LIBER QUARTERLY
THE JOURNAL OF THE ASSOCIATION OF EUROPEAN RESEARCH LIBRARIES

Vol. 22, no. 4 (2012) 285–309 | ISSN: 1435-5205 | e-ISSN: 2213-056X

The Searchbench — Combining Sentence-semantic, Full-text and Bibliographic Search in Digital Libraries

Ulrich Schäfer

German Research Center for Artificial Intelligence (DFKI),
Saarbrücken, Germany,
ulrich.schaefer@dfki.de

Bernd Kiefer

German Research Center for Artificial Intelligence (DFKI),
Saarbrücken, Germany,
kiefer@dfki.de

Christian Spurk

German Research Center for Artificial Intelligence (DFKI),
Saarbrücken, Germany,
Christian.spurk@dfki.de

Jörg Steffen

German Research Center for Artificial Intelligence (DFKI),
Saarbrücken, Germany,
steffen@dfki.de

Rui Wang

German Research Center for Artificial Intelligence (DFKI),
Saarbrücken, Germany,
wang.rui@dfki.de

Igitur publishing | http://liber.library.uu.nl/ | URN:NBN:NL:UI:10-1-114289

Benjamin Weitz

German Research Center for Artificial Intelligence (DFKI),
Saarbrücken, Germany,
benjamin.weitz@dfki.de

Magdalena Wolska

Computational Linguistics, Saarland University, Saarbrücken, Germany,
magda@coli.uni-saarland.de

Abstract

We describe a novel approach to precise searching in the full content of digital libraries. The Searchbench (for search workbench) is based on sentence-wise syntactic and semantic natural language processing (NLP) of both born-digital and scanned publications in PDF format. The term born-digital means natively digital, i.e. prepared electronically using typesetting systems such as LaTeX, OpenOffice, and the like. In the Searchbench, queries can be formulated as (possibly underspecified) statements, consisting of simple subject-predicate-object constructs such as `'algorithm improves word alignment'`. This reduces the number of false hits in large document collections when the search words happen to appear close to each other, but are not semantically related. The method also abstracts from passive voice and predicate synonyms. Moreover, negated statements can be excluded from the search results, and negated antonym predicates again count as synonyms (e.g. not include = exclude).

In the Searchbench, a sentence-semantic search can be combined with search filters for classical full-text, bibliographic metadata and automatically computed domain terms. Auto-suggest fields facilitate text input. Queries can be bookmarked or emailed. Furthermore, a novel citation browser in the Searchbench allows graphical navigation in citation networks. These have been extracted automatically from metadata and paper texts. The citation browser displays short phrases from citation sentences at the edges in the citation graph and thus allows students and researchers to quickly browse publications and immerse

into a new research field. By clicking on a citation edge, the original citation sentence is shown in context, and optionally also in the original PDF layout.

To showcase the usefulness of our research, we have a applied it to a collection of currently approx. 25,000 open access research papers in the field of computational linguistics and language technology, the ACL Anthology (http://aclweb.org/anthology). The Searchbench user interface is a web application running in every modern, JavaScript-enabled web browser, also on smart phones and tablet computers. The system is a free and public service at http://aclasb.dfki.de. Because the NLP technology is domain-independent, it could also be applied to newspaper texts, technical documentation, or scientific publications from other disciplines. The aim of this paper is to make the benefits of this new, language technology based approach known in library research and related fields.

This article summarises 9 peer reviewed publications from the past three years that have been published in international conferences and workshops in the area of computational linguistics, and tries to present them in an appropriate way to the LIBER audience. The original papers contain more details and are freely available from the author's homepage[1] or via the Searchbench[2].

Key Words: sentence-semantic search; natural language processing; citation browser

1. Introduction

Searching in the ever and faster increasing amount of digitally available publications is tedious and often unsatisfactory. The main reason is that a search for keywords often delivers too many unspecific or unrelated results. Natural language processing can help in making a search more precise and efficient. In this article, we summarise our research that has been conducted over the last three years on precise searching in digital scientific libraries by using natural language processing, viz. deep syntactic parsing with sentence semantic output. The research also led to a practical system, the Searchbench, and a free online service, the ACL Anthology Searchbench[3], that can be used to test the research results and benefits for search in scholarly publications (Schäfer, Kiefer, Spurk, Steffen, & Wang, 2011). The approaches are domain-independent and thus can also be applied to other text domains as long as edited text with well-formed English sentences is predominant.

The article is structured as follows. In Section 2, we describe the common preprocessing of the PDF documents to extract the publication texts at high quality and recover document structures. Section 3 explains basic NLP terms and summarises the natural language parsing architecture that was developed for semantically analysing every sentence in the document collection. In Section 4, we present an overview of the key feature of the Searchbench: sentence-semantic search (statements search) and the user interface of the Searchbench. Further research on automatic terminology, taxonomy and glossary extraction from scientific text is presented and discussed in Section 5. Citation analysis and the graphical citation browser are presented in Section 6. An application-oriented evaluation that proves the usefulness of sentence-semantic analysis is discussed in Section 7. Section 8 discusses related work. We conclude and present an outlook in Section 9.

2. Rich Text Extraction with Logical Document Structure

Input to the Searchbench and its various text analysis and processing workflows for parsing, citation analysis, terminology extraction etc. are PDF documents and associated bibliographic metadata. The motivation is to provide uniform access to the paper's content and structure, be they born-digital or scanned. This allows to index older, digitised publications as well as recent ones generated using typesetting software such as OpenOffice, LaTeX, etc. that directly produce PDF files. The method has been shown to work successfully even for type-written scientific papers from the early 1960s.

