Adapting a Semantic Question Answering System to the Web

Sven Hartrumpf
Intelligent Information and Communication Systems (IICS)
University of Hagen (FernUniversität in Hagen)
58084 Hagen, Germany
Sven.Hartrumpf@fernuni-hagen.de

Abstract

This paper describes how a question answering (QA) system developed for small-sized document collections of several million sentences was modified in order to work with a monolingual subset of the web. The basic QA system relies on complete sentence parsing, inferences, and semantic representation matching. The extensions and modifications needed for useful and quick answers from web documents are discussed. The main extension is a two-level approach that first accesses a web search engine and downloads some of its document hits and then works similar to the basic QA system. Most modifications are restrictions like a maximal number of documents and a maximal length of investigated document parts; they ensure acceptable answer times. The resulting web QA system is evaluated on the German test collection from QA@CLEF 2004. Several parameter settings and strategies for accessing the web search engine are investigated. The main results are: precision-oriented extensions and experimentally derived parameter settings are needed to achieve similar performance on the web as on small-sized document collections that show higher homogeneity and quality of the contained texts; adapting a semantic QA system to the web is feasible, but answering a question is still expensive in terms of bandwidth and CPU time.

1 Introduction

There are question answering (QA) systems intended for small-sized document collections (*textual QA systems*) and QA systems aiming at the web (*web(-based) QA systems*). In this paper, a system of the former type (InSicht, see (Hartrumpf, 2005b)) is transformed into one of the latter type (called InSicht-W3 for short). In Sect. 2, the textual QA system for German is presented. The extensions and modifications required to get it working with the web as a virtual document collection are described and discussed in Sect. 3. The resulting system is evaluated on a well-known test collection and compared to the basic QA system (Sect. 4). After the conclusion, some directions for further research are indicated.

2 The Basic QA System

The semantic[1] QA system that was turned into a web QA system is InSicht. It relies on complete sentence parsing, inferences, and semantic representation matching and comprises six main steps.

In the *document processing* step, all documents from a given collection are transformed into a standard XML format (CES, corpus encoding standard, see http://www.cs.vassar.edu/CES/) with word, sentence, and paragraph borders marked up by XML elements *w*, *s*, and *p*, respectively. Then, all preprocessed documents are parsed by WOCADI (Hartrumpf, 2003) yielding a syntactic dependency structure and, more importantly, a semantic representation, a semantic network of the MultiNet formalism (Helbig, 2006) for each document sentence. The parser can produce intersentential coreference links for a document.

In the second step (*query processing*), the user's question is parsed by WOCADI. Determining the sentence type (here, often a subtype of *question*) is especially important because it controls some parts of two later steps: query expansion and answer generation.

Next comes *query expansion*: Equivalent and similar semantic networks are derived from the original query network by means of lexico-

[1] *Semantic* in the sense that formal semantic representations of documents and questions are automatically produced by a parser and form the system center.

semantic relations from HaGenLex (Hagen German Lexicon, (Hartrumpf et al., 2003)) and a lexical database (GermaNet), equivalence rules, and inferential rules like entailments for situations (applied in backward chaining). The result is a set of disjunctively connected semantic networks that try to cover many possible kinds of representations of sentences possibly containing an explicit or implicit answer to the user's question.

In the fourth step (*semantic network matching*), all document sentences matching at least one of the semantic networks from query expansion are collected. A two-level approach is chosen for efficiency reasons. First, an index of concepts (disambiguated words with IDs from the lexicon) is consulted with the relevant concepts from the query networks. Second, the retrieved documents are compared sentence network by sentence network to find a match with a query network.

Answer generation is next: Natural language (NL) generation rules are applied to semantic networks that match a query network in order to generate an NL answer string from the deep semantic representations. The sentence type and the semantic network control the selection of generation rules. The rules also act as a filter for uninformative or bad answers. The results are tuples of generated answer string, numerical score, supporting document ID, and supporting sentence ID.

To deal with different answer candidates, an *answer selection* step is required. It realizes a quite simple but successful strategy that combines a preference for more frequent answers and a preference for more elaborate answers. The best answers (by default only the single best answer) and the supporting sentences are presented to the user that posed the question.

