Open Competitie 2005
Exacte Wetenschappen

1 Project Details

1.a) Project Title. Question Answering as Semistructured Information Retrieval

1.b) Project Acronym. QASSIR

1.c) Principal Investigator. Prof.dr. Maarten de Rijke
Information & Language Processing Systems (ILPS)
Informatics Institute
University of Amsterdam
E-mail: mdr@science.uva.nl
URL: http://ilps.science.uva.nl/

2 Project Content

2.a) Summary. People have access to unprecedented amounts of information over the Internet. Making sense of the information, usually in free-text, requires advanced methods for focused information retrieval. This proposal brings together two ways of focused information retrieval: *question answering* (QA) and *semistructured information retrieval*. Specifically, we aim to address challenges raised by today's QA systems by recasting the QA task as a semistructured information retrieval task.

XML is the de facto standard for capturing metadata and semantically rich information. XML retrieval holds the promise of providing more focused, and semantically more informed, information access than traditional document retrieval. How can this help QA? By performing the tagging and extraction work required for QA off-line, at indexing time, we create text (and data) collections with multiple annotation layers, against which QA is to be performed as an "answer retrieval" task. Documents marked up with multiple annotations may have tag spans that overlap without being nested: they are not legal XML documents, but *semistructured* documents; retrieving information from such documents is called *semistructured information retrieval*. We devise collection annotation schemes, query languages for semistructured retrieval, data-driven mappings from questions to queries, and build on recent advances in semistructured retrieval so as to identify contexts that can serve as candidate answers. By tackling QA as a semistructured information retrieval task, we can incorporate multiple data sources and annotation layers in the QA process, all tightly integrated within a single framework. This provides a theoretically transparent model for QA, and addresses a number of practical challenges faced by traditional QA systems.

2.b) Abstract for laymen (in Dutch). *Information Retrieval* (IR) is het vakgebied dat zich bezig houdt met de ontsluiting van grote documentencollecties. Voor het grote publiek is dit vrijwel synoniem met het concept van de Internetzoekmachines zoals Google. Echter, systemen als Google geven als antwoord op een zoekvraag een heel document, en laten het aan de gebruiker over om hier de gewenste informatie uit te extraheren. Huidig onderzoek in de IR spitst zich met name toe op methoden die verder gaan dan het vinden van hele documenten. Twee belangrijke voorbeelden zijn *XML Retrieval*, waar het systeem een willekeurig onderdeel van een document als resultaat kan geven, en vraag-antwoord systemen, waar het systeem het preciese antwoord op een zoekvraag geeft. Het onderhavige voorstel benadert de vraag-antwoord taak als een zoektaak in gestructureerde documenten. Door vooraf, "off-line", de documenten te analyseren and te structureren, ontstaat een corpus dat geannoteerd is met semantische en linguïstische metadata. Hierop kunnen methoden uit de XML retrieval direct worden toegepast, om dat deel van de tekst te vinden dat precies alle relevante informatie bevat om een gegeven vraag te beantwoorden.

3 Classification

The project falls within the discipline of Computer Science. The proposed research fits within the following NWO classes (*rapport Verkenningscommissie 1996*):

2. Data- en kennissystemen

2.2 *Datamining en -warehousing: zoeken naar verborgen verbanden in heterogene databases.*

2.3 *Information retrieval: o.a. hypertext, hypermedia en digitale libraries.*

2.8 *Natuurlijke taalverwerking: o.a. text processing en automatisch vertalen.*

4. Interactie

4.5 *User interface systemen: o.a. generieke applicaties en componenten voor integratie van verschillende media.*

Relevant *Nationale Onderzoeksagenda Informatica 2001-2005 (NOAG-i)* research themes:

4. Multimedia (MM): knowledge disclosure; organization of non-structured data; editors on the semantic level;

6. Intelligent Systems (IS): Search and retrieval techniques; knowledge distillery.

4 Composition of the Research Team

Name	Title	Role	Expertise	Affiliation
de Rijke, M.	Prof.dr.	Applicant	information processing and Internet	U. Amsterdam
Kamps, J.	Dr.	Co-applicant	information storage and retrieval	U. Amsterdam
Ahn, D.	Dr.	Postdoc	question answering	U. Amsterdam
Azzopardi, L.	Dr.	Postdoc	information storage and retrieval	U. Amsterdam
Jijkoun, V.	Drs.	Ph.D. Student	applied language processing	U. Amsterdam
Sigurbjörnsson, B.	Drs.	Ph.D. Student	XML retrieval	U. Amsterdam
Callan, J.	Prof.dr.	Advisor	retrieval, appl. language processing	CMU
Litkowski, K.	Dr.	Advisor	question answering, XML	CL Research
Monz, C.	Dr.	Advisor	retrieval for question answering	U. Maryland
Ogilvie, P.	Mr.	Advisor	retrieval for question answering	CMU
NN		OIO	semistructured retrieval	To be funded by NWO
NN		Sci. programmer	testing and system development	To be funded by NWO

The project will be supervised by both applicants, Maarten de Rijke and Jaap Kamps. The envisaged promotor will be Prof. de Rijke. Members of the existing QA team at UvA are available for consultation on our existing question answering and retrieval software infrastructure (Ahn, Jijkoun). Our leading expert on XML retrieval, Börkur Sigurbjörnssson, is available for advice on XML retrieval methods and query languages. Azzopardi brings in expertise on language modeling and semistructured retrieval. The local research team is backed up by international experts. Callan, Litkowski, and Ogilvie all contribute expertise on generalizations of XML-oriented text retrieval algorithms to support language-related tasks, while Monz contributes expertise on retrieval for QA and QA system optimization.

5 Research School

This research will be embedded in SIKS, the Dutch Research School for Information and Knowledge Systems.

6 Description of the Proposed Research

6.1 Scientific Problem and Desired Results

Background. People have access to more information than ever before. Much of it is available in free text, and there is a huge need for tools to help organize, classify, and store the information, and to allow better access to the stored information. Research in information retrieval (IR) has made significant progress in addressing this problem. Large parts of this work have found their way into our everyday world. In addition, significant progress has been made in our understanding of document retrieval methods.

Today's IR systems allow us to locate documents that might contain the pertinent information, but most leave it to the user to extract the useful information from a ranked list. This leaves the user with a large amount of text to consume. Library experience, Web engines like AskJeeves, help desks, and IR engine query logs all show that there is a real need for tools that reduce the amount of text one might have to read to obtain the desired information. To address this need, the IR community is exploring ways of *pinpointing* highly relevant information. This proposal brings together two ways of pinpointing information: *question answering* (QA) and *semistructured information retrieval*. Semistructured information is data (textual data, in our case) supplied with non-rigid, "loose" structure, unlike, for instance, relational databases with their strict schemes. XML has emerged as the de facto standard for capturing semistructured information; XML retrieval is a departure from standard document retrieval in which each individual XML element, ranging from words, phrases, clauses or sentences to full-blown articles, is a retrievable unit [23].