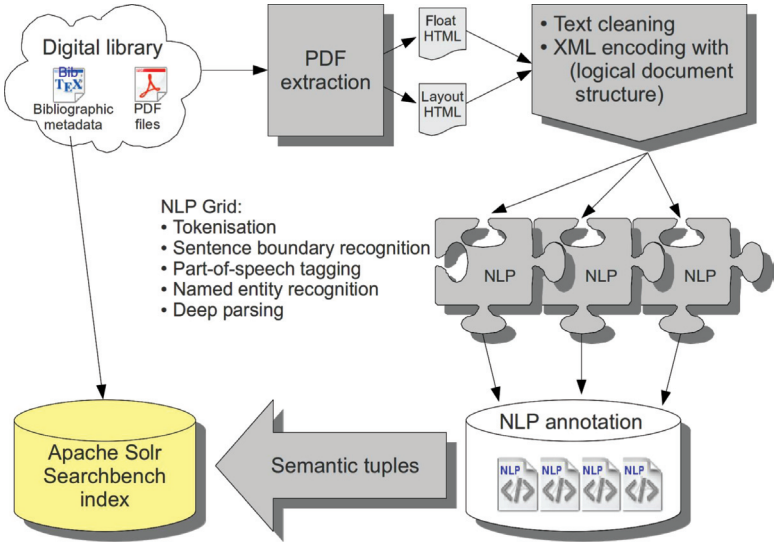
The text extraction is based on commercial OCR (optical character recognition) software that operates on the PDF documents directly.

The focus of the Searchbench text extraction process was to retrieve complete sentences from scientific papers for NLP analysis. Hence distinguishing running text from section headings, figure and table captions, tables or footnotes is an important intermediate task. PaperXML is a simple logical document markup structure we specifically designed for scientific papers. It features tags for section headings (with special treatment of abstract and references), footnotes, figure and table captions. The full DTD is given in Schäfer and Weitz (2012). In paperXML, tables with their layout and character style information such as boldface or italics are preserved.

The result of the OCR product comes in two different output formats, *layout* and *float*, that in parts contain complementary information. Our extraction algorithm uses the layout variant as primary source. *Layout* tries to render the extracted text in HTML as closely as possible to the original layout. It preserves page breaks and the two-column layout that many conference papers are formatted with. In the *float* variant, page and line breaks as well as multiple column layout are removed in favour of a running text in reading order, which is indispensable for our NLP to function properly. However, some important layout-specific information such as page breaks is not available in the float format. Both variants preserve table layouts and character style information such as boldface or italics. Reading order in both variants may differ. The code ensures that no text is lost when aligning both variants and generating the consolidated XML condensate, paperXML. It interprets textual content, font and position information to identify the logical structure of a scientific paper.

The upper part of Figure 1 depicts the overall workflow. The output is used to feed NLP components such as taggers, parsers or term extraction for the

Fig. 1: Searchbench offline processing from PDF-to-XML extraction to semantic index generation.



Searchbench's index generation. The extraction algorithm initially computes the main font of a paper based on the amount of characters with the same style. Based on this, heuristics allow to infer styles for headings, footnotes etc. While headings typically are typeset in boldface in recent publications, old publications styles, e.g., use uppercase letters.

On the basis of such information, special section headings such as abstract and references are inferred. Similarly, formatting properties are used to identify figure and table captions, etc. and generate corresponding markup. A special element is inserted for text fragments that do not look like normal, running text. The implemented preprocessing workflow and the paperXML format is described in more detail in Schäfer and Weitz (2012). Berg, Oepen and Read (2012) and Schäfer, Read and Oepen (2012) also propose and discuss an alternative approach for high precision extraction from born-digital PDF files. Its main advantage is 100% error-free character recognition.

3. Deep Parsing of Scientific Paper Content

In this Section, we describe the natural language processing analysis of scientific papers underlying the Searchbench index. It is illustrated in the middle and lower part of Figure 1. The general idea of the semantics-oriented access to scholarly paper content is to apply NLP analysis to each sentence and distill a structured semantic representation per sentence and subclause that can be searched for, in addition to fulltext. Various levels of analysis such as part-of-speech tagging, named entity recognition, chunking, shallow and deep parsing are suitable for the task.

We briefly explain some basic NLP concepts. The simplest tools, also mostly used for full-text indexing in traditional search indices, are tokenisers and stemmers. A tokeniser separates words and punctuation into distinct units and it may also assign classes such as number or uppercase word. A stemmer abstracts from morphological variants such as the plural suffix "s" of nouns. This makes full-text searching more tolerant.

A part-of-speech (PoS) tagger assigns word classes to each word in the input; adjectives, nouns, verbs, prepositions, determiners, pronouns, adverbs, etc. Because not all words are known in a general lexicon, the PoS tagger also

has the important task to guess the type of unknown words using statistical language model and context (words left and right).

A named entity recogniser (NER) is a special tagger for recognising specific types of open class words such as person, organisation, company or product names, locations, and time expressions. They often bear important semantic information.

A chunker combines multiple words such as "the green book", a noun phrase, into phrase units, also named "chunks".

A parser then builds on the results of the pre-processing taggers and a lexicon and syntactically analyses the sentence structure (syntactic subject, verb, objects, and their connections). Based on an information-rich lexicon, the deep parser used in our NLP pipeline in addition also computes a sentence-semantic representation which many mainstream shallow parsers are unable to provide.

Other terms of natural language processing are e.g. explained in the glossary of NLP terms[4], or one can use the statements query `'s:<Term> p:is'` in the ACL Anthology Searchbench (explained below) to find definitions for a term in the paper collection. Introductions to natural language processing are given, e.g., in the books by Jurafsky and Martin (2008), and Bird, Klein and Loper (2009). The latter is also available online[5] along with an open source package that allows to try basic NLP tools.

The core of the sentence-semantics index generation is the deep parser PET (Callmeier, 2000) operating the open-source ERG grammar of English (Flickinger, 2002). The ERG not only handles detailed syntactic analyses of phrases, compounds, coordination, negation and other linguistic phenomena that are important for extracting semantic relations, but also generates a meaning representation of the input sentence in a format that resembles first-order predicate logics from which the (simplified) predicate-argument structure is then derived.

To make the deep parser robust, it is embedded in a hybrid NLP workflow starting with a tokeniser, a part-of-speech tagger, and a named entity recogniser. These components help to identify and classify open class words such as person names, events (e.g. conferences) or locations. The trigram-based

tagger helps to guess part-of-speech tags of words unknown to the deep lexicon. For both unknown words and named entities, generic lexicon entries are generated in the deep parser. Ambiguities resulting in multiple readings per input sentence are ranked using a statistical parse ranking model.

We skip further details in order to remain appropriate for the audience. Further research results and technical details of the deep parsing and semantics extraction process are described in Schäfer and Kiefer (2011) and Schäfer *et al.* (2011), and Schäfer (2006). The system as of November 2012 contained approx. 25,000 papers published between 1965 and 2012, with a total of 106,296,773 tokens and 4,896,493 sentences. About 70% of the papers were scanned PDFs published before the year 2000. The rest are born-digital PDFs mostly published starting from the year 2000. Every year, the ACL Anthology grows by approx. 1700–2700 papers.