The first step, document processing, is run offline in InSicht; it is run online in InSicht-W3 to avoid unacceptable parsing costs before the system can be used. The remaining five steps will be left mostly unchanged for InSicht-W3, in parts just differently parameterized.

3 QA System Extensions for the Web

3.1 A Naive Approach to Web-based QA

The simplest approach to turning InSicht into a web QA system would be to collect German web pages and work with the resulting document collection as described in Sect. 2. However, a deep semantic analyzer will need several years to parse all web pages in German (even if excluding the pages from the much larger *deep web*). The following formula provides a rough estimate[2] of the CPU years needed:

$$\begin{aligned} t &= \frac{\#\text{documents} \cdot \#\text{sent\_per\_document}\,[sent]}{\text{parser\_speed}\,[sent/h]} \\ &= \frac{500,000,000 \cdot 320\,[sent]}{4000\,[sent/h]} \\ &= 40,000,000\,h \quad \approx \quad 4,566\,a \end{aligned}$$

This long time indicates that the naive approach is currently not an option. Therefore a multi-level approach was investigated. In this paper, only a two-level approach is discussed, which has been tried in shallow QA systems.

3.2 A Two-Level Approach to Web-based QA

As in other applications, a web search engine can be used (as a first level) to preselect possibly relevant documents for the task at hand (here, answering a question posed by a user). In InSicht-W3, the web is accessed when similar and equivalent semantic networks have been generated during query expansion. Clearly, one cannot directly use this semantic representation for retrieving document URLs from any service out there on the web—at least till the arrival of a web annotated with formal NL semantics. One must transform the semantic networks to an adequate level; in many web search engines, this is the level of search terms connected in a Boolean formula using *and* and *or*.

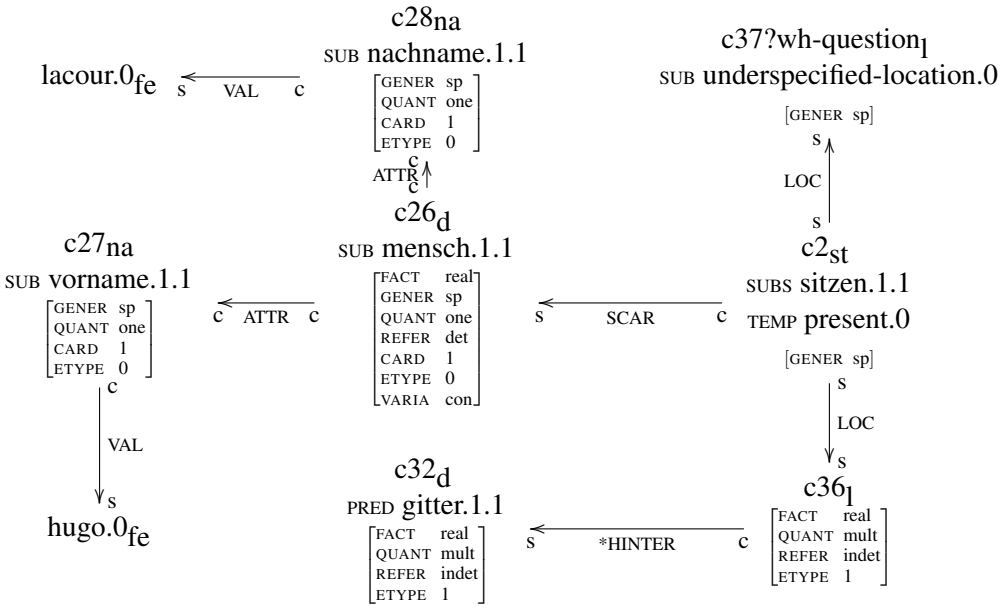
To derive a search engine query from a semantic network, all *lexical concepts* from the semantic network are collected. For example, question qa04_068 from QA@CLEF 2004 in example (1) leads to the semantic network depicted in Fig. 1 (and related semantic networks; for a QA@CLEF 2004 question, 4.8 additional semantic networks on average).

