GoWeb: Semantic Search and Browsing for the Life Sciences

Dissertation

zur Erlangung des akademischen Grades Doktoringenieur (Dr.-Ing.)

vorgelegt an der
Technischen Universität Dresden
Fakultät Informatik

eingereicht von

Dipl.-Inf. Heiko Dietze
geboren am 1. Februar 1980 in Dresden

Betreuer	Prof. Dr. Michael Schroeder, TU-Dresden, Fakultät Informatik
Gutachter	Dr. Albert Burger, Heriot-Watt University, School of Mathematical and Computer Sciences, Edinburgh

Tag der Verteidigung 20. Oktober 2010

Dresden, den 11.08.2010

ABSTRACT

Searching is a fundamental task to support research. Current search engines are keyword-based. Semantic technologies promise a next generation of semantic search engines, which will be able to answer questions. Current approaches either apply natural language processing to unstructured text or they assume the existence of structured statements over which they can reason.

This work provides a system for combining the classical keyword-based search engines with semantic annotation. Conventional search results are annotated using a customized annotation algorithm, which takes the textual properties and requirements such as speed and scalability into account. The biomedical background knowledge consists of the GeneOntology and Medical Subject Headings and other related entities, e.g. proteins/gene names and person names. Together they provide the relevant semantic context for a search engine for the life sciences. We develop the system GoWeb for semantic web search and evaluate it using three benchmarks. It is shown that GoWeb is able to aid question answering with success rates up to 79%.

Furthermore, the system also includes semantic hyperlinks that enable semantic browsing of the knowledge space. The semantic hyperlinks facilitate the use of the eScience infrastructure, even complex workflows of composed web services.

To complement the web search of GoWeb, other data source and more specialized information needs are tested in different prototypes. This includes patents and intranet search. Semantic search is applicable for these usage scenarios, but the developed systems also show limits of the semantic approach. That is the size, applicability and completeness of the integrated ontologies, as well as technical issues of text-extraction and meta-data information gathering.

Additionally, semantic indexing as an alternative approach to implement semantic search is implemented and evaluated with a question answering benchmark. A semantic index can help to answer questions and address some limitations of GoWeb. Still the maintenance and optimization of such an index is a challenge, whereas GoWeb provides a straightforward system.

CONTENTS

1 Motivation **1**

1.1 Definition of Open Problems . . . 2

1.2 Thesis Outline . . . 3

2 Background **5**

2.1 Semantic Search . . . 5

2.1.1 RDF-based semantic search engines . . . 7

2.1.2 Web-based semantic search engines . . . 9

2.1.3 Literature-based semantic search engines . . . 12

2.1.4 Summary . . . 14

2.2 Question Answering . . . 15

2.2.1 Types of Questions and Answers . . . 16

2.2.2 Answering Strategies . . . 17

2.2.3 Competitions . . . 20

2.3 Ontological Background Knowledge . . . 21

2.4 Data and Text Sources . . . 23

2.4.1 Document and Content Types . . . 25

2.4.2 Web search . . . 26

2.4.3 Automated Content Retrieval . . . 27

2.4.4 Summary . . . 36

2.5 Text Mining and Entity Recognition . . . 36

2.5.1 Finding Ontology Concepts in Text . . . 36

2.5.2 Entity Recognition . . . 41

2.5.3 Algorithmic principles for concept and entity recognition . . . 41

2.5.4 Summary . . . 44

2.6 Indexing and Ranking for Information Extraction . . . 45

2.6.1 Indexing Techniques . . . 45

2.6.2 Static Similarity Measures for Ranking . . . 49

2.6.3 Summary . . . 51

3 GoWeb – Semantic Web Search for the Life Science **53**

3.1 Algorithm . . . 54

3.1.1 Choosing an Algorithm for Annotation . . . 54

3.1.2 Implementing a Radix Tree for Annotation . . . 54

3.1.3 Runtime Assessment for the Annotation Algorithms . . . 56

3.1.4 Description of the Entity Recognition Algorithm . . . 58

3.1.5 Co-Occurrence-Based Filter . . . 58

3.2 Architecture . . . 58

3.3 Evaluation: Answering Research Questions 61
3.3.1 Genes and Functions 61
3.3.2 Symptoms and Diseases 61
3.3.3 Proteins, Diseases and Evidences 68
3.4 Comparison of GoWeb to other Approaches 71
3.5 Summary 74