The shallow part of the pre-processing (including PoS tagging) is also shared with the preprocessing for the citation browser (described in more detail in Section 6) and terminology, taxonomy and glossary extraction (Section 5).

4. Sentence-Semantic Search and Searchbench User Interface

The idea of a sentence-semantic search is to search for similar *statements* in text instead of (or in combination with) keywords or phrases. A statement query is formulated by a possibly underspecified statement, expected to occur similarly in the text. A statement may consist of simple subject-predicate-object constructs such as `'s:semantics p:helps r:retrieval'`, where `'s:'` indicates subject, `'p:'` the predicate and `'r:'` stands for rest (direct and indirect objects, adjuncts, etc.). The sample query matches sentences such as "*More than this, (Schutze & Pedersen, 1995) performed experiments which have shown that semantics can actually help retrieval performance.*"

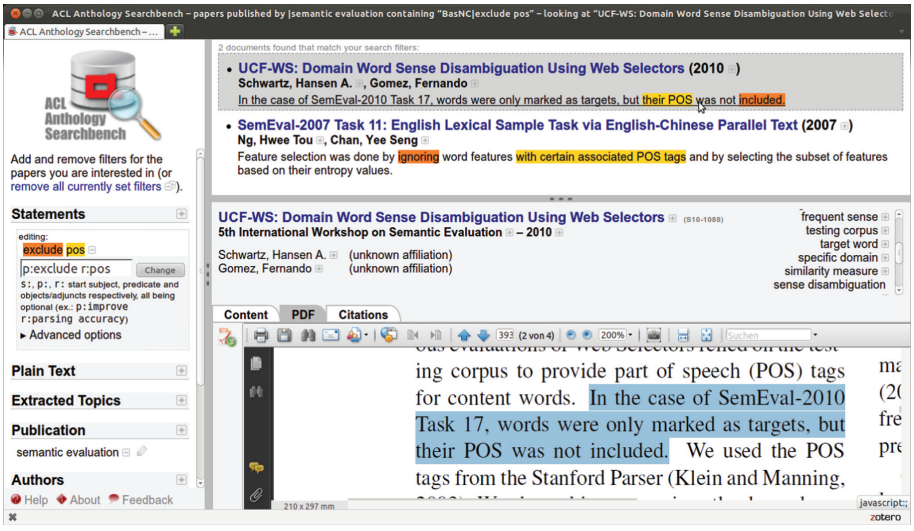
Parts of a query can be omitted, e.g. `'improve search efficiency'` (which is the canonical search format, equivalent to `'p:improve r:search efficiency'`) matches any subject. Even the predicate can be omitted: `'s:Peter r:Mary'` matches any sentence where Peter is subject and Mary is object, and can be used to find the relations between Peter and Mary expressed in the text.

Sentence semantic searching reduces the number of false hits when words happen to appear close to each other, but are not semantically related. By applying full, deep parsing as done in the Searchbench, it is also possible to abstract from passive variants and synonyms, e.g., a search for `'method reduce noise'` would also find sentences of the form `'noise was decreased by …method'`.

Moreover (and by default), negated statements are excluded from the search results, e.g. for the former query example, `'method does not reduce noise'` would be eliminated. Similarly to synonyms, antonyms in conjunction with negation would count as positive statements (`'does not increase = decrease'` which is of course not fully equivalent). For computing verb synonyms and antonyms, we use synsets from WordNet (Fellbaum, 1998) intersected with the most frequent verbs in the full paper corpus.

The Searchbench user interface is a web application running in every modern, JavaScript-enabled web browser. As can be seen in Figure 2, the display is divided into three parts: (1) a sidebar on the left (Filters View), where different filters can be set that constrain the list of found documents; (2) a list of

Fig. 2: Searchbench search for statement "exclude POS", semantically equivalent "POS was not included" is found.



found documents matching the currently set filters in the upper right part of the window (Results View); (3) the Document View in the lower right part offers different views of the current document.

A focus in the user interface design has been to allow the user to very quickly browse the papers of the ACL Anthology and then to find small sets of relevant documents based on metadata and content. Changes in the collection of filters automatically update the Results View. Metadata and searchable content from both the Results View and the Document View can easily be used with a single click as new filters. Filters can easily be removed with a single click. Manually entering filter items is assisted by auto-suggestions computed from the corpus. Accidental filter changes can easily be corrected by going back in the browser history.

The following filter types are supported:

- *Statements:* filter by semantic statements as described above. There are two ways in which a new statement filter can be set: (1) entering a statement manually; (2) clicking a sentence in the Document Content View and choosing the statements of this sentence that shall be set as new statement filters, i.e. it is possible to formulate and refine queries 'by example'.
- *Keywords:* filter by simple keywords with a full-text search (phrases or token-wise).
- *Extracted Topics:* filter by topics of the articles that were extracted with the unsupervised term extraction; Section 5.
- *Publication metadata* such as filter by title, event (conference name), author name(s), publication year, affiliation institution, or affiliation location and country.

Found papers always match all currently set filters. For each filter type, multiple different filter items can be set; one could search for papers written jointly by people from different research institutes on a certain topic, for example. Matches of the statements filter and the keywords filter are highlighted in document snippets for each paper in the Results View and in the currently selected paper of the Document View.

Besides a header displaying the metadata of the currently selected paper (including the automatically extracted topics on the right), the Document View provides three subviews of the selected paper:

- the Document Content View is a raw list of the sentences of the paper and provides different kinds of interaction with these sentences (inspection of extracted semantic structure, in original PDF context, compose new query from selected sentence);
- the PDF View shows the original PDF version of the paper;
- the Citations View provides bibliography information and citation information including a link to the the graphical Citation Browser described in Section 6.

The overall system is described in more detail in Schäfer *et al.* (2011).

5. Terminology, Taxonomy, and Glossary Extraction

The current version of the Searchbench features a filter called *Extracted Topics.* These topics are domain-specific multi-word terms that were fully automatically extracted for each paper. The same method is also applied globally for all papers to define a large super set of possible terms. This set acts as filter to eliminate noise (unwanted, rare terms) in individual papers. Up to 10 terms are shown in the Document View of a paper in the user interface. The selection is done according to the best rankings based on a *termhood* measure that is computed during the term computation.