(1) Wo sitzt Hugo Lacour hinter Gittern?
Where sits Hugo Lacour behind bars?
'Where is Hugo Lacour imprisoned?'

The lexical concepts are roughly speaking those concepts (represented as nodes in graphical form) whose names are not of the form *c1*, *c2*, etc. The lexical concepts are named after the lemma of the

[2] The average number of sentences per document has been determined from a sample of 23,800 preprocessed web documents in InSicht-W3.

Figure 1: Semantic network for CLEF question qa04_068: *Wo sitzt Hugo Lacour hinter Gittern?* (*'Where is Hugo Lacour imprisoned?'*). Some edges are shown folded below the node name.



corresponding word plus a numerical homograph identifier and a numerical reading identifier.

As current web search engines provide no (or only rough) lemmatization for German, one must go one step further away from the semantic network by generating full forms for each word belonging to a lexical concept. (Others have tried a similar approach; for example, Neumann and Sacaleanu (2005) construct IR queries which can be easily adapted to different IR engines.) In the example, the node *sitzen.1.1* (*to sit*) leads to 26 different full forms (shown below in example (2)). These forms can be used as search terms; as alternatives, they are connected disjunctively. Thus for each semantic network from query expansion, a conjunction of disjunctions of word forms is constructed. Word forms can belong to words not occurring in the question because query expansion can introduce new concepts (and thereby new words), e.g. by inferential rules. The Boolean formulae for several semantic networks are connected disjunctively and the resulting formula is simplified by applying logical equivalences. What one can pass to a web search engine as a *preselection query* for question (1) is shown in prefix notation in example (2):

(2) *(or*
(and
(or Gitter Gittern Gitters)
(or *Hugo Hugos*)
(or *Lacour Lacours*)
(or gesessen gesessene gesessenem gesessenen gesessener gesessenes saß saßen saßest saßet saßt sitze sitzen sitzend sitzende sitzendem sitzenden sitzender sitzendes sitzest sitzet sitzt säße säßen säßest säßet))
. . . ; query derived from second query network
)

If a search engine does not offer Boolean operators (or limits Boolean queries to some hundred characters), one can transform the Boolean query into DNF and issue for each disjunct a search engine query that requires all its terms (for example, by prepending a plus sign to every term).

There are several problems related to breaking down a semantic network representation into a Boolean formula over word forms. For example, lexical concepts that do not stem from a surface form in the question should be excluded from the query. An example is the concept *nachname.1.1* (*'last name'*) which the named entity subparser of WOCADI installs in the semantic network when constructing the semantic representation of a human being whose last name is specified in the text (see Fig. 1).

The generation of a preselection query described above indicates that there is a fundamental *level mismatch* between the first level (web search engine) and the second level (an NL understanding system like the QA system InSicht): words

(plus proximity information[3]) vs. concepts within (possibly normalized) semantic representations of sentences (or texts). This level mismatch deteriorates the precision of the first level and thereby the recall of the second level. The latter might be surprising. The precision of the first level (viewed in isolation) is not problematic because the second level achieves the same precision no matter how low the precision of the first level is. But precision *does* matter because the limited number of documents that the first level can return and that the second level can efficiently process might lead to not seeing relevant documents (i.e. lower recall) or at least to seeing them much later (which can imply unacceptable answer times). Or to put it differently: Very low precision in the first level can lead to lower recall of the whole multi-level system if there are any retrieval set limits for the first level.

At the end of the first level, the retrieved web documents are simplified by a script that uses the text dump function of the *lynx* web browser to extract the textual content. The resulting document representations can be processed by InSicht.

3.3 Deficiencies of Web Search Engines

Web search engines typically restrict the length of queries, e.g. to 1000 bytes, 1500 bytes, or 10 terms. But the rich morphology of languages like German can lead to long and complex preselection queries so that one needs to reduce preselection queries as far as possible (e.g. by dropping rare word forms). An additional strategy is to split the query into several independent subqueries.

The QA system is currently based on matching semantic representations with the granularity of a sentence only. Thus one would achieve much better precision if the web search engine allowed a *same-sentence* operator or at least a proximity operator (often named *NEAR*) of say 50 words which ignores linguistic units like sentences and paragraphs but suffices for our goals. Although some web search engines once had some kind of proximity operator, currently none of the search engines with large indices and Boolean queries seems to offer this facility.