4 Semantic Browsing 75
4.1 Semantic Hyperlinks 75
4.1.1 Semantic Web Services 76
4.1.2 Compositions of Web Services 77
4.1.3 Semantic Hyperlinks in GoPubMed Extended 77
4.1.4 Website Annotation with GoWeb 78
4.2 Architecture 81
4.3 Evaluation: Usability Study 83
4.4 Summary 84

5 Specialized Semantic Search Engines 87
5.1 GoPatents 88
5.1.1 Patents 88
5.1.2 Patent Classification Schemes 90
5.1.3 Patent Mining and Retrieval 91
5.1.4 GoFreePatentsOnline 92
5.1.5 GoPIZ 92
5.2 GoCell 95
5.3 MousePubMed 96
5.3.1 Extracting Gene Names, Anatomy Concepts and Developmental Stages . 96
5.3.2 Experiment Designs 98
5.3.3 Conclusion 101
5.4 GoECDC 102
5.4.1 Examples Knowledge-based Searching 103
5.5 Summary 108

6 Towards Answering Questions with Semantic Indexing 109
6.1 Semantic Index using a Sentence-based Approach 109
6.2 TREC Genomics 2006 Question Answering Benchmark 111
6.3 Evaluation 111
6.4 Conclusion 116

7 Conclusion and Future Work 117
7.1 Future Work 119

Bibliography 121

A Evidence Data for GoWeb 141
A.1 Evidences for GoWeb Results with Google questions 141
A.2 TREC Genomics 2006 Question and GoWeb Answers 150

B Additional Data for TREC Genomics Question Answering 157
B.1 Mapping of TREC Genomics 2006 Question to Query Parameters 157
B.2 Answers to TREC Genomics 2006 using TREC Genomics 2006 Corpus 162

PUBLICATIONS

- *Peer-Reviewed Article*
  Heiko Dietze and Michael Schroeder
  GoWeb: A semantic search engine for the life science web
  In: *Proceedings of the Intl. Workshop on Semantic Web Applications and Tools for the Life Sciences SWAT4LS*, Editors: Albert Burger, Adrian Paschke, Paolo Romano and Andrea Splendiani, November 2008.
  and
  In: *BMC Bioinformatics*, 10 (Suppl 10): S7, 2009. doi:10.1186/1471-2105-10-S10-S7
  Basis for the results presented in Chapter 3 of this thesis

- *Book Chapter* and *WorkShop*
  Heiko Dietze, Dimitra Alexopoulou, Michael R. Alvers, Liliana Barrio-Alvers, Bill Andreopoulos, Andreas Doms, Jörg Hakenberg, Jan Mönnich, Conrad Plake, Andreas Reischuck, Loïc Royer, Thomas Wächter, Matthias Zschunke, and Michael Schroeder.
  GoPubMed: Exploring PubMed with Ontological Background Knowledge
  In: *Bioinformatics for Systems Biology*, Editors: Stephen A. Krawetz, Humana Press, 2008.
  and
  In: *Ontologies and Text Mining for Life Sciences: Current Status and Future Perspectives*, Dagstuhl Seminar Proceedings, Nr. 08131, Editiors: Michael Ashburner, Ulf Leser,Dietrich Rebholz-Schuhmann, Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, Germany, 2008.

  Contribution: main author for writing the book chapter, part of the programming and maintenance team of the GoPubMed system
  Excerpts of this publication are used in the background section of this thesis

- *Book Chapter*
  Thomas Wächter, Dimitra Alexopoulou, Heiko Dietze, Jörg Hakenberg and Michael Schroeder
  Searching Biomedical Literature with Anatomy Ontologies
  In: *Anatomy Ontologies for Bioinformatics*, Editors: Albert Burger, Duncan Davidson and Richard Baldock, Springer, 2007

  Contribution: part of the programming and maintenance team of MousePubMed
  Publication used in Chapter 5 of this thesis

- *Peer-Reviewed Article*
  Dimitra Alexopoulou, Bill Andreopoulos, Heiko Dietze, Andreas Doms, Fabien Gandon, Jörg Hakenberg, Khaled Khelif, Michael Schroeder and Thomas Wächter,
  Biomedical word sense disambiguation with ontologies and meta-data: automation meets accuracy, In BMC Bioinformatics, 2009, 10:28, doi:10.1186/1471-2105-10-28.