QA systems return short passages or direct answers to questions rather than (URLs pointing to) whole documents. E.g., in reply to *Which painter is famous for painting sunflowers?*, a QA system would answer *Vincent van Gogh*, ideally, providing some context (e.g., sentence or short snippet) justifying the choice of the answer. The TREC QA track, running since 1999, has been and continues to be a very significant initiative in evaluating QA systems [91]. The annual TREC cycles, beginning with a modest version of the QA task, have progressed through task variations representing demanding test conditions, with large text collections, different user interaction scenarios and carefully designed performance evaluation.

Research Problem. Despite the rapid progress, we see a number of challenges with today's QA work. Most importantly, modern QA systems are substantial team efforts, involving the design and maintenance of a large number components (question analysis, passage retrieval, answer extraction, answer ranking), often organized in a long pipeline, sometimes with intricate fallback mechanisms [65] and branching [1]. In short, QA systems are far from being the shrink-wrapped solutions that today's retrieval systems have become. Many hidden dependencies between the components make fine-tuning of modules and optimization of the whole system very difficult [27, 66, 76]. And the typical pipeline organization of QA systems makes it hard to define theoretically transparent models for QA [22, 82]. The problem is compounded by the fact that moves to ever richer types of question require additional tagging and analysis steps, leading to longer pipelines, and, moreover, to serious performance issues in case most of the components are called at question time (as is the case with the majority of today's QA systems) [33, 82].

How can we move closer to QA as an effective and trainable "shrink wrapped" method for textual information access? Our proposed solution starts from the following observation. For performance reasons, several QA systems [2, 4, 19, 37, 39, 54, 75] move some or all of their expensive processing steps off-line, to be carried out at indexing time, thus trading increased storage requirements for performance improvements at question time. In such a setting, the QA task changes to become an intelligent retrieval task against previously tagged, extracted, and mined (textual) data, often with multiple annotation layers—essentially, semistructured data. Our proposed solution is to push this idea as far as possible, by re-interpreting QA as retrieval against semistructured data. This, then, is the main *research problem* of the QASSIR proposal:

> *How can we model the question answering task as a semistructured information retrieval task, employing a transparent architecture of tightly integrated processing steps, without customized "glue", able to address increasingly hard types of question in an effective manner?*

Recent advances in semistructured information retrieval and models for storing and querying multiple (possibly, non-nested and conflicting) annotation hierarchies, make our view on QA as semistructured retrieval very attractive and timely.

Research Aims. Our main research problem gives rise to four specific key research aims: (a) automatic annotation of data collections, (b) defining query languages (c) mapping natural language questions to semistructured queries, and (d) evaluating semistructured queries. Below, we describe these aims in more detail, and in Subsection 6.2 we detail our strategies for approaching them.

(a) Document Processing. Here the aim is to move as much as possible of the expensive tagging and information extraction required for QA off-line. That is, we need to create a semistructured document collection, with multiple annotation layers, eventually giving rise to multiple indexes against which queries derived from questions will be evaluated. Markup should concern both structural information (document structure: articles, paragraphs, sentences, tables, captions, etc.) and linguistic information (including part-of-speech tags, named entities, syntactic phrases, grammatical and functional relations, co-reference resolution, semantic roles, that we have found to be helpful in our ongoing QA work [2, 10]). We should be able to realize the annotations using existing standard tools (such as statistical taggers and parsers) as well as language processing tools that we have developed in-house [7]. The challenge here is that we *have* to work with multiple text collections, marked-up in multiple ways, together with collections of structured data, tables, as well as metadata. While existing tools will be used to generate annotations, a challenge remains to design annotation schemes and indexing techniques that will meet the requirements at the retrieval step, especially with regards to overlapping but not necessarily nested markup.

(b) Query Language. Here, the aim is to define query languages for flexible and effective access to the extracted information. The task of querying the kind of semistructured documents relevant to QASSIR is a natural meeting point of two disciplines: the structural constraints (either on the element to be returned or on its context) call for methods from the database field, and the textual nature of the documents calls for approaches from IR (cf. [90, Section 5]). For QASSIR, any language for querying semistructured data has to address both types of constraint and at the same time allow efficient processing at the evaluation stage. Additionally, the query language needs to be able to cope with multiple annotations of the same data—a step outside the common tree-like document structure; and for different annotation schemes over the same document collection, it should be possible to express relationships between elements in these schemes, such as order and proximity.

(c) Question-to-Queries Mapping. Given a suitable query language, we need to devise automated, data-driven mappings from questions to semistructured queries. That is, with every question, we need to associate *content and structure* queries, referring to different types of annotation and corresponding to different answer extraction strategies. We will need to associate *sets* of queries with a given question, not individual queries: each query in the set corresponds to a different way of identifying candidate answers. From our multi-stream approach to QA we know that it is important to attempt as many different ways of identifying answers as possible to try and bridge the so-called 'vocabulary gap,' the phenomenon that questions and their answers may be phrased in different vocabularies [2, 37, 38].[1]

(d) Query Evaluation. Finally, we need to design and build a semistructured information retrieval system that will identify answers in semistructured documents by evaluating queries generated from questions. How should the queries be evaluated? How should the resulting ranked lists be integrated into a single result list? One of the big questions here is: How strictly should the content requests embedded in the *content and structure* queries be interpreted? And, how exact should the structure matching be? Evidence from XML retrieval suggests that appropriate structural constraints can significantly improve early precision scores [5, 78, 79]. However, relations between hierarchies may not be exact due to errors in language processing, so approximate matching on structure and content is important and should be reflected in the result ranking [3]. Related to this, is the following: for *content and structure* queries against a single annotation we know how to bring the various pieces of evidence together, and use these to rank the retrieved elements [5, 46, 79], but combining evidence from multiple annotation layers remains a challenge.

6.2 Research Method

At the heart of our approach to processing questions as semistructured queries we will deploy a retrieval engine based on statistical language models, which have been used extensively within recent IR research, and which we believe provide a very natural setting for combining "evidence for answerhood" from multiple sources. Our team has successfully used language models for its work on semistructured documents (such as XML documents and web documents) at CLEF, INEX, and TREC [4, 5, 42, 45, 46, 48, 62, 72, 73, 78, 78, 79, 80]. We first explain our general strategy for putting our XML IR achievements to work for QA; after that we detail the envisaged methods for the research aims (a)–(d).

Overall Strategy. The Initiative for the Evaluation of XML Retrieval [30, INEX] has set up test collections to evaluate two kinds of XML retrieval tasks: *content only* and *content and structure*. As stated previously, with each question we associate a set of *content and structure* queries that will be answered against a text (and data) collection with various forms of structural, syntactic, and semantic mark-up. We aim to have a baseline QA-as-SSIR system up and running as soon as possible after the start of the project. While we advocate a very strongly data-driven approach in pursuing are research aims (a)–(d), the envisaged baseline system will most likely be a hybrid system, mixing data-driven and rule-based efforts.