3.4 Pragmatic Limits for Web-based QA

Pragmatic limits are needed to ensure acceptable answer times when dealing with the vast number of web documents. The following constraining parameters are implemented in InSicht-W3:

number of documents d (BT-R) [4] The number of documents retrieved from a search engine is limited by InSicht-W3. A separate query is sent to the search engine for each top-level disjunct, which corresponds to one semantic network. For each of the q subqueries, d/q documents are retrieved, at most.

Many search engines employ a maximal number of hits h. (A popular choice is currently 1000.) Due to other filtering constraints (like snippet testing as defined below), the limit h can inhibit the performance of InSicht-W3 even if h is greater than d.

document format (BT-R) Only documents encoded as HTML or plain text are covered. Other formats are omitted by adding a format constraint to the preselection query, by investigating the URL, and by interpreting the result of the UNIX program *file*.

document language (BPT-R) Only documents that are mainly written in German are treated. A language identification module is available in InSicht-W3, but the selection of document language in the web search engine was precise enough.

document length l (T-R) If a document is longer than l bytes it is shortened to the first l bytes (current value: 1,000,000 bytes).

document URL (BPT-R) The URL of a document must belong to certain top-level domains (.at, .ch, .de, .info, .net, .org) and must not be on a blacklist. Both conditions aim at higher quality pages. The blacklist is short and should be replaced by one that is collaboratively maintained on the web.

snippet testing (BPT-R) Many search engines return a *text snippet* for each search hit that typically contains all word forms from the search engine query. Sometimes the snippet does not fulfill the preselection query. A stronger constraint is that a snippet passage

[3] At least some years ago and hopefully in the future again, see Sect. 3.3.

[4] Trade-offs (and goals) are indicated in parentheses, between bandwidth (B), precision (P), (answer) time (T), on the one hand, and recall (R), on the other hand.

without ellipsis (. . .) must fulfill the preselection query. This realizes an imperfect *same-sentence* or similar proximity operator.

word form selection (BT-R) To shorten preselection queries for highly inflecting word categories (e.g. German verbs), InSicht-W3 omits full forms that are less likely to be contained in answering document sentences (e.g. verb forms that can only be imperative or first and second person). For adjectives, only the forms matching the degree (positive, comparative, and superlative in German) used in the question are included. The most restrictive parameter setting is to choose only one word form per content word, e.g. the word form occurring in the question.

These constraints can be checked before the second level starts.

The following parameters are applicable on the second level:

number of parsed sentences *s* **(T-R)** For each document, at most *s* sentences are parsed (current value: 100).

sentence language (PT-R) As many web pages contain passages in several languages, it is sometimes important to discard sentences that seem to be in a different language than the document itself. If the percentage of unknown words in a sentences reaches a threshold, the sentence is considered to be written in a foreign language

sentence selection (T) Only a sentence whose set of word forms fulfills the preselection query sent to the web search engine is parsed by WOCADI. This realizes a *same-sentence* operator on the second level. Of course, this operator would be more helpful in the web search engine (see Sect. 3.3).

3.5 Parameter Settings of the QA System

In general, the QA system must be parameterized differently when going from a high quality, low-redundancy, small-sized collection (like archives of some newspapers) to a mixed quality, high-redundancy, large-sized collection (like a considerable subset of the web). For example, the variation of concepts (like synonyms, hyponyms, and hyperonyms) is less important on the web than in smaller collections due to its large redundancy and increased chance to find a formulation whose semantic representation contains exactly the concepts from the question. This is helpful because full concept variations lead for some questions to very long preselection queries.

3.6 Language-specific Problems

InSicht-W3 has to deal with several problems in the German language and German web pages, which might be less problematic for English or some other languages. On the one hand, the rich morphology requires to generate word forms for content words derived from query networks; on the other hand, to avoid very long preselection queries the generation must be limited (e.g. with the word form selection parameter).