  Contribution: data gathering, assist in the evaluation of the algorithms
  Excerpts of this publication are used in the background section of this thesis

- *WorkShop*
  Conrad Plake, Andreas Doms, Heiko Dietze, Thomas Wächter, Michael R. Alvers, and Michael Schroeder
  Ontology-based Assisted Curation, In Proceedings of the 3rd International Biocuration Conference, Berlin, Germany, April 16–19 2009.

  Contribution: GoWeb implementation, part of the programming and maintenance team of the GoPubMed system
  Not used as part of this thesis

CHAPTER 1

MOTIVATION

Searching is a fundamental task to support research. This holds true for nearly all research domains, including the life sciences. One of the reasons for the growing importance of fast and reliable search is the fast growth of scientific literature and available biological data. PubMed, one of the biggest literature databases, adds more than $5,000$ new scientific articles per day.

Nowadays, the common research pipeline is data driven. This means, scientist formulate hypotheses and perform experiments, most likely high throughput experiments, which generate enormous amounts of data. To verify their hypotheses scientist are not only required to integrate the experimental data with existing databases, but also to check existing facts possibly retrieved by searching and reading related publications. Searching is key to identify the relevant information.

Current search engines are keyword-based. Such search engines retrieve a list of websites based on a given keyword query and rank those by relevance to a topic. The search results often include for each document the title and a short textual extract containing the keywords.

Semantic technologies promise a next generation of search engines, which will be able to answer questions. Any new search engine or paradigm has to be able to stand its ground against the widely accepted standard of keyword-based search. The goal and proposed advantage of semantic search is to provide better search results. Improvements can be achieved through:

- better ranking strategies, leading to more relevant hits in the top of the search result list.
- reduction of redundant information achieved by grouping.
- increasing coverage to regard different aspects of a search task.
- transparency by explaining results in more detail.
- answering questions in a concise manner.

Ideally, semantic search “understands” the meaning and intended question behind the input and finds the relevant documents by “understanding” the content. But a computer does not “understand”. The algorithms used by computer systems can only work with the immediate user input extended by potentially relevant fact databases. The general approach of semantic search is to match the query against the internal fact database and search for corresponding facts and documents.

This association of facts from the background knowledge with entries in a database or occurrences in a text document is referred to as annotation. Semantic-aware systems fundamentally rely on such large and well designed background knowledge. Knowledge bases, where facts are connected via relationships, form a knowledge network formalized as ontologies. Ontologies are a specification of a conceptualization (Gruber, 1993).

The largest source of information for semantic search is unstructured text. Structured and semantically typed factual statements repositories exist, but are magnitudes smaller leaving a semantic gap. Current approaches for semantic search either apply natural language processing to unstructured text or make direct use of factual statements.

The semantic gap motivates a continued need for classical search using unstructured text. On the other side standard search provides only a listing of ranked results. This is insufficient for more advanced applications. The combination of keyword search and semantics may help to improve the acceptance of semantic search (Guha et al., 2003).

1.1 Definition of Open Problems

The goal this thesis is to contribute advances in the scientific field of semantic search and related algorithms for the following open problems.

Open Problem 1: Combination of semantics and ontological background knowledge with classical keyword search

Keyword based search can profit from semantic annotation and ontologies as background knowledge. The task is to combine the simplicity of keyword search with the advanced semantic features. Furthermore, the task has to address the organization and navigation in large sets of search results. The problem of the limited amount of annotated resources in contrast to free text mandates the employment of text mining or entity recognition technologies. A suitable evaluation will need to show, how the approach performs in terms of usability and benchmarks with respect to domain specific question answering.

Open Problem 2: Integration of biomedical entities and resources for semantic search

Based on the strategies developed in *Open Problem 1*, the task is to create a semantic search system for the biomedical domain. The system needs to be able to handle biomedical research tasks and questions using the web as data source. The task includes the selection and implementation of appropriate matching algorithms for the annotation of text with ontology concepts and entity recognition. The goal is to implement such algorithms while maintaining fast response time required by an interactive system. The biomedical domain provides already a rich set of ontologies (Smith et al., 2007) and entity data sources, e.g., protein and gene names from UniProt[1]. The proposed system should enable access to relevant data and resources (e.g., web services) through the identified entities.