The extraction process does not require domain knowledge or domain-specific resources. The approach is an elaborated and extended variant of the C-value/NC-value method (Frantzi, Ananiadou & Mima, 1998) using mostly statistical properties such as frequencies, word co-occurrence, containment of longer multi-words in shorter ones, and part-of-speech tags with generic patterns over them. To this aim, the extraction process uses the Searchbench's NLP preprocessing results such as PoS tagging.

An impressionistic evaluation and user feedback by various domain experts shows that most of the automatically extracted terms are good domain terms and also are mostly relevant for the per-paper *Extracted Topics* field. Given

the number of papers, a manual correction is neither feasible nor wanted. However, applying additional filters to remove unwanted terms such as *important task* could certainly further improve the user satisfaction.

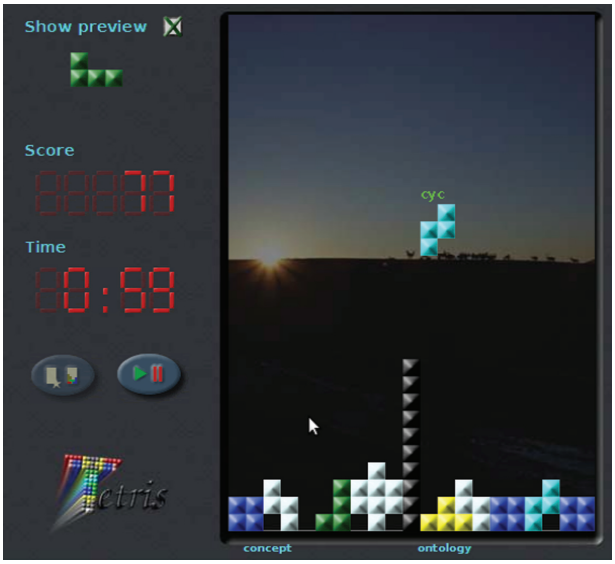
Based on the extracted multi-word domain terms, several future extensions to the Searchbench are possible. We briefly discuss three that have been implemented and investigated research-wise with our extracted texts, but are not yet integrated in the Searchbench user interface. *Single word domain terms* can be easily extracted as part of multi-word domain terms. This has already been done to obtain additional terms for the two other extensions, *domain taxonomy extraction* and *domain glossary extraction* from the texts. Moreover, *abbreviations* can be easily extracted from the sentences along with their full wording by applying patterns with parentheses and filtering by frequency.

A taxonomy is a collection of terms arranged in a hierarchy that reflects the semantic relation between terms with respect to generality and specificity, e.g., a car is a vehicle, etc. Automatic domain taxonomy extraction on the basis of the above described approach and Searchbench text extraction could support several interesting applications, including the possibility to use extracted hypernyms (more general terms), hyponyms (more specific terms) and derived synonyms for tolerant semantic search. To this aim, we used 241,806 automatically extracted multi-word terms as input (definition "anchors"). Furthermore, we extracted approx. 10,000 concept pairs in is-a relations (definitions) from the sentences using simple definition phrase patterns such as "TERM2 is a TERM1", "TERM1 such as TERM2", etc. The idea has first been decribed by Hearst (1992).

Correctness of the process has been verified by crowd-sourcing: To attract domain experts that would identify correct and invalid is-a pairs, we used "games with a purpose". The popular games of Tetris (Figure 3) and Invaders were modified to support concurrent and efficient annotation of domain term pairs during playing. High quality of the resulting annotations was ensured by exploiting redundancy: at least five-way agreement was required for a candidate is-a pair to be considered correctly extracted. Based on the crowd-sourced evaluation, the extraction method achieved a precision around 80%.

A simple taxonomy is created from the set of extracted is-a pairs by iterating over the list of candidates ordered by their reliability as follows: 1) add an is-a pair from the reliability-ordered list temporarily to the taxonomy,

Fig. 3: Tetris as game with a purpose.



2) test for cycles in the taxonomy, 3) if a cycle is found, remove the temporarily added pair, otherwise keep it in the taxonomy and proceed with the next pair. Cycles need to be removed because the definition of a term should never recur to itself, not even over several other term definition 'hops'. The complete approach including the Games with a purpose evaluation is described in Wolska, Schäfer and Pham (2011).

Automatic glossary extraction from text is similar to taxonomy extraction, but it does not necessarily involve another domain term on the right-hand side of a definition sentence. It could also be an appraisal, an opinion or a hint. The glossary consists in a collection of sentences that describe a term. Typically, several similar descriptions for a term can be found in a larger text collection from the same domain, i.e. redundancy could be exploited. Again, we used the extracted multi-word domain terms as anchors and *definienda*. We developed a 'shallow' approach using lexico-syntactic patterns specifically tailored to find glossary sentences. The patterns were extended on the corpus using a bootstrapping approach, i.e. extracted glossary sentences are used to derive new patterns that in turn deliver more glossary sentences and so on.

We compared the results with those of a less task-specific, completely different approach, using a Searchbench query on the same texts. The query was formulated by sending the statements query `'s:<Term> p:is'` to the Searchbench. Surprisingly, although simpler in its idea, the second approach did not perform worse in an evaluation by domain experts who blindly judged 200 extracted sentences (100 for each method), i.e. without knowing which method was used, using a 5-point scale. An observation and explanation from the detailed results is that the deep approach delivers more precision and less noise. Consequently, future research should address a combination of both approaches, i.e., applying further patterns to the deep approach.

Further details of the glossary extraction implementations and evaluations can be found in Reiplinger, Schäfer and Wolska (2012).

6. Citation Analysis and Citation Browser Application

Citations are important means to structure the vast amount of scientific publications. They are of invaluable importance to beginners in a scientific field as they ultimately point to seminal, original work and knowledge not explicitly available or repeated in every publication. Citations are also the primary discourse links in scientific discussions which typically span over years or even decades. Furthermore, citations are helpful to understand and reproduce findings. Thus, they form a predominant feature for every reader.