The reason we strive to have a baseline system up and running early on in the project, is the following. The history of information retrieval is a showcase of theoretical progress going hand-in-hand with experimental evaluation. The scientific evaluation of IR systems is rooted in the Cranfield experiments [17, 18]. This has been continued in recent years within the framework of the Text REtrieval Conference [TREC, 25, 92], and its various regional and task-specific counterparts such as CLEF [16], NTCIR [71], and INEX [30]. QASSIR will re-use existing collections and test-suites as far as possible and extend them were the

[1] Possible queries, using the INEX 2004 query language, generated for "*Which painter is famous for painting sunflowers?*" are:

(1) `//sentence[about(.,painter) AND about(.,sunflowers)]//person`,
extracting persons from sentences "about" painters and sunflowers

(2) `//clause[contains(./verb,paint) AND about(./object,sunflowers)]//subject`,
extracting subjects of verb *paint* with object *sunflowers*

(3) `//Creation[about(./CreatedEntity, sunflowers)]//Creator`,
extracting semantic arguments *Creator* from "sunflower creation" events.

need arises. In particular, we are considering the following collections: First, the AQUAINT corpus of TREC's QA track. Second, the test suites of CLEF's (Dutch) QA track. Third, the corpora used at INEX. Where the need arises, we will supplement these with new corpora and/or new assessments. Throughout the project we will assume that all required document processing can be done off-line, at indexing time.

Addressing Our Research Aims Next, we outline the proposed strategies for each of our four research aims (a)–(d).

(a) Document Processing. The required annotations will be realized using existing standard tools as well as NLP tools developed locally [7]. These tools will produce XML annotations, each of which can be stored as separate 'stand-off" annotation, in separate XML files, to simplify maintenance. A baseline approach towards indexing that supports queries involving multiple, possibly overlapping annotations is to automatically merge the annotations to a single XML document providing full access to the extracted information, at query time. In order to produce well-formed XML mark-up after merging, one can use a mixture of inline and stand-off (with character off-sets to refer to original collection text) annotation schemes [31].

While this solution to the problem of overlapping markup is a sensible baseline, it "solves" the problem at the physical level [21]. To understand the mathematical nature of overlapping markup, [84, 85] introduced the so-called GODDAG model. Dekhtyar and Iacob [20], Iacob and Dekhtyar [28] proposed to query this conceptual model directly using an extended version of XPath. The formal elegance and mathematical transparency of this approach, motivates a change to XML-based Concurrent Markup schemes [28] to store multiple layers of annotation of the same data in a later stage of the project.

In the retrieval step ((d) below) we combine structural and content aspects; for this reason, we will index both the text and the XML structure of the collection, i.e., we build inverted indices both for words and for XML elements in the annotation [5, 24, 97].

(b) Query Languages. At INEX, an XPath-like query language has been introduced, one which is appropriate for XML IR. The language has a syntax similar to XPath, but does not have the same strict semantics [3, 81, 88]. Its most notable feature is the `about` function having a best-match semantics based on the relevance of the document component for the expressed information need; see Query 1 in the example above. We will take the 2004 version of the INEX query language as our starting point, but note that enrichments are called for: (i) multiple annotations of the same data—a step outside the common tree-like document structure; (ii) for annotation schemes over the same document collection, it should be possible to express relationships between elements in these schemes, such as order and proximity. The question-answer pair data used in aim (c) below will help inform us about the required expressivity.

A first practical step towards such a rich language is Extended XPath [28]. Developing extensions of the INEX query language that implement all required features will be a key component of the project.

(c) Question-to-Queries Mapping. To associate questions with sets of semistructured queries the project will ultimately aim for a purely data-driven approach. Initially, though, to quickly get an early baseline version going, we mix data-driven with hand-written methods. For instance, the question *How many seats are there in the cabin of a Concorde* can automatically be mapped to the XPath query

(4) `CHUNK[@type="NP"][.//WORD[@pos="CD"]][contains(@text,seat)]`,

extracting noun chunks (phrases) containing numerals and the word "seats". Initially, this mapping can be performed by a statistical machine learning-based question classifier followed by hand-coded rule-based query generator. In a later stage we will take the question processing module, will expand the set of hand-coded conversion rules to account for new types of annotation generated in step (a). Then, using manually corrected output of the module for a substantial number of questions as training base, we will extend our machine learning-based question classifier to complement and correct where necessary the manually created rules.

Finally, we will pursue a third alternative to generating queries: using a large collection of question-answer pairs to *automatically* induce the mapping, following a data-driven approaches to QA pioneered in [76, 82] and using in-house methods for extracting question-answer pairs from the Web [33]; the main novelty here will be that the induced mappings will concern not only lexical targets (as in [76, 82]) but also structural ones.

(d) Element Retrieval. Given the queries generated from an input question, we view the document (and data) collections as a large set of answer elements of different granularity. Our proposal is to use a three-step procedure to retrieve and rank answer elements: *decomposition* (where we break the queries up into a number of IR or database queries, together with sources against which these queries have to be answered, each of which constrains different elements), *retrieval* (where we collect evidence about relevant elements from each source), and *mixture* (where we use the structural constraints in the queries to mix the evidence from the multiple sources so as to provide a ranking of answer elements). For the retrieval and mixture steps we use a multinomial language modeling approach which has proved particularly effective in settings where evidence from multiple sources needs to be combined, such as web retrieval [48, 53, 73], XML retrieval [72, 78], and multi-lingual retrieval [26, 99].

The main challenge here is to extend the semistructured retrieval methods to work with multiple, possibly conflicting hierarchies, corresponding to different annotation layers. Other scientifically interesting aspects of the retrieval component are "partial" evaluation of a query when some of the annotations are not available during retrieval but can be generated on-the-fly, and propagation of retrieval scores in evaluation of queries with complex logic (e.g., containing AND's and OR's).

Further Reflections. One important difference of the proposed approach from the traditional pipeline architecture, is the flexibility and control of our unified QA-as-SSIR approach. Since our query evaluation module allows non-exact structure matching, we can easily add or remove layers of annotation, effectively adjusting the boundary between the retrieval and extraction stages in the classical architecture. This has theoretical, practical and methodological benefits.

First, re-casting QA as evaluation of semistructured queries allows us to study directly the interaction between retrieval and extraction steps, which is difficult to address in the traditional setting. How do different types of pre-processing of the data collection, and the use of richer indexes affect the end-to-end performance of the QA system? As to the practical side, while in a perfect world all text processing is done off-line, making the online stage much faster, in practice, this might not be possible for all collections (e.g., the Web). Our unified approach to QA allows us to find a compromise between the time and computational resources spent offline pre-processing the data, and spent online, during the actual question processing, on adding the required annotations. From a methodological perspective, the approach gives us transparent access to multiple data sources, with different degrees of structuredness. E.g., Query (3) in the example above can be used to extract answers from automatically annotated newspapers and from a structured database of information about artifacts (e.g., similar to IMDb), while Query (1) can be evaluated on any free text collection annotated with sentence boundaries and named entities.

