentation will follow in section IV.

[5] http://wordnet.princeton.edu, accessed 17.10.2008 – cf. [11]

[6] http://gate.ac.uk, accessed 17.10.2008 – cf. [12]

[7] http://lucene.apache.org, accessed 17.10.2008

queries are subsequently built through means of a wizard, selecting attributes from the background ontologies (SUMO[8] and others). The system itself is specialized in US-Patent discovery, which resembles - in our view - an enterprise search environment.

A more general approach was published by Duke et al. (cf. [15]) - their "Squirrel" system is also based on a specific ontology (in this case, the ontology allows to create user interest profiles which can be interpreted as pragmatic approach to ignorance modelling as described in section II) and integrates several semantic components.

Using quite a different methodology, Peter Mika presented "Microsearch" (cf. [5]), a Software developed at Yahoo! Research, which is able to enhance conventional search results by visualizing embedded metadata. The metadata Microsearch uses, stems from microformats, embedded in otherwise standard HTML pages. His approach explicitly assumes further advances (in terms of quantity of annotated web pages) in the Web 2.0 movement but he claims that even now Microsearch may motivate users to provide metadata for web pages thus bootstrapping its own data basis. Mika also proposes the positive effect of aggregating information from different result pages, e.g. by combining personal and geographic information.

While there is much more current related work worth to be discussed here, we decided to place the more pragmatic approaches next to ours in order to enable comparing sometimes subtle differences.

IV. A Semantic Approach on Search

While there still exists a situation of hidden knowledge and knowledge leaks, and the open access movement still appears to be in it's early stages, a conventional tool to look up institutional information is "enterprise search", i.e. web search technology, imposed on an intranet infrastructure.

Unfortunately, current search engine technology is mostly based on (syntactic) open web[9] search (cf. [13], [16], etc.), which in turn is based on common information retrieval techniques. These provide only basic tools, which are not very effective in a highly socialized and informationwise fine grained environment. Other tools, like link structure exploitation, also don't work too well here (cf. [17]). To be more specific: while intranet *recall* seems an issue of providing a highly customized technical solution, the *precision* of search results can - by definition - only be raised by tuning the search engines relevancy[10] algorithms.

Semantic Web methods on the other hand are well suited to shape indexed knowledge according to the real informational situaton and needs of institution members. Providing semantically rich machine readable information about resources and the principle of distributed extensibility are key aspects of the Semantic Web theory. Yet, one major drawback is that they still depend on a large amount of manual annotation work (sometimes it is simply assumed that the WWW will eventually contain appropriately annotated ressources, cf. [18]). This indeed is one well-known problem of knowledge engineering, that annotating text basically is a huge amount of work with no apparent use to the annotator himself. Even in cases where people apparently want to annotate text (e.g. via the so-called "Web 2.0" technologies, i.e. folksonomies and such) they do it rather in a way that they gain reputation in their respective community but not in order to provide semantic annotations for automatic information retrieval (cf. [5]). Because of this issue, we think that annotation must come from automatic methods, if they are to be employed on large volumes of data.

Furthermore, we like the view of Chakrabarti that schema-free searches must be enabled, but schema knowledge should be honored by a query language (this enables freetext keyword searches but still rewards complex processing, cf. [19]). While following his advice forbids using strict schematic query wizards or formalized query languages (as proposed in [12] or [14]), it reveals the necessity for applying NLP methods (and enabling manual editing) in order to discover semantic relationships within the data.

These thoughts, in combination with those given in section II, lead us to an approach that combines enterprise search with semantic web technology.

While this in general is not really a novel idea, we *will* add some new aspects to it, expecting to overcome the difficulties we described aforehand.

A. The Hypothesis

As stated in section III on the preceding page, we aim for *Semantic Search* but not for *Semantic Web Search* for the following reasons: Search is often limited to searching literal text *or* URI nodes and is implemented as specific function within a RDF framework (cf. [20] or [21], [22]). We feel that is an unnecessary limitation because search functionality and RDF framework functionality should be tightly and efficiently integrated. It also should be possible to integrate schema information and use description logics and such when required.

We propose that fusing search engine and semantic web technology at the right level, i.e. enabling semantic annotations and intra-institutionwise distributed extensibility – while maintaining freetext search functionality – will create a certain amount of synergy which can raise the effectiveness of a semantic search approach in an institutional (enterprise) environment. From our preliminary evaluation of some query logs of our institution we found that queries are strongly biased towards personal information (~28% of all queries) and organizational or structural queries, related to the institution (~36%), such as querying for departments, scripts, elearning courses, etc.. This enterprise-search related aspect of course will have a great impact on the kind of semantics we need to employ — especially named entity processing should be treated with high priority.