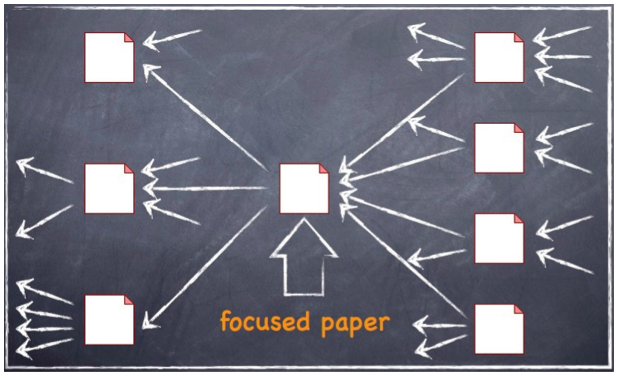
Therefore, besides textual metadata search for fields such as author, title, conference/publication, year and affiliation, the search interface is equipped with a graphical citation browser. The system is based on HTML5 and, as the rest of the Searchbench, also works on mobile devices and tablet computers. The citation browser not only supports quick graphical navigation, but also displays keywords from citation sentences to indicate citation types (Figure 6). By clicking on an edge between citing and cited documents, the original citation sentence(s) is/are shown in context, and optionally also in the original PDF layout (the latter requires Acrobat reader and Firefox on a Windows or Linux operating system).

The overall preprocessing workflow is as follows. The initial input comes from the Searchbench text extraction process described above. The raw text is input to the CRF-based citation reference matcher ParsCit (Councill, Giles, & Kan, 2008). ParsCit finds citations in running text and tries to link them to bibliographic references listed at the end of an article. Each sentence containing a citation plus up to three previous and subsequent sentences is then aligned with the corresponding ParsCit's XML output. These sentences can later be inspected in the citation context view in the user interface on the basis of a XML-to-HTML transformation (Figure 7). The citation graph is computed on the basis of ParsCit output and on the bibliographic metadata of the papers to be indexed. The complete graph for 22,500 papers contains approx. 125,000 nodes and about 305,000 edges.

In the visible citation graph, each node represents a paper and the edges represent citation relations between the papers. The layout algorithm is a variant of the fan-out algorithm described by Schäfer and Kasterka (2010). It always has one paper in the centre, and cited papers left and citing papers on the right-hand side, with arrow heads indicating the citation direction (Figure 4).

The advantage of the fan-out layout is that it avoids overlapping vertical edges in case a citing paper cites another paper that also cites the paper in the centre (analogously for cited papers). In such a case, the graph is expanded horizontally to provide space for the intermediate node, instead of arranging it

Fig. 4: Focused paper in the centre, cited papers on the left, citing papers on the right.



vertically (Figure 5). In addition, the citation depth for citing and cited papers can be modified in the user interface by adjusting the number with a slider.

Before drawing the graph, the positions of edges and nodes are rearranged according to the fan-out constraints. In addition, another modification replaces straight edges by Bézier curves (Bézier, 1968) in order to avoid overlapping (mostly horizontal) edges. This makes it easier to select labelled edges for inspection of the citation context (described below).

A screenshot is displayed in Figure 6. The nodes show meta-information (first author, year of publication, publication ID) about the papers they represent and, when hovering them with the mouse, further information such as full author list, title and conference are displayed in a pop-up box.

Clicking on a node brings the respective paper into the centre with its local citation graph. We used the rule-based classifier from Schäfer and Kasterka (2010) and extract the keywords or phrases that would have led to a classification into a citation class (but do not compute the class itself as mentioned above). In case no pattern matched, the main (finite) verb of the citation sentence is determined using the statistical part-of-speech tagger. The resulting keyword is displayed as edge label of the citation link.

Fig. 5: Fan-out layout: avoid crossing edges caused by citations on the same level.

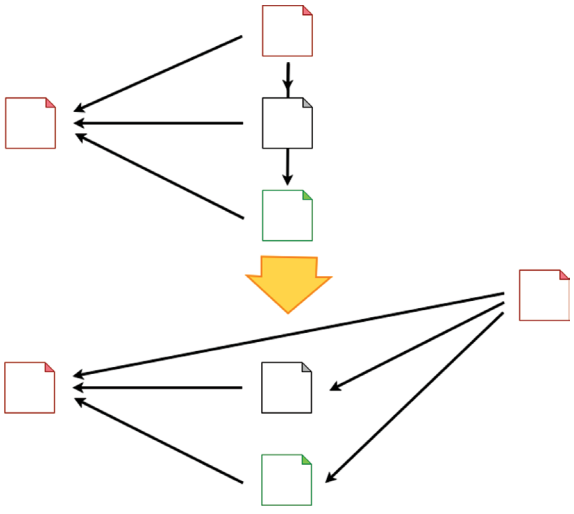
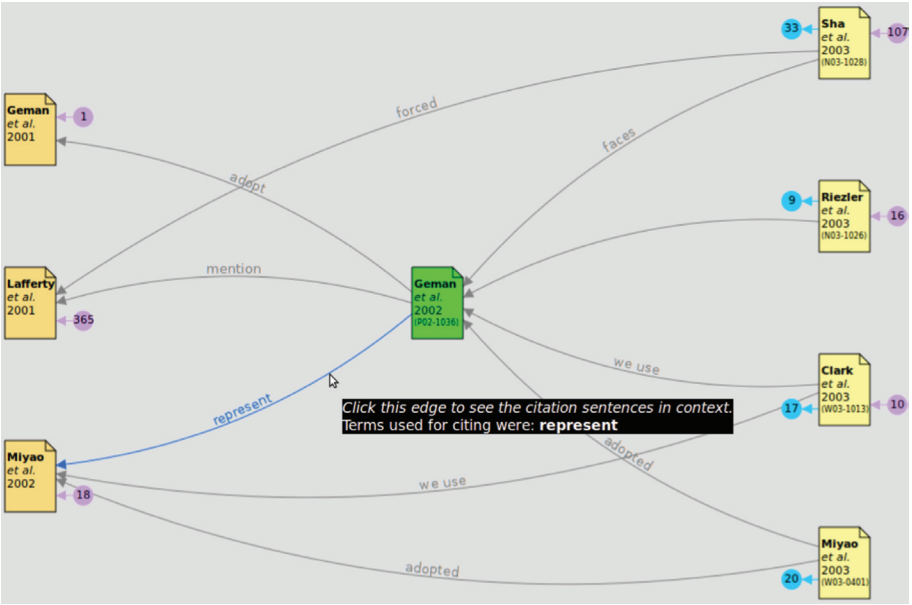


Fig. 6: Citation browser. By clicking on an edge, a citation context viewer opens. It displays the citation sentence(s) in context, optionally also highlighted in the original PDF (in Acrobat Reader).