6.3 Scientific Significance

The significance of the project lies in several areas, both societal and scientific. With its overall aim of laying the groundwork for shrink-wrapped, trainable QA systems that will enable users to obtain real answers to real questions, the long-term societal benefits of the proposal are obvious. The main scientific interest of the proposal is in considering QA as a task of "answer snippet retrieval" against data collections with multiple annotation layers—the results of offline processing. By tackling QA as a semistructured retrieval task in this manner we facilitate, and provide a formal grounding for, the incorporation of multiple data sources and multiple layers of (linguistic) analysis in the QA process. More generally, our proposal of a retrieval model which takes into account richer representations has implications broader than QA: we expect it to find new applications, from sentiment analysis through retrieval-assisted-by-layout, to integration of retrieval technology with the semantic web.

The scientific interest of our proposal for document retrieval and, in particular, for XML retrieval is the following. The community now has emerging standards for formulating *content and structure* information needs; however, these have only been tested on mostly layout-oriented annotation schemes—not on semantically rich annotation schemes, certainly not on multiple ones. A semistructured retrieval task with multiple, overlapping tag hierarchies, poses an interesting challenges for the XML retrieval community, in terms of query format, indexing, and retrieval proper.

Finally, the proposed research is an ideal breeding ground for the interaction between natural language processing (NLP) and IR, a long-standing research issue [86]. By redefining QA as a "ranked answer snippet" retrieval task, it directly addresses some of the challenges raised by Spärck Jones [83], and creates opportunities for systematically investigating the impact on the retrieval performance of bringing in additional, linguistically rich annotation layers. Furthermore, applied NLP in the 1970s and early 1980s was mostly based on extensive hand-coding of linguistic and real-world knowledge. These approaches worked quite well as, for instance, natural language interfaces to databases, but proved to be of little use for processing less restricted texts [8]. Data-driven NLP became the dominant paradigm in the 1990s, making heavy use of machine learning and statistical processing [60]. Our proposal pushes linguistically informed, and semantically rich, QA in a very strongly data-driven direction.

6.4 Related Research

Much recent QA research is corpus- or extraction-based, aimed at returning highly relevant snippets from a corpus of text documents. In contrast, QA systems developed in the 1960s and 1970s were mostly natural language front-ends to database engines [66, Chapter 2]. State-of-the-art QA systems combine extraction-based strategies with methods based, ultimately, on database/knowledge lookups; see, e.g., [15]. For performance and scaleability reasons, there is a trend to move much of the computationally expensive extraction off-line, thus "returning" to mostly lookup-based architectures. Prager et al. [75]'s work provides an early example of this trend, by identifying and storing semantic types at index time. Our proposal takes this to an extreme, by moving *all* extraction off-line, and recasting the QA process as a retrieval process.

Litkowksi [54, 55] has developed an XML-based QA system, where passages are parsed to produce parse trees which are then used to provide linguistically rich annotations; QA is then performed by searching the passages using (strict) XPath queries generated by a manually-built rule-based system. Our proposal differs in many respects, chief amongst which are the fact that we want to allow for multiple annotation schemes, that we aim to automatically generate question-to-queries mappings, and allow for a query language that allows multiple hierarchies and in which both structural and content constraints are not interpreted strictly. Recently, Bouma et al. [11] automatically annotated entire CLEF QA collection for Dutch with syntactic parses and used the annotations for both offline information extraction and online matching of syntactic structures. In our proposal we go one step further by making structure matching part of the retrieval. In our existing work on QA [1, 2, 34, 35, 36, 37, 38, 39], we employ a multi-stream approach in which multiple QA strategies attempt to identify candidate answer. Some of these access existing knowledge sources (WordNet [98], Wikipedia [96]), others use purpose-built knowledge bases, using information extraction techniques, while yet others perform online extraction. In the latest version of our QA system [31], we do *all* corpus annotation at indexing time, reducing the online extraction stage to evaluating XPath queries on passages retrieved using standard IR methods. This is a first practical step to implementing QA as semistructured retrieval, and it has given rise to the QASSIR proposal.

Ogilivie [4]'s proposal is close in spirit to ours. He proposes to replace the traditional retrieval step early in the pipeline of extraction-based QA systems, with an XML-retrieval step. Since XML retrieval provides better pinpointing facilities than traditional (document, passage, or sentence-level) retrieval, it is assumed that later stages of the QA pipeline will perform better. In contrast, we have different motivations (performance concerns, methodological issues, and the desire to provide a formal grounding to QA) and these lead us to integrate the whole QA process into a "single" (semistructured) retrieval process.

As to question-to-queries mapping, in most modern QA extraction-based systems this is organized around a question classifier, which in turn often relies on a manually built classification of answer types. Data-driven proposals similar in spirit to ours, but mapping only to queries with no or little structure, are due to Chakrabarti [13], Soricut and Brill [82]; [76] also proposes a conditional exponential model to filter candidate snippets. Question-to-*structured*-query mapping has been, continues to be, at the heart of much work on natural language front-ends to database; see e.g., [93, 94]. Most of this work depends on elaborately designed hierarchies of question types [8], but recent work on automated mapping of questions to formal query languages is promising [74]. In recent exploratory work [31], we use a combination of data-driven and rule-based methods for question classification and mapping to semistructured queries.

Content and structure XML retrieval is one of the main tasks evaluated at INEX 2002, 2003, 2004 and, again, at 2005. Our decompose-retrieve-mix approach to answering *content and structure* topics was very effective for XML retrieval at INEX 2003 and 2004, achieving top scoring results [30]; at the time of writing INEX 2005 is ongoing. The underlying retrieval model (language modeling) has proved to be very effective for structured document retrieval, and, more generally, for settings in which relevancy evidence from multiple sources needs to be combined. *Content and structure* XML (and, more generally, semistructured) queries, can be viewed as a natural extension of traditional fielded search [9]. Moreover, we have shown that the use of structured queries improves early precision of the focused retrieval, while structure is used as a search hint, not as a strict requirement [3]. Much recent work goes back to ideas due to Clarke et al [14]. There are several recent proposals for a *content and structure* query language that allows one to query multiple hierarchies; the proposal due to IBM Haifa [12] is especially relevant.

6.5 Local Embedding

Scientific Embedding. The planned research will be carried out within the Information and Language Processing Systems (ILPS) group at the Informatics Institute of the University of Amsterdam. The group's leading theme is *intelligent information access*. Tackling this theme requires a mixture of information retrieval, applied natural language processing, and knowledge representation and reasoning. The ILPS Group (with around twenty staff, post-docs, and Ph.D. students) is a major player in the experimental evaluation of IR systems. The group has become a regular participant in the Cross-Language Evaluation Forum [CLEF, 16]; the Text REtrieval Conference [TREC, 87]; and the Initiative for the Evaluation of XML Retrieval [INEX, 30].