Open Problem 3: Semantic search and indexing for heterogeneous data sources

The different research questions require different levels of detail. A system has to be able to use more specialized background knowledge and customized data sources. This includes other relevant sources such as patents or the capability to search and navigate local repositories and intranets. The resulting algorithmic task has to annotate and present larger individual documents or more complex data schemes. Similarly, an adaption for more specialized biomedical ontologies is required. For instance, ontologies may differ in the naming conventions of concepts, thus requiring different annotation algorithms. Both tasks address the portability issue of semantic search solutions.

[1] `http://www.uniprot.org/`

1.2 Thesis Outline

The following thesis is structured in five main sections. The background in Chapter 2 introduces semantic search. This includes sections on data sources, indexing and ranking algorithms, ontological background knowledge, text mining, and entity recognition. Introduced are also the closely related semantic web technologies. A section for Question answering is included. Question answering provides the background and benchmarks for the evaluation of the proposed semantic search systems.

Chapter 3 describes the ideas, algorithms, and implementation of the semantic search system GoWeb. The section provides a discussion on the performance of GoWeb as web-based semantic search system using three benchmarks. The GoWeb system proposes a solution to the *Open Problems 1* and *2*. Chapter 4 introduces semantic browsing as an addition for semantic enabled question answering. This includes a connection to web services as basis for data analysis pipelines and workflows. Semantic browsing, as described in this chapter, addresses the *Open Problem 2*.

Chapter 5 introduces systems that handle data sources other than the normal web for semantic search. This addresses *Open Problem 3*. The presented systems show how a system can process patents, XML documents, literature databases or intranet repositories as data sources. Also more specialized information needs in biomedical sub-domains are discussed. The final Chapter 6 describes a semantic index as an alternative approach to solve the *Open Problem 1*. It is evaluated using a biomedical question answering benchmark.

CHAPTER 2

BACKGROUND

2.1 Semantic Search

Many current search engines are keyword-based. Such search engines retrieve a list of web sites based on a given keyword query and rank those by relevance to a topic. The search results often include for each document the title and a short textual extract containing the keywords.

Semantic technologies promise a next generation of search engines, which will be able to answer questions. Any new search engine or paradigm has to be able to stand its ground against the widely accepted standard of keyword-based search. The goal and proposed advantage of semantic search is to provide better search results. Improvements can be achieved through:

- better ranking strategies, leading to more relevant hits in the top of the search result list.
- reduction of redundant information achieved by grouping.
- increasing coverage to regard different aspects of a search task.
- transparency by explaining results in more detail.
- answering questions in a concise manner.

There are many existing semantic search engines on the web, for instance, Table 2.1 lists and compares 26 engines. There are three main features to classify the search engines: the information source, information extraction approach, and presentation of the results.

Information Source

The information source of semantic search engines can be classified into three sub categories: structured, semi-structured, and unstructured. First, there are structured statements, often provided by RDF repositories. Second, for semi-structured sources, the most common ones are Wikipedia and scientific literature databases in XML. Third, there is unstructured text, as available on the web. Furthermore, the level of the used input differs. For web pages, there is the option to use the whole web page or a short textual extract, called snippet. Similarly, in scientific articles, there is the distinction between full text articles and their summary, called abstract. For more details on the different data sources please see Section 2.4.

In comparison, structured statements provide the most in-depth options for semantic search, as they allow reasoning. With reasoning it is possible to combine several statements into answers for complex questions. However, the amount of such structured data is still small in comparison to the unstructured data, such as text in web pages or scientific publications. Information extraction

ontologies	(1) implicit through RDF, (2) GO, (3) MeSH
text mining	(4) NLP, (5) label extraction, (6) Ontology terminology, (7) biomedical entities, (8) Wikipedia terminology
type of documents	(9) RDF related, (10) web pages, (11) snippets, (12) abstracts, (13) full text
clustering of results	(14) RDF types, (15) extracted categories, (16) textual labels, (17) ontology, (18) answers, (19) query aspects
result type	(20) RDF resource, (21) extracted text, (22) answer, (23) snippet, (24) sentence, (25) full text, (26) cluster, (27) induced ontology, (28) abstract