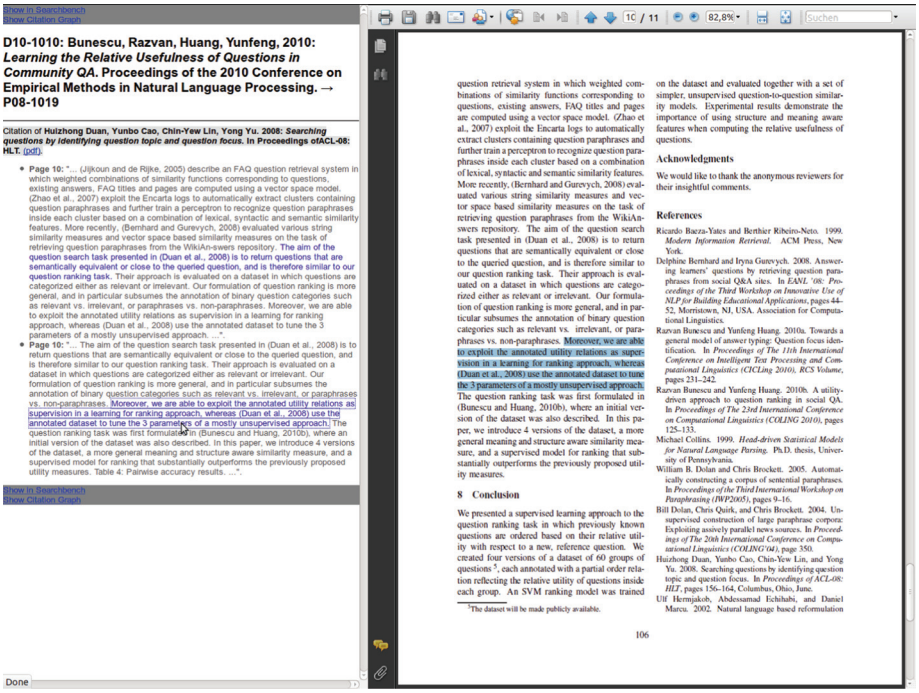


Multiple citations to the same reference are not shown on the edge label, but will be enumerated when the user moves the mouse over an edge. By clicking on an edge, the user can inspect the corresponding citation sentence in context — in most cases even highlighted in the original PDF layout (Figure 7).

One can use the time range slider on the bottom of the user interface to reduce the size of the graph, limiting it to publications in the specified time range. For larger graphs, only a filtered graph is shown by default, because larger graphs can be unclear and confusing and can take long to load on slow systems.

Filtering is done so that the highest possible number of papers below a configurable threshold is displayed using papers from the years around the year of the centred paper. The user can then choose to display a larger graph by using the time range slider.

Fig. 7: Citation context view.



More details can be found in Schäfer and Kasterka (2010), Weitz and Schäfer (2012). An approach to automatic citation classification into pre-defined categories such as use, refutation, etc., is discussed in Dong and Schäfer (2011). However, as the automatic classification is based on machine learning techniques that make errors, the results are not shown in the public Searchbench, but could in principle be integrated, e.g. as edges coloured according to the their citation category.

7. Evaluation

An information retrieval-like evaluation for a sentence-semantic search is hard to accomplish, and would compare apples and oranges. The reason is that the result sets differ considerably depending on the query structure, and

precise semantic queries are not available for the baseline, simple full text-based search. The general observation is that a sentence-semantic search often delivers precise results with a low percentage of unrelated results. However, to meet the users' expectations, query reformulations are sometimes necessary, because too few or too many results are shown.

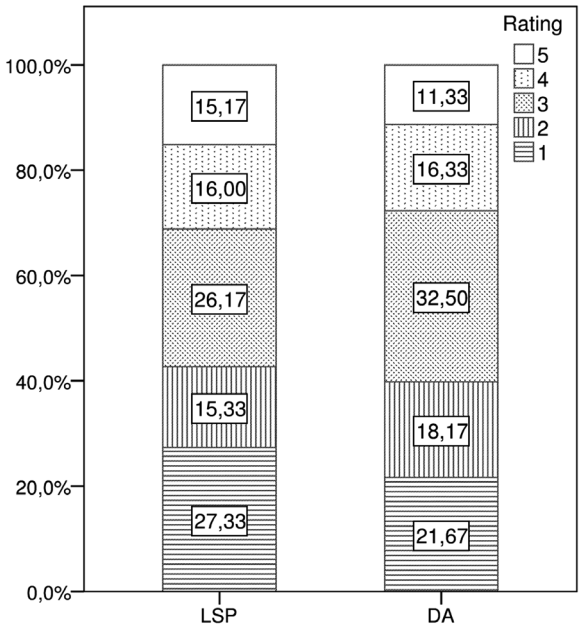
Because of this rationale, we decided to go for a different kind of evaluation that addresses a specific search task, namely finding glossary sentences (mostly definition) in the text. For this scenario, we can directly compare sentence-semantic searches using deep analysis with an alternative approach based on lexico-syntactic patterns (LSP) that is specifically tailored to the task, but does not involve sentence-semantic analyses. It has already been sketched near the end of Section 5 above. As it turns out, the overall evaluation result is equally good for the Searchbench's deep analysis (DA) approach. This observation is made in spite of the fact that for the DA approach, only a single pattern is used (`'s:<Term> p:is'`), while the LSP approach relies on a list of 20 different patterns.

Candidate definition sentences were presented to 6 human domain experts by a web interface displaying one sentence at a time in random order. Judges were asked to rate sentences on a 5-point ordinal scale with the following descriptors:

5: The passage provides a precise and concise description of the concept
4: The passage provides a good description of the concept
3: The passage provides useful information about the concept, which could enhance a definition
2: The passage is not a good enough description of the concept to serve as a definition; for instance, it's too general, unfocused, or a subconcept/superconcept of the target concept is defined instead
1: The passage does not describe the concept at all.

The evaluation results are depicted in Figure 8. Around 57% of the LSP ratings and 60% of DA ratings fall within the top three scale-points (positive ratings) and 43% and 40%, respectively, within the bottom two scale-points (low ratings).