ILPS research activities that are directly related to the present proposal include our world-class research on XML retrieval [40, 41, 42, 44, 45, 46, 63, 78, 79], question answering for English and for Dutch [1, 34, 35, 37], data-driven natural language processing [2, 32, 70], as well as web retrieval based on language modeling and/or multiple sources of evidence [34, 47, 51, 69]. These activities will benefit from, and directly feed into, the proposed research. Further activities within ILPS that will provide synergy with the proposed research are our long-standing work on (online) opinion extraction [43, 49, 64], and on multilingual retrieval (against multiple, distributed collections) [50, 52], as well as our recent work on the foundations of semistructured data [6, 61, 62], document-centric XML query languages [81, 88], and on ontology-based retrieval, filtering, and resource selection [29, 77, 89].

Within the ILPS group, a number of IR related projects are ongoing or have recently been completed: IR for Question Answering (1999–2003, [66]), XML Retrieval (2002–2006), Question Answering (2001–2007, 2004–2007, 2004-2007), Ontology-based Information Retrieval (2004–2008), Information Access for Blogs (2005–2008), QA against Wikipedia (2005–2008), Metadata and IR (2006–2010). Together, these activities ensure that the the present proposal will be solidly embedded in local research activities and yield a broad range of long-term scientific benefits.

Technical Embedding. It is important to point out that the implementation work required for this project will not start from scratch. The ILPS group has considerable infrastructure, including a full-fledged IR engine [67], which implements language modeling on top of the Lucene [56] open source engine, and XML IR engine, as well as competitive open domain QA systems [37, 38, 39, 68]. The group is actively developing evaluation test suites (organizing WebCLEF [95] and Dutch QA at CLEF [16]) NLP and IR tools, ranging from annotation helpers such as anaphora resolvers and named-entity taggers to extensions of publicly-available IR suites for scaleable and effective handling of large collections of text, as well as XML and semistructured data. Components of the QA system developed at ILPS will serve as starting points or prototypes for the subsystems that need to be developed within the proposed project. For example, the existing question classifier, based on machine learning and representation of a question as a set of syntactic and semantic features, can be used as a base for development of the question-to-query mapping module. Other components of ILPS' QA system, like type checking [77], answer filtering and re-ranking modules [1], will also be used to set up a working baseline infrastructure.

7 Work Program

The project will run from mid 2006 until mid 2010. Each year, the experimental evaluation will be based on participations in that year's TREC QA track, as well as in suitable tasks at the annual INEX workshops (e.g., the natural language task, which was introduced in 2004, and which is aimed at assessing the effectiveness of mapping information needs in natural language to the INEX query language). Publication venues that we will aim for include IR related conferences such as SIGIR and ECIR, while work on suitable query languages for the project will be submitted to conferences that cater for IR and database researchers, such as CIKM and WebDB. We expect the Ph.D. student to be employed on this project to have at least three journal/major conference publications by the end of the project.

7.1 Stepwise Planning

We aim to get a baseline system working during year 1, and make successive refinements in later years.

Year 1 (focus on research aims (a) and (c) Ph.D. student: Preparatory phase. Read relevant literature. Write literature overview paper. Get acquainted with document processing tools, and with the local XML retrieval and QA systems, and with the evaluation methods available. Take the INEX 20004 *content and structure* query language, the University of Amsterdam's existing question classifier to create a baseline question-to-queries mapping, as well as its existing filtering and answer merging modules. Collaborate on TREC QA and INEX submissions within the ILPS team.

Programmer: move locally available tagging and extraction off-line, creating multiple "semantic" indexes. Solve engineering problems and align the annotation layers. Take the local Lucene-based XML retrieval engine as starting point for experiments on QA as semistructured retrieval.

Year 2 (focus on research aims (b) and (c) Ph.D. student: Focus on question-to-queries mapping, exploring a number of variations on the initial query language, various feature sets, and creating additional question-answer pairs for training purposes if the need arises (building on local experience in creating QA test collections [57, 58, 59]). Integrate question-to-queries mapping with the semistructured retrieval engine. Write journal paper on question-to-queries mapping for IR and/or NLP audiences. Collaborate on TREC QA and INEX submissions within the ILPS team.

Programmer: infrastructure for extensive machine learning experiments for question-to-queries mapping. Extend existing Lucene-based semistructured retrieval engine to accomodate features of the new query language and its variations.

Year 3 (focus on research aim (d) Ph.D. student: refine the retrieval model; integrate filtering and re-ranking into the retrieval engine, so as to arrive at a "one step" approach to QA which is built around semistructured retrieval. Subsequent refinements of the retrieval model will focus on integrating tagging and extraction uncertainties in the retrieval model. Write journal paper on language models and semistructured retrieval. Explore the impact of additional annotation layers and knowledge sources. Collaborate on TREC QA and INEX submissions within the ILPS team.

Programmer: bring question-to-queries mapping and semistructured retrieval modules up to a state when the system can be made available to real world users. Launch/maintain online demo. Use query logs/visitor feedback for debugging and refinement.

Year 4 Ph.D. student: refining query processing; strict vs. approximate interpretations of content and structure constraints in the queries. Finish the thesis.

Programmer: document and package modules. Release semistructured information retrieval system underlying the QA system. Document/release further resources generated by the project.

Submit final report to NWO.

7.2 Training and Education

Training and education are aimed at developing scientific expertise, and acquiring professional competencies. With respect to the former, the Ph.D. student should become able to fully understand, critically analyze, and contribute to research at the frontiers of science. We address these issues both informally and formally. At an informal level, the ILPS group provides a stimulating intellectual climate, with a range of world-class experts working on related projects, with regular visitors (both academic and industrial), and with regular events (such as multiple seminars, small-scale workshops, national and international events). More formally, supervision and training are governed by a number of instruments. First, students have two supervisors; supervisors and students together draw up a highly individual training and education plan, which is revised (if necessary) annually and which includes undergraduate and masters courses that supplement or broaden the student's expertise. Furthermore, students will follow graduate courses within the SIKS research school and at international summer schools. A working visit to a foreign university or research institute is a key ingredient of the plan. The training and education plan also covers more general professional competencies. Towards the end of their first year, Ph.D. students write a detailed proposal for their thesis research and submit it to their supervisors and, after their approval, to an independent institute-wide doctoral committee that monitors Ph.D. students' progress on an annual basis. These formal progress evaluations are themselves subject to monitoring by external experts.

We actively work to equip our students to function well in the professional environment of a university or research institute. We encourage the development of academic leadership skills by involving Ph.D. students in various aspects of academic life, such as project management, proposal writing, seminar/workshop organization, teaching, etc. Since teaching a topic is one of the best ways to fully master it, Ph.D. students take active part in the supervision of undergraduate students during final year projects; they also act as teaching assistants for undergraduate courses.

Both applicants practice an "open door" environment in which informal meetings occur naturally and frequently. In addition, the local research team members will have formal project meetings on a weekly basis, discussing both the research and training aspects, including mundane practical issues. Our attitude toward Ph.D. students is based on doing collaborative research where the initiative will gradually shift from the supervisors to the students, thus avoiding a strict (and, in our opinion counterproductive) dichotomy between supervisor and student.