[8] http://www.ontologyportal.org/, accessed 17.10.2008
[9] i.e. extranet/WWW
[10] See section II on page 1

Furthermore, we think that the approach should be reasonably open to allow for other kinds of semantics, especially subsymbolic ones, like TSRs (cf. [6]).

B. Architecture and Implementation

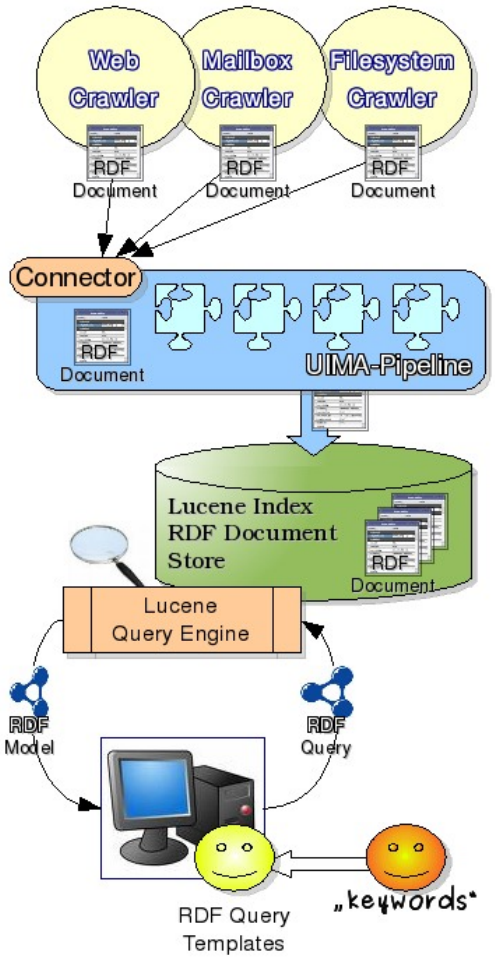

Fig. 1. Proposed Nebula5 Architecture

Our approach to semantic enterprise search is based on a distributed modular architecture, named *Nebula5* and shown in Figure 1:

1) Independent crawlers that syntactically transforms web, office, pdf and other documents into trivial RDF models (including metadata statements that are discovered by shallow processing, e.g. from HTML <meta>-tags). The crawlers are based on the aperture software (cf. [23], [21]) and up to now are little more than the aperture crawler part in combination with a web based data publishing process.
2) The generated models (and also RDF schemata) are retrieved from the crawlers and converted into so-called RDF documents by a UIMA-based processing pipeline (cf. [24]). Table I contains a description hereof, i.e. of Apache Lucene documents that contain (fragments of) RDF models and some metadata[11]. The pipeline is constructed according to the institutions informational needs and might contain modules for language detection, tokenization, pos-tagging, named entity recognition and such. The pipeline successively applies NLP and shallow techniques on the data in order to enrich the RDF models, e.g. by adding person or time data, extracted from the textual content. This course of action still is quite common, for example in searching emails (cf. [26]). Because search queries are social artifacts as well, they are being processed by the pipeline as well in order to enrich them with syntactic and semantic information that will eventually guide the query result evaluation process. One should note, however, that common (statistical) NLP tools will probably not work well on queries, because they usually consist of one or two words only (a preliminary experiment has shown that pos-tagging a corpus of about 1.500 queries resulted in an accuracy of less than 10% on single-word-queries).
3) Eventually, the RDF documents are stored in an unmodified Lucene index for subsequent query processing.

TABLE I
RDF DOCUMENT DEFINITION

Attribute	Example entry	Description
ID	*/pub/index.html*	Unique Identifier for given domain
DOMAIN	*http://www.uhh.de*	Domain Identifier
TIME	*200810152230232*	Last access time
DOCTYPE	*Document*	"Document", "Schema" (or anything else)
READ	*world*	Group names
WRITE	*webadmin staff*	Group names
PREFIX	*foaf: <http://xmlns.com/foaf#>*	Schema namespaces
TRIPLE	*:bob foaf:name "'Bob"'.*	RDF model serialized in (prefixed) N-Triples form and stored line-by-line as text

One of the main differences between our architecture and others (typically similar to the one described by Lei et al., cf. [27]) is this: while the common approach to freetext semantic search (and also to semantic query expansion, such as explained in [28] or [16]) aims to translate natural language queries/ keyword queries into formal expressions – which are subsequently used to search a model repository for matching RDF statements, we instead use conventional freetext queries on our RDF documents.