Fig. 8: Evaluation Results for a Lexico-syntactic pattern (left) and a Searchbench's deep analysis (right).



This specific evaluation demonstrates the big potential offered by sentence-semantic searching. The pattern-based and deep glossary extraction approach as well as evaluation results are presented in Reiplinger *et al.* (2012).

8. Related Work

Text mining, natural language processing and citation analysis for scientific publications have been discussed and addressed by many research groups since the seminal work of Garfield (1965). While most groups either only address meta-information such as bibliographic data for citation analysis, or do text mining in abstracts or content (Ananiadou, 2007), the Searchbench is probably the first system that addresses both text analytics with a semantic index as well as citation analysis and search based on a common pre-processing and an integrated user interface on a non-trivial document collection.

Garzone (1996), DiMarco, Kroon and Mercer (2006), and Teufel, Siddharthan and Tidhar (2006) address citation function analysis based on citation sentences and their context.

The only approach, to our best knowledge, that applies deep linguistic parsing to generate a structured semantic search index as we do, is the Medie/Info-Pubmed system (Ohta *et al.*, 2010), but operating on Medline abstracts only, while we address the full content of research papers, including citations and references.

Many webportals for scientific publications such as those from publishers or whole disciplines (e.g. DBLP for computer science) only provide fulltext or bibliographic search. However, extracted additional information such as reference lists are often corrected manually. In contrast, our portal automatically processes the papers, and no manual correction took place.

9. Summary and Outlook

We have presented the *Searchbench*, a new approach for precision-oriented text search and graphical citation link navigation in digital (scientific) libraries.

The ACL Anthology Searchbench is a 24/7 service freely available on the web. It offers combined sentence-semantic, bibliographic metadata and fulltext search in the complete ACL Anthology, a collection of thousands of research publications from the past 50 years. The system works on every modern web browser including smartphones and tablet computers supporting JavaScript and HTML5.

The natural language processing technology underlying the current system is fully domain-independent and thus can also be applied to publications in other research areas or other kinds of well-edited texts such as technical documentation, legal and newspaper texts, etc.

Besides search, the sentence-semantic indexing can also form the basis for other knowledge extraction processes from text such as question answering, information, taxonomy, ontology extraction and many more. We are looking forward to cooperation requests from scientific and other libraries.

Natural language processing tools make errors. As can be seen from the live system, the ACL Anthology Searchbench, the results are not perfect, but we are confident that further investigations into the direction we and other NLP groups have started, will turn scientific literature search into a more pleasant, efficient and successful task in the future.

Acknowledgments

This work has been funded by the German Federal Ministry of Education and Research, projects TAKE (FKZ 01IW08003) and Deependance (FKZ 01IW11003), the DFG Cluster of Excellence Multimodal Computing and Interaction (M2CI) — *Robust, Efficient and Intelligent Processing of Text, Speech, Visual Data and High Dimensional Representations* — Open Science Web, and has been conducted in the context of the world-wide DELPH-IN consortium.

References

Ananiadou, S. (2007, October). The national centre for text mining: a vision for the future. *Ariadne, 53*. Retrieved December 15, 2012, from http://www.ariadne.ac.uk/issue53/ananiadou

Berg, Ø. R., Oepen, S., & Read, J. (2012). Towards high-quality text stream extraction from PDF. Technical background to the ACL 2012 Contributed Task. In R. E. Banchs (Ed.), *Proceedings of the ACL-2012 Main conference workshop on rediscovering 50 years of discoveries* (pp. 98–103) (Jeju, Republic of Korea). Retrieved December 15, 2012, from http://www.aclweb.org/anthology-new/W/W12/W12-3211.pdf

Bézier, P. (1968). *How Renault uses numerical control for car body design and tooling.* SAE Paper 680010. New York, NY: Society of Automotive Engineers congress. doi:10.4271/680010

Bird, S., Klein, E., & Loper, E. (2009). *Natural language processing with Python: Analyzing text with the natural language toolkit*. Sebastopol: O'Reilly Media. Also available online: http://nltk.org/book/.

Callmeier, U. (2000). PET – A platform for experimentation with efficient HPSG processing techniques. *Natural Language Engineering 6*(1), 99–108.

Councill, I. G., Giles, C. L., & Kan, M. -Y. (2008). ParsCit: An open-source CRF reference string parsing package. In N. Calzolari, K. Choukri, B. Maegaard, J. Mariani, J. Odjik, S. Piperidis & D. Tapias (Eds.), *Proceedings of the 6th international conference on language resources and evaluation (LREC-2008)* (pp. 661–667) (Marrakesh, Morocco). Retrieved December 15, 2012, from http://www.lrec-conf.org/proceedings/lrec2008/pdf/166_paper.pdf

DiMarco, C., Kroon, F., & Mercer, R. (2006). Using hedges to classify citations in scientific articles. In J. G. Shanahan, Y. Qu & J. Wiebe (Eds.), *Computing attitude and affect in text theory and applications* (pp. 247–263). Springer.

Dong, C., & Schäfer, U. (2011). Ensemble-style self-training on citation classification. In H. Wang & D. Yarowsky (Eds.), *Proceedings of the 5th international joint conference on natural language processing (IJCNLP2011)* (pp. 623–631) (Chiang Mai, Thailand). Retrieved December 15, 2012, from http://aclweb.org/anthology/I11-1070.pdf

Fellbaum, C. (1998). *WordNet: An electronic lexical database*. Cambridge, MA: MIT Press.

Flickinger, D. (2002). On building a more efficient grammar by exploiting types. In D. Flickinger, S. Oepen, H. Uszkoreit & J. Tsujii (Eds.), *Collaborative language engineering. A case study in efficient grammar-based processing* (pp. 1–17). Stanford, CA: CSLI Publications.

Frantzi, K., Ananiadou, S., & Mima, H. (1998). Automatic recognition of multi-word terms: the C-value/NC-value method. In C. Nikolaou & C. Stephanidis (Eds.), *Proceedings of the 2nd European conference on research and advanced technology for digital libraries* (pp. 585–604) (Crete, Greece). Springer, Lecture Notes in Computer Science 1513.

Garfield, E. (1965). Can citation indexing be automated? In M. E. Stevens, V. E. Giuliano & L. B. Heilprin (Eds.), *Statistical association methods for mechanical documentation.* NBS Misc. Pub. 269. Washington, DC: National Bureau of Standards.