In 2007 the 30th Annual International ACM SIGIR Conference will be held in Amsterdam; this will provide the Ph.D. student with unique opportunities networking and contacts. We expect the student to become actively involved in the local organization of the conference and of special student sessions.

8 Expected Use of Instrumentation

The project will require substantial data storage facilities that are not provided for in the regular budget of the university. We estimate a need for at least 3 terabytes of storage in a RAID-5 configuration.

9 Literature

Five most relevant publications of the research team

[1] V. Jijkoun and M. de Rijke. Answer selection in a multi-stream open domain question answering system. In *Proceedings 26th European Conference on Information Retrieval (ECIR'04)*, LNCS. Springer, 2004.

[2] V. Jijkoun, M. de Rijke, and J. Mur. Information extraction for question answering: Improving recall through syntactic patterns. In *Proceedings of the 20th International on Computational Linguistics (COLING 2004)*, 2004.

[3] J. Kamps, M. Marx, M. de Rijke, and B. Sigurbjörnsson. Structured queries in XML retrieval. In *Fourteenth Conference on Information and Knowledge Management (CIKM 2005)*, 2005.

[4] P. Ogilvie. Retrieval using structure for question answering. In V. Mihajlovic and D. Hiemstra, editors, *Proceedings of the First Twente Data Management Workshop (TDM'04)*, pages 15–23, 2004.

[5] B. Sigurbjörnsson, J. Kamps, and M. de Rijke. Processing content-oriented XPath queries. In *Thirteenth Conference on Information and Knowledge Management (CIKM 2004)*, 2004.

Other references

[6] L. Afanasiev, M. Franceschet, M. Marx, and M. de Rijke. CTL model checking for processing simple XPath queries. In *Proceedings Temporal Representation and Reasoning (TIME 2004)*. IEEE Computer Society Press, 2004.

[7] D. Ahn, S. Fissaha Adafre, V. Jijkoun, and M. de Rijke. The University of Amsterdam at Senseval-3: Semantic Roles and Logic Forms. In *Proceedings of Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, pages 49–53, 2004.

[8] J. Allen. *Natural Language Understanding*. Addison Wesley, 1994.

[9] R. Baeza-Yates and B. Ribeiro-Neto, editors. *Modern Information Retrieval*, 1999. ACM Press, New York and Addison Wesley Longman, Harlow.

[10] R. Bernardi, V. Jijkoun, G. Mishne, and M. de Rijke. Selectively using linguistic resources throughout the question answering pipeline. In *Proceedings 2nd CoLogNET-ElsNET Symposium*, 2003.

[11] G. Bouma, J. Mur, G. van Noord, L. van der Plas, and J. Tiedemann. Question answering for dutch using dependency relations. In *Proceedings of the CLEF 2005 Workshop*, 2005. To appear.

[12] A. Broder, D. Carmel, Y. Maarek, M. Mandelbrod, and Y. Mass. Extendign the XML fragment model to support querying over annotated text. In *Proceedings of the Joint Workshop on XML, IR, and DB*, pages 7–11, 2004.

[13] S. Chakrabarti. Discovering links between lexical and surface features in questions and answers. In *Proceedings Workshop on Link Analysis and Group Detection (LinkKDD2004)*, 2004.

[14] C. Clarke, G. Cormack, and F. Burkowski. An algebra for structured text search and a framework for its implementation. *The Computer Journal*, 38:43–56, 1995.

[15] C. Clarke, G. Cormack, T. Lynam, C. Li, and G. McLearn. Web reinforced question answering. In *Proceedings of the Tenth Text REtrieval Conference (TREC 2001)*, 2001.

[16] CLEF. Cross Language Evaluation Forum, 2004. URL: `http://www.clef-campaign.org/`.

[17] C. W. Cleverdon. Report on the testing and analysis of an investigation into the comparative efficiency of indexing systems. Technical report, College of Aeronautics, Cranfield UK, 1962.

[18] C. W. Cleverdon. The Cranfield tests on index language devices. *Aslib*, 19:173–192, 1967.

[19] T. Clifton and W. Teahan. Knowing-aboutness: Question-answering using a logic-based framework. In *27th European Conference on Information Retrieval (ECIR)*, 2005.

[20] A. Dekhtyar and I. Iacob. A framework for management of concurrent xml markup. *Data Knowl. Eng.*, 52(2): 185–208, 2005.

[21] S. DeRose. Markup overlap: A review and a horse. In *Proceedings of Extreme Markup Languages 2004*, 2004.

[22] A. Echihabi and D. Marcu. A noisy-channel approach to question answering. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 16–23, 2003.

[23] N. Fuhr, M. Lalmas, S. Malik, and Z. Szlávik, editors. *Proceedings of the INitiative for the Evaluation of XML Retrieval (INEX 2004)*, LNCS, 2005. Springer Verlag.

[24] T. Grust. Accelerating XPath Location Steps. In *Proc. SIGMOD*, pages 109–120. ACM Press, 2002. ISBN 1-58113-497-5.

[25] D. K. Harman, editor. *The First Text REtrieval Conference (TREC-1)*, 1993. National Institute for Standards and Technology. NIST Special Publication 500-207.

[26] D. Hiemstra and W. Kraaij. Twenty-One at TREC-7: Ad-hoc and cross-language track. In E. Voorhees and D. Harman, editors, *The Seventh Text REtrieval Conference (TREC-7)*, pages 227–238. National Institute for Standards and Technology. NIST Special Publication 500-242, 1999.

[27] E. Hovy, U. Hermanjakob, C.-Y. Lin, and D. Ravichandran. Using knowledge to facilitate pinpointing of factoid answers. In *Proceedings COLING 2002*, 2002.

[28] I. E. Iacob and A. Dekhtyar. Towards a query language for multihierarchical xml: Revisiting xpath. In *The Eighth International Workshop on the Web and Databases (WebDB 2005)*, 2005.

[29] L. IJzereef, J. Kamps, and M. de Rijke. Biomedical retrieval: How can a thesaurus help? In *Proceedings of the 4th International Conference on Ontologies, Database and Applications of Semantics (ODBASE'05)*. Springer, 2005.

[30] INEX. INitiative for the Evaluation of XML retrieval, 2004. URL: http://inex.is.informatik.uni-duisburg.de:2003/.

[31] V. Jijkoun, D. Ahn, E. Tjong Kim Sang, K. Müller, and M. de Rijke. The University of Amsterdam at QA@CLEF 2005. In *Working Notes of the CLEF 2005 Workshop*, 2005.

[32] V. Jijkoun and M. de Rijke. Enriching the output of a parser using memory-based learning. In *Proceedings 42nd Annual Meeting of the Association for Computational Linguistics (ACL 2004)*, pages 311–318, 2004.

[33] V. Jijkoun and M. de Rijke. Retrieving answers from frequently asked questions pages on the web. In *Thirteenth Conference on Information and Knowledge Management (CIKM 2005)*, 2005.

[34] V. Jijkoun, J. Kamps, G. Mishne, C. Monz, M. de Rijke, S. Schlobach, and O. Tsur. The University of Amsterdam at TREC 2003. In *TREC 2003 Working Notes*. National Institute for Standards and Technology, 2003.