Another difference is that there is no need for a tight coupling of web documents and index items: a crawler might

[11]In addition to the RDF-related Triple fields, based on attribute-type semantics, other fields may be added in order to support further types of semantics. For example a TSR field can be added in order to allow for family resemblance words semantics (cf. [25]).

be constructed to split documents at certain points and merge others for several reasons, leaving references to the original pages only as in-document "rss:link"-properties. This allows for creating "views" on the data where each view shows a different semantical focus, a different interpretation of the content.

The most important key heuristic is hidden in the post-processing step of our architecture – by querying the index we encounter three cases:

1) Using conventional keywords only: documents containing these keywords will be discovered and ranked according to the Lucene $tf \times idf$ scheme. Additionally, RDF URI nodes can be discovered, too – exploiting the fact that most RDF URIs contain semantically relevant, human readable parts. For example, a keyword search for "bob homepage" will also reward indexed items containing "<foaf:homepage>" – especially when in conjunction to the literal fragment "bob". This can be quite useful, because many homepages in an institutional environment do not explicitly state that they are homepages!
2) Submitting a mixture of keywords and RDF URIs: queries like "foaf:homepage bob" will find "Bobs homepage" – but not "Jills homepage" with a reference to Bob! Because Lucene query analyzers eliminate non-alphanumeric characters, a domain-less URI is treated like a keyword; i.e. the query ":homepage bob" will not be restricted to the FOAF[12] domain but rather work like an ordinary keyword query in the above explained way. In the special case of web documents containing microformats such as RDFa[13] these will be implicitly honoured the same way.
3) Submitting RDF URIs only will exhibit documents with certain semantic properties: the query "foaf:homepage" will return all indexed items that contain homepages in the sense of the homepage element of the FOAF schema, plus the FOAF schema itself (as it also contains the fragment "foaf:homepage".

The query results (possibly filtered by a predefined document relevance threshold or by a first-N-documents-only heuristic) are merged into a single resulting RDF model that can be searched by means of templates, implemented structured RDF querying languages, e.g. SPARQL (cf. [29]), in order to provide end-user application functionality[14]. Most prominently, this will be the list of relevant links to web pages (fetched by applying a SPARQL search for "rss:link"-nodes), but a wealth of other applications is possible as well (for other examples, cf. e.g. [5]).

In this way, a query-centric RDF model is constructed dynamically on each search occasion that reflects the "ignorance-artifact" created by the user. Because schemata are discovered as well (and can be further tracked by using the PREFIX RDF document fields), we are not restricted to structured RDF queries only but can also apply description logics in order to further examine query results. For example, we can deduce subclassing etc. On the other hand, when it's just a portion of the textual content, it is being searched for, we can simply output the value of the "nie:content" predicate triple. In this way, we are able to defer complex processing until it is really needed.

C. An Example Scenario

Let us suppose, for instance, an information pool of public staff profiles should be created that provides research domain information. The system should answer questions that are semantically similar to "I am new to this university. which professor can i ask about topic X?"[15].

Our proposed system would employ a web crawler to harvest our university web documents. The resulting RDF documents are then run through a POS-tagger, a text boundary detector and a person entity tagger within the UIMA framework (and possibly more components). Subsequently, a set of person information extraction components (e.g. phone number extractors, etc.) is applied on the resulting data, thus creating new RDF documents that resemble person profiles[16] of the gathered data and links to the original web pages. Further components might collect bibliographic references associated with these people and search library databases for associated keywords. The keywords found are then consolidated into a short list and added to the respective persons profile. After indexing the profile RDF document, a user can search for these keywords in order to find people as well.

For example, when crawling the university website, we see that a person named Wolfgang M. was co-author of a paper titled "Hybrid parsing: using probabilistic models as predictors for a symbolic parser" – the university library database associates the following fictitious keywords to this article: *linguistics parsing stochastic grammar*, and other publications of this person may include the keywords: *analysis aspects reasoning artificial intelligence*. From his web page, we discover that Mr. M. is currently professor at the informatics department at the University of Hamburg. Creating a list of keywords from these publications and associating the most common ones with the author results in a set of author-related keywords. These can be added to the profile RDF document which in turn is associated with Mr. M.'s homepage. Other information (such as his profession) is added in the same way. A hypothetical user querying for "professor artificial intelligence linguistics" will then discover the (high ranked) homepage of Wolfgang M. even though it factually only contains the term "artificial".