The morphological phenomenon of separable prefix verbs in German requires special treatment. Certain word forms of such verbs must appear as one orthographic word or two orthographic words (often separated by many intervening words). The choice depends on the word order of the sentence or clause that contains the verb. InSicht-W3's preselection queries contain both variants, e.g. for the word form *absitzt* the following formula is generated: *(or absitzt (and ab sitzt))*.

In German web pages, two orthography systems can be found, the old one and the new one introduced in 1998. As both systems are often mixed in pages (and question sets), the NL parsing must be flexible and exploit the links between old spellings, new spellings, and spelling variants. The parser normalizes all words and concept names to the old orthography. For preselection queries, word forms for all orthography systems must be generated.

Not only the orthography system might change in a web page, also the language itself might switch (often without correct markup), e.g. between English and German. Therefore the sentence language parameter from Sect. 3.4 was introduced. Such effects also occur for other languages.

German texts contain important characters that are not covered by the ASCII character set. (The most frequent ones are *ä*, *ö*, *ü*, *Ä*, *Ö*, *Ü*, and *ß*.) Unfortunately, different character encodings are used for German web pages, sometimes with a mismatch of declared (or implied) encoding and actually used encoding. This problem has been successfully delegated to the *lynx* web browser (see end of Sect. 3.2).

Table 1: Evaluation results for the German questions from QA@CLEF 2004. InSicht used all 200 questions, InSicht-W3 only 170 questions (see text).

setup		answers (%)					K1
system	document collection	non-empty			empty		
		right	inexact	wrong	right	wrong	
InSicht	QA@CLEF document collection	30.5	2.5	0.0	10.0	57.0	0.28
InSicht-W3	web (as virtual document collection)	22.9	2.9	2.4	0.0	71.8	0.18

4 Evaluation of the Web QA System

The resulting web QA system is evaluated on an established set of test questions: QA@CLEF 2004 (Magnini et al., 2005). Some questions in this set have an important implicit temporal context. For example, the correct answer to questions like qa04_090 in example (3) critically depends on the time this present tense question is uttered.

(3) Wer ist Präsident von UNICE?
Who is president of UNICE?

In QA@CLEF, a supported correct answer to this question can be the name of a UNICE president only from a certain time period because the QA@CLEF document collection consists of newspaper and newswire articles from 1994 and 1995 (and because there are no documents about the history of UNICE). On the web, there are additional correct answers.[5] Questions with implicit time restriction (30 cases) are excluded from the evaluation so that 170 questions remain for evaluation in InSicht-W3. Alternatives would be to refine these questions by making the temporal restriction explicit or to extend the gold standard by answers that are to be considered correct if working on the web.

Table 1 contains evaluation results for InSicht-W3: the percentages of right, inexact, and wrong answers (separately for non-empty answers and empty answers) and the K1-measure (see (Herrera et al., 2005) for a definition). For comparison, the results of the textual QA system InSicht on the QA@CLEF document collection are shown in the first row. The percentages for non-empty answers differ for right answers and wrong answers. In both aspects, InSicht-W3 is worse than InSicht. The main reason for these changes is that the structure and textual content of documents on the web are much more diverse than in the QA@CLEF collection. For example, InSicht-W3 regarded some sequences of words from unrelated link anchor texts as a sentence, which led to some wrong answers especially for definition questions. Also for empty answers, InSicht-W3 performs worse than InSicht. This is in part due to the too optimistic assumption during answer assessment that all German questions from QA@CLEF 2004 have a correct answer on the web (therefore the column labelled *right empty answers* contains 0.0 for InSicht-W3). However, InSicht-W3 found 12 right answers that InSicht missed. So there is a potential for a fruitful system combination.

The impact of some interesting parameters was evaluated by varying parameter settings (Table 2). The query network quality q_{min} (column 2) is a value between 0 (worst) and 1 (best) that measures how far away a derived query network is from the original semantic network of the question. For example, omitting information from the semantic network for the question (like the first name of a person if also the last name is specified), leads to a reduction of the initial quality value of 1. The value in column 2 indicates at what threshold variant query networks are ignored. Column 3 corresponds to the parameter of word form selection (see Sect. 3.4). The two runs with $d = 300$ and $q_{min} = 0.8$ show that morphological generation pays off, but the effect is smaller than in other document collections. This is probably due to the fact that the large and redundant web often contains answers with the exact word forms from the question. Another interesting aspect found in Table 2 is that even with $d = 500$ results still improve; it remains to be seen at what number of documents performance stays stable (or maybe degrades). The impact of a lower quality threshold q_{min} was not significant. This might change if one operates with a larger parameter d and more inferential rules.