[35] V. Jijkoun, G. Mishne, and M. de Rijke. Building infrastructure for Dutch question answering. In A. de Vries, editor, *Proceedings DIR 2003*, 2003.

[36] V. Jijkoun, G. Mishne, and M. de Rijke. Preprocessing documents to answer Dutch questions. In *Proceedings BNAIC'03*, 2003.

[37] V. Jijkoun, G. Mishne, and M. de Rijke. How frogs built the Berlin Wall. In *Proceedings CLEF 2003*, LNCS. Springer, 2004.

[38] V. Jijkoun, G. Mishne, M. de Rijke, S. Schlobach, D. Ahn, and K. Müller. The University of Amsterdam at QA@CLEF 2004. In *Working Notes for the CLEF 2004 Workshop*, 2004.

[39] V. Jijkoun, G. Mishne, C. Monz, M. de Rijke, S. Schlobach, and O. Tsur. The University of Amsterdam at the TREC 2003 Question Answering Track. In *Proceedings TREC 2003*, pages 586–593, 2004.

[40] J. Kamps, M. de Rijke, and B. Sigurbjörnsson. Topic field selection and smoothing for XML retrieval. In A. P. de Vries, editor, *Proceedings of the Fourth Dutch-Belgian Information Retrieval Workshop (DIR 2003)*, pages 69–75, 2003.

[41] J. Kamps, M. de Rijke, and B. Sigurbjörnsson. The University of Amsterdam at INEX 2003. In *INEX 2003 Workshop Proceedings*, 2003.

[42] J. Kamps, M. de Rijke, and B. Sigurbjörnsson. The Importance of Length Normalization for XML Retrieval. *Information Retrieval*, 8(4):631–654, 2005.

[43] J. Kamps and M. Marx. Words with attitude. In *Proceedings of the 1st International Conference on Global WordNet*, pages 332–341. CIIL, Mysore India, 2002.

[44] J. Kamps, M. Marx, M. de Rijke, and B. Sigurbjörnsson. The importance of morphological normalization for XML retrieval. In N. Fuhr, N. Gövert, G. Kazai, and M. Lalmas, editors, *Proceedings of the First Workshop of the INitiative for the Evaluation of XML retrieval (INEX)*, pages 41–48. ERCIM Publications, 2003.

[45] J. Kamps, M. Marx, M. de Rijke, and B. Sigurbjörnsson. XML retrieval: What to retrieve? In C. Clarke, G. Cormack, J. Callan, D. Hawking, and A. Smeaton, editors, *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 409–410. ACM Press, New York NY, 2003.

[46] J. Kamps, M. Marx, M. de Rijke, and B. Sigurbjörnsson. Best-match querying from document-centric XML. In S. Amer-Yahia and L. Gravano, editors, *Proceedings Seventh International Workshop on the Web and Databases (WebDB 2004)*, pages 55–60, 2004.

[47] J. Kamps, M. Marx, C. Monz, and M. de Rijke. Exploiting structure for information retrieval. In M.-F. Moens, R. De Busser, D. Hiemstra, and W. Kraaij, editors, *Proceedings of the Third Dutch Belgian Information Retrieval Workshop (DIR-2002)*, pages 19–26. K.U. Leuven, Leuven, 2002.

[48] J. Kamps, G. Mishne, and M. de Rijke. Language models for searching in Web corpora. In E. M. Voorhees and L. P. Buckland, editors, *The Thirteenth Text REtrieval Conference (TREC 2004)*. National Institute of Standards and Technology. NIST Special Publication 500-261, 2005.

[49] J. Kamps, R. J. Mokken, M. Marx, and M. de Rijke. Using WordNet to measure semantic orientation of adjectives. In *LREC 2004*, 2004.

[50] J. Kamps, C. Monz, M. de Rijke, and B. Sigurbjörnsson. The University of Amsterdam at CLEF-2003. In C. Peters, editor, *Results of the CLEF 2003 Cross-Language System Evaluation Campaign*, pages 71–78, 2003.

[51] J. Kamps, C. Monz, M. de Rijke, and B. Sigurbjörnsson. Approaches to robust and web retrieval. In *The Twelfth Text REtrieval Conference (TREC 2003)*. National Institute for Standards and Technology, 2004.

[52] J. Kamps, C. Monz, M. de Rijke, and B. Sigurbjörnsson. Language-dependent and language-independent approaches to cross-lingual text retrieval. In C. Peters, M. Braschler, J. Gonzalo, and M. Kluck, editors, *Cross-Language Information Retrieval, CLEF 2003*, Lecture Notes in Computer Science. Springer, 2004.

[53] W. Kraaij, T. Westerveld, and D. Hiemstra. The importance of prior probabilities for entry page search. In K. Järvelin, M. Beaulieu, R. Baeza-Yates, and S. H. Myaeng, editors, *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 27–34. ACM Press, New York NY, USA, 2002.

[54] K. Litkowksi. Question answering using XML-tagged documents. In *Proceedings of the Eleventh Text REtrieval*

Conference (TREC-11), 2003.

[55] K. Litkowksi. Use of metadata for question answering and novelty tasks. In *Proceedings of the Twelfth Text REtrieval Conference (TREC 2003)*, 2004.

[56] Lucene. Jakarta Lucene: A high-performance, full-featured text search engine library, 2004. URL: http://jakarta.apache.org/lucene/.

[57] B. Magnini, S. Romagnoli, A. Vallin, J. Herrera, A. Peñas, V. Peinado, F. Verdejo, and M. de Rijke. Creating the DISEQuA Corpus: a Multilingual Test Set for the Monolingual Question Answering Tasks at CLEF 2003. In C. Peters, editor, *Working Notes for the CLEF 2003 Workshop*, 2003.

[58] B. Magnini, S. Romagnoli, A. Vallin, J. Herrera, A. Peñas, V. Peinado, F. Verdejo, and M. de Rijke. The Multiple Language Question Answering Track at CLEF 2003. In C. Peters, editor, *Working Notes for the CLEF 2003 Workshop*, 2003.

[59] B. Magnini, A. Vallin, C. Ayache, G. Erbach, A. Peñas, M. de Rijke, D. Rocha, K. Simov, and R. Sutcliffe. Overview of the CLEF 2004 Multilingual Question Answering Track. In *Working Notes for the CLEF 2004 Workshop, 2004*, 2004.

[60] C. Manning and H. Schütze. *Foundations of Statistical Natural Language Processing*. MIT Press, 1999.

[61] M. Marx. Conditional XPath, the first order complete XPath dialect. In *Proceedings of PODS'04*, 2004.

[62] M. Marx and M. de Rijke. Semantic characterizations of navigational XPath. In V. Mihajlovic and D. Hiemstra, editors, *Proceedings of the First Twente Data Management Workshop (TDM'04)*, pages 67–73, 2004.