Semantic Search Engines	structured/ unstructured	ontologies	text mining	number of documents	type of documents	clustering of results	result type	highlighting	scientifically evaluated
Swoogle	rdf	1		$\gg 10^6$	9		20		yes
SWSE	rdf	1		$\gg 10^6$	9	14	20		yes
Sindice	rdf	1		$\gg 10^6$	9		20		yes
Watson	rdf	1		$\gg 10^6$	9		20		yes
Falcons	rdf	1		$\gg 10^6$	9	14	20	yes	yes
CORESE	rdf	1		$\gg 10^6$	9		20		yes
WikiDB	rdf	1		$\gg 10^6$	9		20		
Hakia	txt		4	$\gg 10^9$	10	15	21	yes	yes
START	txt		4	$\gg 10^9$	10		22		yes
Ask.com	txt		4	$\gg 10^9$	10		23		
BrainBoost	txt		4	$\gg 10^9$	10		24	yes	
AnswerBus	txt		4	$\gg 10^9$	10		25	yes	
Cuil	txt		4,8	$\gg 10^9$	10	15	21	yes	
Clusty	txt		5	$\gg 10^9$	10	16	23,26	yes	
Carrot	txt		5	$\gg 10^9$	11	16	23,26	yes	yes
PowerSet	wiki		4,8	$\gg 10^6$	10	15	23,25	yes	
QuAliM	wiki/txt		4,8	$\gg 10^6$	11,10		22		yes
askMedline	xml	3		$\gg 10^6$	12		28		yes
EAGLi	xml	2	4,6	$\gg 10^6$	12	18	22,28	yes	yes
GoPubMed	xml	2,3	6,7,8	$\gg 10^6$	12	17	23,27,28	yes	yes
ClusterMed	xml	3	5	$\gg 10^6$	12	16	26,28	yes	yes
IHop	xml	3	6,7	$\gg 10^6$	12	19	24,28	yes	yes
EBIMed	xml	2,3	6,7	$\gg 10^6$	12	17	24,27	yes	yes
XplorMed	xml	3	5,6	$\gg 10^6$	12	17	21,28	yes	yes
Textpresso	xml	2	6	$\gg 10^6$	13	17	28	yes	yes
Chilibot	xml		7	$\gg 10^6$	12		24	yes	yes

Table 2.1: Comparison of semantic search engines

techniques are required in order to overcome this so-called semantic gap and use unstructured text as source for semantic search. Semi-structured sources offer some facts in a structured way, but they still contain large parts of free text requiring information extraction.

Information Extraction Approach

The second classification feature for semantic search engines is the information extraction approach. There are various techniques used during the extraction step, such as natural language processing, information retrieval, clustering, text mining techniques, and ontologies. The most complex approach to extract facts is natural language processing (Manning and Schütze, 1999). This approach is characterized by a pipeline of different computer linguistics algorithms and techniques to extract relevant entities and relations. A related method is the named entity recognition, in this process known entities, such as person, company, or protein names, are located and classified in text. Similarly, text mining techniques can be used to extract facts from text. An important technique for text mining is the matching of ontology concepts in text, in which the identified ontology concepts constitute the extracted facts. Next to ontologies, there are other terminologies of interest for extraction. For instance, Wikipedia provides a wide range of topics, but not a defined structure or relation between topics, as with ontologies. See Section 2.3 for more details on ontologies.

An alternative to matching and recognition techniques, such as ontology-based text mining, are the clustering methods. The clustering algorithms use only the input text to extract features and group texts with similar features in a cluster.

Presentation

From the user perspective the presentation and different types of search results are another classification criterion. Depending on the index information source, the type of results is often similar. For instance, RDF search engines mostly return links to RDF resources, whereas web search is often centered on text extracts of web sites. Likewise for literature search, the results often only comprise the abstracts. Alternatively, there are answer types that try to summarize the search result (e.g., cluster, extracted texts or sentences, and induced relevant parts of an ontology) or provide a direct answer to a question. A second aspect of the result presentation is the grouping of multiple search hits with common aspects to provide a compact result view.