[5] Or just one correct answer, which differs from the one in QA@CLEF, when one interprets the present tense in the question as referring to today.

Table 2: Influence of parameters on InSicht-W3 results. Parameter d is the maximal number of documents used from the search engine results (see Sect. 3.4); q_{min} is the minimal query network quality.

parameter setting			results						
d	q_{min}	morph. generat.	#docs. from first level per quest.	#sent. matching pre. query per quest.	%docs. with matching sent.	right non-empty answ. (%)	inexact answ. (%)	wrong non-empty answ. (%)	K1
100	0.9	frequent	18.1	30.5	59.7	18.8	4.1	1.8	0.13
200	0.9	frequent	34.4	59.0	60.4	20.6	3.5	2.9	0.14
300	0.9	frequent	45.5	76.9	60.2	22.4	3.5	2.4	0.16
300	0.8	frequent	48.4	84.2	61.7	22.4	3.5	2.4	0.16
300	0.8	none	57.3	91.3	61.5	19.4	3.5	3.5	0.12
500	0.8	frequent	68.9	119.8	62.2	22.9	2.9	2.4	0.18

Error analysis and classification is difficult as soon as one steps to the vast web. One could start with a classification of error reasons for wrong empty answers. The error class that can be most easily determined is that the search engine returned no results for the preselection query. 40 of the 170 questions (23.5%) belong to this class. This might surprise users that believe that they can find nearly every bit of information on the web. But there are areas where this assumption is wrong: many QA@CLEF questions relate to very specific events of the year 1994 or 1995. Twelve years ago, the web publishing rate was much lower than today; and even if the relevant pieces of information were on the web at that time (or in the following years), they might have been moved to archives or removed in recent years. Other error reasons are very difficult and labor-intensive to separate:

1. Result pages from the first level do not contain an answer, but if one requests more documents (e.g. by raising the parameter d) result pages with an answer will be found.

2. Result pages from the first level contain an answer, but one or more components of the textual QA system cause that the answer is not found. Subclasses could be defined by adapting the hierarchy that Hartrumpf (2005a) applied to evaluate InSicht.

On average, the 40th search engine result is the first that contains a right answer (in the best InSicht-W3 run).[6] Although this result is for the system and not for a real user (they differ in their strengths and weaknesses in NL understanding), it indicates that human users of web search engines might find the correct answer quite late if at all. This is a strong argument for more advanced, second level tools like a semantic QA system: How many users will read through 40 search engine results and (possibly) 40 underlying web pages?

The answer time of InSicht-W3 currently ranges between 15 and 1200 seconds. The document download time was excluded because it depended on too many external factors (caches, intranet effects, parallel vs. sequential downloads) to be measured consistently and reliably.

5 Related Work

There are few QA systems for German. The system described by Neumann and Xu (2003) works on German web pages. Its general approach differs from InSicht because it relies on shallow, but robust methods, while InSicht builds on sentence parsing and semantic representations. In this respect, InSicht resembles the (English) textual QA system presented by Harabagiu et al. (2001). In contrast to InSicht, this system applies a theorem prover and a large knowledge base to validate candidate answers. An interesting combination of web and textual QA is presented by Vicedo et al. (2005): English and Spanish web documents are used to enhance textual QA in Spanish.

[6] This high number is in part an artifact of the preselection query generation. In a more thorough analysis, human users could be given the NL question and be asked to formulate a corresponding search engine query.

One of the first web QA systems for English was Mulder (Kwok et al., 2001). Mulder parses only the text snippets returned by the search engine, while InSicht-W3 parses the underlying web pages because the text snippets often have omissions ('...') so that full parsing becomes problematic or impossible. InSicht-W3's approach needs more time, especially if the web pages are not in the local cache that InSicht-W3 maintains in order to reduce its bandwidth requirements.