[63] M. Marx, J. Kamps, and M. de Rijke. The University of Amsterdam at INEX-2002. In N. Fuhr, N. Gövert, G. Kazai, and M. Lalmas, editors, *INEX 2002 Workshop Proceedings*, pages 24–28, 2002.

[64] G. Mishne. Experiments with mood classification in blog posts. In *Style2005 - 1st Workshop on Stylistic Analysis Of Text For Information Access, at SIGIR 2005*. ACM Press, 2005.

[65] D. Moldovan, S. Harabagiu, R. Girju, P. Morarescu, F. Lacatusu, A. Novischi, A. Badulescu, and O. Bolohan. LCC Tools for Question Answering. In E. Voorhees and D. Harman, editors, *The Tenth Text REtrieval Conference (TREC 2002)*. National Institute for Standards and Technology. NIST Special Publication 500-251, 2003.

[66] C. Monz. *From Document Retrieval to Question Answering*. PhD thesis, University of Amsterdam, 2003.

[67] C. Monz and M. de Rijke. Shallow morphological analysis in monolingual information retrieval for Dutch, German and Italian. In C. Peters, M. Braschler, J. Gonzalo, and M. Kluck, editors, *Evaluation of Cross-Language Information Retrieval Systems, CLEF 2001*, volume 2406 of *Lecture Notes in Computer Science*, pages 262–277. Springer, 2002.

[68] C. Monz and M. de Rijke. Tequesta: The University of Amsterdam's textual question answering system. In E. M. Voorhees and D. K. Harman, editors, *The Tenth Text REtrieval Conference (TREC 2001)*, pages 519–528. National Institute for Standards and Technology. NIST Special Publication 500-250, 2002.

[69] C. Monz, J. Kamps, and M. de Rijke. The University of Amsterdam at TREC 2002. In Voorhees and Buckland [92], pages 603–614.

[70] K. Müller. Semi-automatic construction of a question treebank. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC 2004)*, 2004.

[71] NTCIR. NII-NACSIS Test Collection for IR systems, 2004. URL: http://research.nii.ac.jp/ntcir/index-en.html.

[72] P. Ogilvie and J. Callan. Language models and structured document retrieval. pages 33–44. ERCIM, 2003.

[73] P. Ogilvie and J. Callan. Combining structural information and the usee of priors in mixed named-page and homepagae finding. In *The Twelfth Text REtrieval Conference (TREC 2003)*. National Institute for Standards and Technology, 2004.

[74] A. Popescu, O. Etzioni, and H. Kautz. Towards a theory of natural language interfaces to databases. In *Proceedings of the Conference on Intelligent User Interfaces*, 2003.

[75] J. Prager, D. Radev, E. Brown, A. Coden, and V. Samn. The use of predictive annotation for question answering in TREC 8. In *Proceedings of the Eighth Text REtrieval Conference*, 2000.

[76] G. Ramakrishnan, S. Chakrabarti, D. Paranjpe, and P. Bhattacharyya. Is question answering an acquired skill? In *World Wide Web Conference 2004*, 2004.

[77] S. Schlobach, M. Olsthoorn, and M. de Rijke. Type checking in open-domain question answering. In *Proceedings of the 16th European Conference on Artificial Intelligence (ECAI 2004)*. IOS Press, 2004.

[78] B. Sigurbjörnsson, J. Kamps, and M. de Rijke. An element-based approch to XML retrieval. In N. Fuhr and S. Malik, editors, *Proceedings INEX 2003*, pages 19–26, 2004.

[79] B. Sigurbjörnsson, J. Kamps, and M. de Rijke. Multiple sources of evidence for XML retrieval. In *Proceedings 27th Annual International ACM SIGIR Conference (SIGIR 2004)*, 2004.

[80] B. Sigurbjörnsson, J. Kamps, and M. de Rijke. Processing content-and-structure queries for XML retrieval. In V. Mihajlovic and D. Hiemstra, editors, *Proceedings of the First Twente Data Management Workshop (TDM'04)*, pages 32–38, 2004.

[81] B. Sigurbjörnsson and A. Trotman. Queries, INEX 2003 working group report. In *Proceedings of the 2nd INEX Workshop*, 2004.

[82] R. Soricut and E. Brill. Automatic question answering: Beyond the factoid. In *Proceedings HLT/NAACL*, 2004.

[83] K. Spärck Jones. Is question answering a rational task? In *Proceedings 2nd ELSNET-CologNet Workshop*, 2003.

[84] C. Sperberg-McQueen and C. Huitfeldt. Concurrent document hierarchies in mecs and sgml. *Journal of the Association for Literary and Linguistic Computing*, 14(1):29–42, 1999.

[85] C. Sperberg-McQueen and C. Huitfeldt. GODDAG: A data structure for overlapping hierarchies. In *ACH-ALLC '99*, number 2023 in LNCS, pages 139–160, 2004.

[86] T. Strzalkowski. *Natural Language Information Retrieval*. Kluwer Academic Publishers, 1999.

[87] TREC. Text REtrieval Conference, 2004. URL: http://trec.nist.gov/.

[88] A. Trotman and B. Sigurbjörnsson. Narrowed Extended XPath I (NEXI), 2004. Available on the INEX 2004 web-site. URL: http://inex.is.informatik.uni-duisburg.de:2004/.

[89] W. van Hage, M. de Rijke, and M. Marx. Information retrieval support for ontology construction and use. In *Proceedings 3rd International Semantic Web Conference (ISWC 2004)*, LNCS. Springer, 2004.

[90] V. Vianu. A Web odyssey: from Codd to XML. In *Proc. PODS*, pages 1–15. ACM Press, 2001. ISBN 1-58113-361-8.

[91] E. Voorhees. Overview of the TREC 2003 question answering track. In *Proceedings Twelfth Text Retrieval Conference (TREC 2003)*, pages 54–68, 2004.

[92] E. M. Voorhees and L. P. Buckland, editors. *The Eleventh Text REtrieval Conference (TREC 2002)*, 2003. National Institute for Standards and Technology. NIST Special Publication 500-251.

[93] B. Webber. Questions, answers and responses: Interacting with knowledge-base systems. In *On Knowledge Base Management Systems (Islamorada)*, pages 366–402, 1985.

[94] B. Webber. Question answering. In *Encyclopedia of Artificial Intelligence*, pages 814–822. Wiley, 1992.

[95] WebCLEF. The CLEF Crosslingual Web Track, 2005. URL: http://ilps.science.uva.nl/WebCLEF/.

[96] Wikipedia. The free encyclopedia, 2004. URL: http://www.wikipedia.org/.

[97] I. H. Witten, A. Moffat, and T. C. Bell. *Managing Gigabytes: compressing and indexing documents and images*. The Morgan Kaufmann series in multimedia information and systems. Morgan Kaufmann Publishers, San Francisco CA, 1999.

[98] WordNet. A lexical database for the English language, 2004. URL: http://wordnet.princeton.edu/.

[99] J. Xu and R. Weischedel. A probabilistic approach to term translation for cross-lingual retrieval. In *Language Modeling for Information Retrieval*, chapter 6. Kluwer Academic Publishers, 2003.