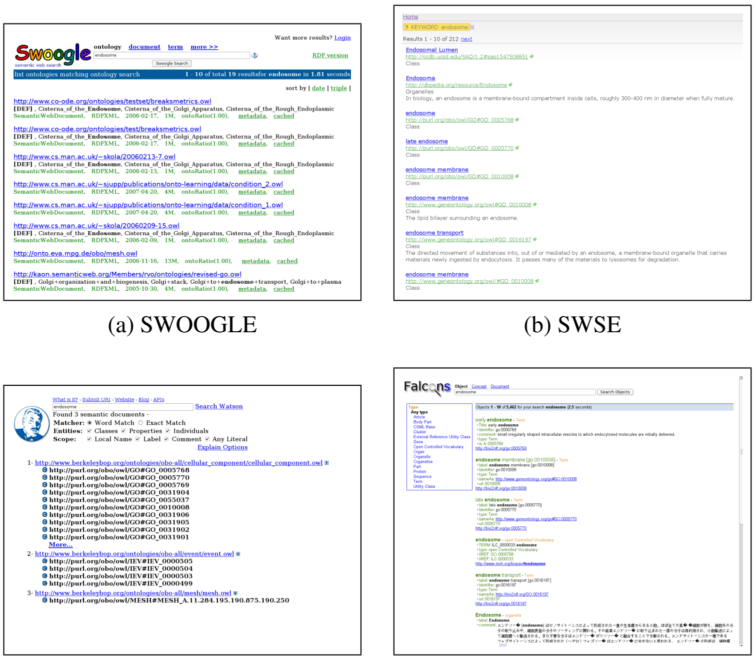
In the following, we give an overview on existing semantic search engines. Based on the data source, the presented search engines are categorized in three classes: RDF, web, and literature. For example systems, the advantages, drawbacks and possible limits of the semantic search approach are discussed.

2.1.1 RDF-based semantic search engines

The semantic web can be a source of structured statements in RDF. In contrast to the web as it is now, the semantic web promotes the use of formal statements and reasoning to deliver advanced services not available on the Web now (Berners-Lee et al., 2001). To facilitate machine-readability and knowledge processing, a set of standards, query languages, and the semantic stack was proposed by the W3C. The stack comprises unique identifiers and XML as common markup language. On top of XML, it defines the Resource Description Framework, RDF to capture subject-predicate-object triples. To model and use RDF, there is the modeling language RDF Schema (RDFS, Brickley et al. 2004) and the query language SPARQL (Prud'hommeaux and Seaborne, 2008).

The Web Ontology Language OWL extends the basic class definitions and triples of RDF at the next level. OWL provides description logic as modeling language and a rule layer (Grigoris and van Harmelen, 2004; Baker and Cheung, 2006). This feature allows the automated inference. For more details on ontologies as background knowledge please see Section 2.3. All of the above standards serve the need to formally represent knowledge and facilitate reasoning over this knowledge. To apply these standards, explicit statements or knowledge is required.

Semantic search engines using existing structured documents such as RDF triple stores comprise Swoogle (Ding et al., 2004), Semantic Web Search Engine (SWSE) (Harth et al., 2006), Semantic Wiki Search (Haase et al., 2009), WikiDB (Clements), Sindice (Tummarello et al., 2007), Watson (d'Aquin et al., 2007), Falcons (Cheng et al., 2008), and the CORESE system (Dieng-Kuntz and Corby, 2005), see Figure 2.1 for example screen shots. The search engines include existing RDF repositories and crawl the internet for formal statements, e.g., OWL files. A search retrieves a list of results with URIs. For SWSE and Falcons the result is enriched with a description and a filtering mechanism for result types. CORESE uses conceptual graphs for matching a query to its databases. WikiDB is slightly different from the others in that it extracts formal knowledge implicit in meta tags of Wikipedia pages and converts it into RDF offering querying with SPARQL. Similar for Semantic Wiki Search, Haase et al. (2009) add semantic search capability to the Semantic MediaWiki (Völkel et al., 2006).