Garzone, M. (1996). *Automated classification of citations using linguistic semantic grammars.* Master's thesis, Dept. of Computer Science, The University of Western Ontario, Canada.

Hearst, M. (1992). Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th COLING conference* (pp. 539–545) (Nantes, France). Retrieved December 15, 2012, from http://www.aclweb.org/anthology-new/C/C92/C92-2082.pdf

Jurafsky, D., & Martin, J. H. (2008). *Speech and language processing: An introduction to natural language processing, speech recognition, and computational linguistics*, 2nd edition. Upper Saddle River, NJ: Prentice-Hall. (Chapter 1 available online).

Ohta, T., Matsuzaki, T., Okazaki, N., Miwa, M., Saetre, R., Pyysalo, S., & Tsujii, J. (2010). Medie and info-pubmed: 2010 update. *BMC Bioinformatics. 11 (Suppl 5)*, P7. doi:10.1186/1471-2105-11-S5-P7.

Reiplinger, M., Schäfer, U., & Wolska, M. (2012). Extracting glossary sentences from scholarly articles: A comparative evaluation of pattern bootstrapping and deep analysis. In R. E. Banchs (Ed.), *Proceedings of the ACL-2012 Main conference workshop on rediscovering 50 years of discovery* (pp. 55–65) (Jeju, Republic of Korea). Retrieved December 15, 2012, from http://www.aclweb.org/anthology-new/W/W12/W12-3206.pdf.

Schäfer, U. (2006). Middleware for creating and combining multi-dimensional NLP markup. In D. Ahn, E. T. K. Sang & G. Wilcock (Eds.), *Proceedings of the 5th workshop on NLP and XML (NLPXML-2006): Multi-dimensional markup in natural language processing, 11th Conference of the European chapter of the Association for Computational Linguistics (EACL-2008)* (pp. 81–84) (Trento, Italy). Retrieved December, 15, 2012, from http://aclweb.org/anthology/W06-2714.pdf.

Schäfer, U., & Kasterka, U. (2010). Scientific authoring support: A tool to navigate in typed citation graphs. In M. Piotrowski, C. Mahlow & R. Dale (Eds.), *Proceedings of the 11th annual conference of the North American chapter of the Association for Computational Linguistics: Human language technologies (NAACL HLT 2010) Workshop on Computational Linguistics and Writing: Writing Processes and Authoring Aids* (pp. 7–14) (Los Angeles, CA). Association for Computational Linguistics. Retrieved Deceùber 15, 2012, from http://www.aclweb.org/anthology-new/W/W10/W10-0402.pdf.

Schäfer, U., & Kiefer, B. (2011). Advances in deep parsing of scholarly paper content. In R. Bernardi, S. Chambers, B. Gottfried, F. Segond & I. Zaihrayeu (Eds.), *Advanced language technologies for digital libraries*. Springer LNCS Theoretical Computer Science Series, LNCS 6699 (pp. 135–153). Retrieved, December 15, 2012, from http://www.dfki.de/web/forschung/iwi/publikationen/renameFileForDownload?filename=Schaefer_Kiefer_ALT4DL_2011.pdf&file_id=uploads_993

Schäfer, U., Read, J., & Oepen, S. (2012). Towards an ACL anthology corpus with logical document structure. An overview of the ACL 2012 Contributed Task. In R. E. Banchs (Ed.), *Proceedings of the ACL-2012 Main conference workshop on rediscovering 50 years of discovery* (pp. 88–97) (Jeju, Republic of Korea). Association for Computational Linguistics. Retrieved December 15, 2012, from http://www.aclweb.org/anthology-new/W/W12/W12-3210.pdf.

Schäfer, U., & Weitz, B. (2012). Combining OCR outputs for logical document structure markup. Technical background to the ACL 2012 Contributed Task. In R. E. Banchs (Ed.), *Proceedings of the ACL-2012 Main conference workshop on rediscovering 50 years of discovery* (pp. 104–109) (Jeju, Republic of Korea). Association for Computational Linguistics. Retrieved December 15, 2012, from http://www.aclweb.org/anthology-new/W/W12/W12-3212.pdf.

Schäfer, U., Kiefer, B., Spurk, C., Steffen, J., & Wang, R. (2011). The ACL anthology searchbench. In S. Kurohashi (Ed.), *Proceedings of the 49th annual meeting of the Association for Computational Linguistics: Human language technologies (ACL HLT 2011), System Demonstrations* (pp. 7–13) (Portland, OR, USA). Association for Computational

Linguistics. Retrieved December 15, 2012, from http://www.aclweb.org/anthology-new/P/P11/P11-4002.pdf

Teufel, S., Siddharthan, A., & Tidhar, D. (2006). Automatic classification of citation function. In D. Jurafsky & E. Gaussier (Eds.), *Proceedings of the 2006 conference on empirical methods in natural language processing (EMNLP-2006)* (pp. 103–110) (Sydney, Australia). Retrieved December, 15, 2012 from http://www.aclweb.org/anthology-new/W/W06/W06-1613.pdf

Weitz, B., & Schäfer, U. (2012). A graphical citation browser for the ACL anthology. In N. Calzolari, K. Choukri, T. Declerck, M. U. Doğan, B. Maegaard, J. Mariani, J. Odijk & S. Piperidis (Eds.), *Proceedings of the 8th international conference on language resources and Evaluation (LREC-2012)* (pp. 1718–1722) (Istanbul, Turkey). Retrieved December 15, 2012, from http://www.lrec-conf.org/proceedings/lrec2012/pdf/805_Paper.pdf

Wolska, M., Schäfer, U., & Pham, T. N. (2011). Bootstrapping a domain-specific terminological taxonomy from scientific text. In K. Kageura & P. Zweigenbaum (Eds.), *9th International conference on terminology and artificial intelligence (TIA)* (pp. 17–23) (Paris, France). Retrieved December 15, 2012, from http://tia2011.crim.fr/Proceedings/pdf/TIA05.pdf

Notes

[1] http://www.dfki.de/~uschaefer#2012.

[2] http://aclasb.dfki.de/#yea~|2010-2012*aut~|Schäfer*.

[3] http://aclasb.dfki.de/.

[4] http://language.worldofcomputing.net/category/nlp-glossary.

[5] http://nltk.org/book/.