6 Conclusion and Perspectives

In this paper, an existent textual QA system was extended and modified to work successfully on the German web as a virtual document collection. The main results are: precision-oriented extensions and experimentally derived parameter settings are needed to achieve similar performance on the vast web as on small-sized document collections that show higher homogeneity and quality of the contained texts; taking a semantic QA system to the web is feasible as demonstrated in this paper, but answering a question is still expensive in terms of bandwidth and CPU time.

There are several interesting directions for future work. The first level of InSicht-W3 can be improved by finding a better suited search engine or by building and running a new one in a distributed manner. Ideally, it should support arbitrarily complex Boolean queries, parameterized proximity operators like NEAR/N ($N \in \{1,2,\ldots,100\}$), or even linguistically informed operators like *same-sentence* and *same-paragraph*.

The second level (the textual QA system) can be improved by acquiring more inferential knowledge to allow better query expansion. The matching approach can be extended from the unit *sentence* to a larger linguistic unit like paragraph, text, and even text collection. Distributed architectures and algorithms can reduce answer times.

References

Sanda Harabagiu, Dan Moldovan, Marius Paşca, Rada Mihalcea, Mihai Surdeanu, Răzvan Bunescu, Roxana Gîrju, Vasile Rus, and Paul Morărescu. 2001. The role of lexico-semantic feedback in open-domain textual question-answering. In *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics (ACL-2001)*, pages 274–281, Toulouse, France.

Sven Hartrumpf, Hermann Helbig, and Rainer Osswald. 2003. The semantically based computer lexicon HaGenLex – Structure and technological environment. *Traitement automatique des langues*, 44(2):81–105.

Sven Hartrumpf. 2003. *Hybrid Disambiguation in Natural Language Analysis*. Der Andere Verlag, Osnabrück, Germany.

Sven Hartrumpf. 2005a. Question answering using sentence parsing and semantic network matching. In (Peters et al., 2005), pages 512–521.

Sven Hartrumpf. 2005b. University of Hagen at QA@CLEF 2005: Extending knowledge and deepening linguistic processing for question answering. In Carol Peters, editor, *Results of the CLEF 2005 Cross-Language System Evaluation Campaign, Working Notes for the CLEF 2005 Workshop*. Centromedia, Wien, Austria.

Hermann Helbig. 2006. *Knowledge Representation and the Semantics of Natural Language*. Springer, Berlin.

Jesús Herrera, Anselmo Peñas, and Felisa Verdejo. 2005. Question answering pilot task at CLEF 2004. In (Peters et al., 2005), pages 581–590.

Cody Kwok, Oren Etzioni, and Daniel S. Weld. 2001. Scaling question answering to the web. *ACM Transactions on Information Systems*, 19(3):242–262.

Bernardo Magnini, Alessandro Vallin, Christelle Ayache, Gregor Erbach, Anselmo Peñas, Maarten de Rijke, Paulo Rocha, Kiril Simov, and Richard Sutcliffe. 2005. Overview of the CLEF 2004 multilingual question answering track. In (Peters et al., 2005), pages 371–391.

Günter Neumann and Bogdan Sacaleanu. 2005. Experiments on robust NL question interpretation and multi-layered document annotation for a cross-language question/answering system. In (Peters et al., 2005), pages 411–422.

Günter Neumann and Feiyu Xu. 2003. Mining answers in German web pages. In *Proceedings of the International Conference on Web Intelligence (WI-2003)*, Halifax, Canada.

Carol Peters, Paul Clough, Julio Gonzalo, Gareth J. F. Jones, Michael Kluck, and Bernardo Magnini, editors. 2005. *Multilingual Information Access for Text, Speech and Images: 5th Workshop of the Cross-Language Evaluation Forum, CLEF 2004*, volume 3491 of *LNCS*. Springer, Berlin.

José L. Vicedo, Maximiliano Saiz, Rubén Izquierdo, and Fernando Llopis. 2005. Does English help question answering in Spanish? In (Peters et al., 2005), pages 552–556.