Want more results? Login

endosome

list ontologies matching ontology search 1 - 10 of total 19 resultsfor **endosome** in 1.81 seconds

sort by | date | triple |

http://www.co-ode.org/ontologies/testset/breaksmetrics.owl
[DEF] , Cisterna_of_the_**Endosome**, Cisterna_of_the_Golgi_Apparatus, Cisterna_of_the_Rough_Endoplasmic
SemanticWebDocument, RDFXML, 2006-02-17, 1M, ontoRatio(1.00), metadata, cached

http://www.co-ode.org/ontologies/test/breaksmetrics.owl
[DEF] , Cisterna_of_the_**Endosome**, Cisterna_of_the_Golgi_Apparatus, Cisterna_of_the_Rough_Endoplasmic
SemanticWebDocument, RDFXML, 2006-02-17, 1M, ontoRatio(1.00), metadata, cached

http://www.cs.man.ac.uk/~skola/20060213-7.owl
[DEF] , Cisterna_of_the_**Endosome**, Cisterna_of_the_Golgi_Apparatus, Cisterna_of_the_Rough_Endoplasmic
SemanticWebDocument, RDFXML, 2006-02-13, 1M, ontoRatio(1.00), metadata, cached

http://www.cs.man.ac.uk/~sjupp/publications/onto-learning/data/condition_2.owl
SemanticWebDocument, RDFXML, 2007-04-20, 4M, ontoRatio(1.00), metadata, cached

http://www.cs.man.ac.uk/~sjupp/publications/onto-learning/data/condition_1.owl
SemanticWebDocument, RDFXML, 2007-04-20, 4M, ontoRatio(1.00), metadata, cached

http://www.cs.man.ac.uk/~skola/20060209-15.owl
[DEF] , Cisterna_of_the_**Endosome**, Cisterna_of_the_Golgi_Apparatus, Cisterna_of_the_Rough_Endoplasmic
SemanticWebDocument, RDFXML, 2006-02-09, 1M, ontoRatio(1.00), metadata, cached

http://onto.eva.mpg.de/obo/mesh.owl
SemanticWebDocument, RDFXML, 2006-11-16, 13M, ontoRatio(1.00), metadata, cached

http://kaon.semanticweb.org/Members/rvo/ontologies/revised-go.owl
[DEF] , Golgi+organization+and+biogenesis, Golgi+stack, Golgi+to+**endosome**+transport, Golgi+to+plasma
SemanticWebDocument, RDFXML, 2005-10-30, 4M, ontoRatio(1.00), metadata, cached

(a) SWOOGLE

? KEYWORD endosome

Results 1 - 10 of 212 next

Endosomal Lumen
http://ccdb.ucsd.edu/SAO/1.2#sao1547508851
Class

Endosoma
http://dbpedia.org/resource/Endosome
Organelles
In biology, an endosome is a membrane-bound compartment inside cells, roughly 300-400 nm in diameter when fully mature.

endosome
http://purl.org/obo/owl/GO#GO_0005768
Class

late endosome
http://purl.org/obo/owl/GO#GO_0005770
Class

endosome membrane
http://purl.org/obo/owl/GO#GO_0010008
Class

endosome membrane
http://www.geneontology.org/owl#GO_0010008
Class
The lipid bilayer surrounding an endosome.

endosome transport
http://www.geneontology.org/owl#GO_0016197
Class
The directed movement of substances into, out of or mediated by an endosome, a membrane-bound organelle that carries materials newly ingested by endocytosis. It passes many of the materials to lysosomes for degradation.

endosome membrane
http://www.geneontology.org/owl/#GO_0010008
Class

(b) SWSE

endosome

Found 3 semantic documents -
Matcher: ● Word Match ○ Exact Match
Entities: ☑ Classes ☑ Properties ☑ Individuals
Scope: ☑ Local Name ☑ Label ☑ Comment ☑ Any Literal

1- http://www.berkeleybop.org/ontologies/obo-all/cellular_component/cellular_component.owl
http://purl.org/obo/owl/GO#GO_0005768
http://purl.org/obo/owl/GO#GO_0005770
http://purl.org/obo/owl/GO#GO_0005769
http://purl.org/obo/owl/GO#GO_0031904
http://purl.org/obo/owl/GO#GO_0055037
http://purl.org/obo/owl/GO#GO_0010008
http://purl.org/obo/owl/GO#GO_0031906
http://purl.org/obo/owl/GO#GO_0031905
http://purl.org/obo/owl/GO#GO_0031902
http://purl.org/obo/owl/GO#GO_0031901
More...
2- http://www.berkeleybop.org/ontologies/obo-all/event/event.owl
http://purl.org/obo/owl/IEV#IEV_0000505
http://purl.org/obo/owl/IEV#IEV_0000504
http://purl.org/obo/owl/IEV#IEV_0000503
http://purl.org/obo/owl/IEV#IEV_0000499
3- http://www.berkeleybop.org/ontologies/obo-all/mesh/mesh.owl
http://purl.org/obo/owl/MESH#MESH_A.11.284.195.190.875.190.250