[12] cf. http://xmlns.com/foaf/spec/, accessed 29.10.2008

[13] cf. http://esw.w3.org/topic/RDFa, accessed 29.10.2008

[14] These templates are supposed to be pre-built by staff members

[15] we don't aim for a natural language interface - we are just describing the question informally. a real search engine will require a more formal specification.

[16] Please note that an appropriate level of privacy and security must be maintained but discussing this is beyond the scope of this paper.

In this way, we hope to increase both recall and precision of returned search results.

V. Conclusions

In conclusion, we propose that our approach will enhance enterprise search through:

- the fusion of search engine and semantic web technology thusly introducing additional semantic and pragmatic features, like querying descriptions of objects but without the expense of giving away freetext keyword search. In this way, Chakrabartis insight is honoured, and furthermore, our approach will not perform worse than existing methods (because we use existing search technology at the most basic level).
- loosening the (unnecessary) tight coupling of web documents and search engine database entries, therefore enabling "views" on documents and creating "documentless" index entries.
- integration of schema knowledge in order to allow for information integration and use of description logics, e.g. in order to search for products or items with a given set of requirements
- end-user support by pre-build (staff-constructed) RDF queries enables specific search applications within a general framework on a semantically high level
- the principle of distributed extensibility, being implemented on document level
- flexible automatic annotation as a fundamental concept

Furthermore, the realization of our approach is quite straightforward and not very ressource-intensive because we make strong use of exisiting systems and technology.

VI. Future Work

In the future, we plan to work on the following items:

- completition of our prototype and turning it into a productive system. This encompasses the inclusion of further data sources such as internal and external library records, ontologies from semantic web search engines, etc
- construction of an evaluation scenario that can be used to evaluate our system against others and also report advancement.
- building an interface for manual editing of RDF document content, metadata and structure

References

[1] S. Harnad, "The implementation of the berlin declaration on open access," *D-Lib Magazine*, vol. 11, no. 3, March 2005. [Online]. Available: http://www.dlib.org/dlib/march05/harnad/03harnad.html

[2] P. Gruss, "Berlin declaration on open access to knowledge in the sciences and humanities," http://oa.mpg.de/openaccess-berlin/berlindeclaration.html, accessed October 2008.

[3] S. Raghavan and H. Garcia-Molina, "Crawling the hidden web," in *Proceedings of the 27th VLDB Conference*, 2001, pp. 129–138.

[4] S. W. Liddle, D. W. Embley, D. T. Scott, and S. H. Yau, "Extracting data behind web forms," in *Proceedings of the 28th VLDB Conference*, Hong Kong, China, 2002.

[5] P. Mika, "Microsearch: An interface for semantic search," in *Proceedings of the Workshop on Semantic Search (SemSearch 2008)*, Tenerife, Spain, June 2008, pp. 79–88.

[6] R. Winnemöller, "Constructing text sense representations," in *ACL 2004: Second Workshop on Text Meaning and Interpretation*, G. Hirst and S. Nirenburg, Eds. Barcelona, Spain: Association for Computational Linguistics, July 2004, pp. 17–24.

[7] G. Dorffner, *Konnektionismus: Von neuronalen Netzwerken zu einer "natürlichen" KI*, ser. Leitfäden der angewandten Informatik. Stuttgart: Teubner, 1991.

[8] M. Smithson, "Ignorance and science - dilemmas, perspectives, and prospects," *Science Communications*, vol. 15, no. 2, pp. 133–156, 1993.

[9] A. Broder, "A taxonomy of web search," *SIGIR Forum*, vol. 36, pp. 3–10, 2002.

[10] P. Basile, A. Caputo, A. L. Gentile, M. de Gemmis, P. Lops, and G. Semerar, "Enhancing semantic search using n-levels document representation," in *Proceedings of the Workshop on Semantic Search (SemSearch 2008)*, Tenerife, Spain, June 2008, pp. 29–43.

[11] R. Al-Halimi, R. C. Berwick, J. F. M. Burg, M. Chodorow, C. Fellbaum, J. Grabowski, S. Harabagiu, M. A. Hearst, G. Hirst, D. A. Jones, R. Kazman, K. T. Kohl, S. Landes, C. Leacock, G. A. Miller, K. J. Miller, D. Moldovan, N. Nomura, U. Priss, P. Resnik, D. St-Onge, R. Tengi, R. P. van de Riet, and E. Voorhees., *WordNet An Electronic Lexical Database*, C. Fellbaum, Ed. The MIT Press, 1998.

[12] H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan, "Gate: A framework and graphical development environment for robust nlp tools and applications," in *Proceedings of the 40th Annual Meeting of the ACL*, 2002. [Online]. Available: http://citeseer.ist.psu.edu/context/2035358/0

[13] F. Giunchiglia, U. Kharkevich, and I. Zaihrayeu, "Concept search: Semantics enabled syntactic search," in *Proceedings of the Workshop on Semantic Search (SemSearch 2008)*, Tenerife, Spain, June 2008, pp. 109–123.

[14] J. Codina, E. Pianta, S. Vrochidis, and S. Papadopoulos, "Integration of semantic, metadata and image search engines with a text search engine for patent retrieval," in *ESWC-SemSearch 2008*, Tenerife, Canary Islands, Spain, June 2008, pp. 14–28.

[15] A. Duke, T. Glover, and J. Davies, "Squirrel: An advanced semantic search and browse facility," in *4th European Semantic Web Conference (ESWC)*, Innsbruck, Austria, June 2007, pp. 341–355.

[16] T. Tran, P. Cimiano, S. Rudolph, and R. Studer1, *The Semantic Web*, ser. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2007, vol. 4825/2008, ch. Ontology-Based Interpretation of Keywords for Semantic Search, pp. 523–536.

[17] G.-R. Xue, H.-J. Zeng, Z. Chen, W.-Y. Ma, H.-J. Zhang, and C.-J. Lu, "Implicit link analysis for small web search," in *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*. New York, NY, USA: ACM, 2003, pp. 56–63.

[18] R. Guha, R. McCool, and E. Miller, "Semantic search," in *The Twelfth International World Wide Web Conference (WWW2003)*, Budapest, Hungary, May 2003.

[19] S. Chakrabarti, "Building blocks for semantic search engines: Ranking and compact indexing in entity-relation graphs," in *IIIA-2006: International Workshop on Intelligent Information Access*. Helsinki, Finland: http://cosco.hiit.fi/search/IIIA2006/chakrabrati.html, July 2006.

[20] Hewlett-Packard Development Company, "Larq - free text indexing for sparql," http://jena.sourceforge.net/ARQ/lucene-arq.html, accessed October 2008.

[21] L. Sauermann, G. A. Grimnes, M. Kiesel, C. Fluit, H. Maus, D. Heim, D. Nadeem, B. Horak, and A. Dengel, "Semantic Desktop 2.0: The Gnowsis Experience," in *Proc. of the ISWC Conference*, ser. Lecture Notes in Computer Science, vol. 4273/2006. Springer Berlin / Heidelberg, Nov 2006, pp. 887–900, iSSN 0302-9743 (Print) 1611-3349 (Online). [Online]. Available: http://www.dfki.uni-kl.de/~sauermann/papers/Sauermann+2006d.pdf

[22] E. Minack, L. Sauermann, G. Grimnes, C. Fluit, and J. Broekstra, "The sesame lucene sail: Rdf queries with full-text search," NEPOMUK Consortium, Technical Report 2008-1, February 2008. [Online]. Available: http://nepomuk.semanticdesktop.org/xwiki/bin/download/Main1/Publications/Minack%202008.pdf

[23] C. Fluit, L. Sauermann, and A. Mylka, "The use of rdf in aperture," http://aperture.wiki.sourceforge.net/RDFUsage, accessed October 2008.

[24] D. Ferrucci and A. Lally, "Uima: an architectural approach to unstructured information processing in the corporate research environment," *Nat. Lang. Eng.*, vol. 10, no. 3-4, pp. 327–348, 2004.

[25] R. Winnemöller, "Using meaning aspects for word sense disambiguation," in *9th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing)*, Haifa, Israel, February 2008.

[26] S. Raghavan, R. Krishnamurthy, E. Kandoga, Y. Mass, N. Har'El, M. Bluger, H. Zhu, G. Ramakrishnan, K. Sah, and S. Vaithyanathan, "Ibm omnifind personal e-mail search," http://www.alphaworks.ibm.com/tech/emailsearch, accessed October 2008.

[27] Y. Lei, V. S. Uren, and E. Motta, "Semsearch: A search engine for the semantic web," in *Knowledge Acquisition, Modeling and Management (EKAW)*, S. Staab and V. Svátek, Eds., Podebrady, Czech Republic, October 2006, pp. 238–245.

[28] J. Umbrich and S. Blohm, "Exploring the knowledge in semi structured data sets with rich queries," in *Proceedings of the Workshop on Semantic Search (SemSearch 2008)*, Tenerife, Spain, June 2008, pp. 89–101.

[29] S. Schenk, "A sparql semantics based on datalog," in *KI 2007: Advances in Artificial Intelligence*. Springer, 2007, pp. 160–174. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-74565-5_14