(c) Watson

endosome

Type
Any type
Article
Body Part
CSML Base
Cluster
External Reference Utility Class
Gene
Open Controlled Vocabulary
Organ
Organelle
Organelles
Part
Protein
Sequence
Term
Utility Class

Objects **1** - **10** of **5,462** for your search **endosome** (**2.5** seconds)

early **endosome** - Term
- Title: early **endosome**
- Identifier: go:0005769
- comment: small irregularly shaped intracellular vesicles to which endocytosed molecules are initially delivered.
- type: Term
- is A: 0005768

http://bio2rdf.org/go:0005769

endosome membrane [go:0010008] - Term
- label: **endosome** membrane [go:0010008]
- Identifier: go:0010008
- type: Term
- sameAs: http://www.geneontology.org/go#GO:0010008
- url: 0010008

http://bio2rdf.org/go:0010008

late **endosome** - Term
- label: late **endosome** [go:0005770]
- Identifier: go:0005770
- type: Term
- sameAs: http://www.geneontology.org/go#GO:0005770
- url: 0005770

http://bio2rdf.org/go:0005770

endosome - open Controlled Vocabulary
- TERM: ILC_0000033:**endosome**
- type: open Controlled Vocabulary
- XREF: GO 0005768
- XREF: ILC 0000033

http://www.inoh.org/biopax#**endosome**

endosome transport - Term
- label: **endosome** transport [go:0016197]
- Identifier: go:0016197
- type: Term
- sameAs: http://bio2rdf.org/GO:0016197
- url: 0016197

http://bio2rdf.org/go:0016197

Endosome - organelle
- label: **Endosome**
- comment: エンドソー� (**endosome**) はピノサイトーシスによって形成された一重の生体膜からなる小胞。ほぼ全ての真� �細胞が持ち、細胞外の分子の取り込みや、細胞表面の分子のソーティングに関わる。その結果エンドソー� に取り込まれた一部の分子は再利用され、小胞輸送によって細胞膜へと輸送される。また不要な分子はエンドソー� がリソソー� と融合することで分解される。エンドサイトーシスの一種であるファゴサイトーシスによって形成されれ（ヘテロ）ファゴソー� はエンドソー� に含めないと思われる。エンドソー� の形成は、細胞膜

More

(d) Falcons

Figure 2.1: RDF-based semantic search engines

As mentioned, the above systems are limited by the availability of structured documents, a problem addressed by approaches such as the Semantic MediaWiki (Völkel et al., 2006)[1] and large efforts such as Freebase (Bollacker et al., 2008), which provides an environment to author formal statements. Another effort to create a large scale repository is DBpedia (Bizer et al., 2009). It is an effort to extract structured information from Wikipedia and make this information available on the web, including SPARQL interface. They rely on a pattern-based approach to extract facts from tables in Wikipedia pages. But as users do not need to adhere to a standard in Wikipedia, there are Wikipedia entries, where it is not possible to be able to extract the available information without manual adjustment.

The results of RDF-based semantic search engine are intended to be used as input for reasoning. With reasoning it is possible to answer complex questions, for instance the question: Which football stadiums in Germany have been opened from 2000 to 2010? Starting with a semantic

[1] http://semantic-mediawiki.org

search engine the relevant RDF resources are identified (e.g., in DBpedia) and then a SPARQL query with the time restriction can be used to filter the initial stadium list to the relevant subset to answer the question. Compared to keyword-based search the reasoning approach is more powerful but is limited by the available fact repositories.

The second class of tools next to RDF-based works on unstructured text and therefore does not suffer from this limit. The systems can be distinguished by the document source they work on (Web, Biomedical, Wikipedia), the use of background knowledge in the form of ontologies, the use of text mining techniques such as stemming, concept identification, deep/shallow parsing.

2.1.2 Web-based semantic search engines

There are a number of semantic search engines working on the web. These are Hakia, START (Katz et al., 2006), Ask.com, BrainBoost (Answers.com), AnswerBus (Zheng, 2002), Cuil[2], Clusty[3], and Carrot[4]. Please see Figure 2.2 for example screen shots. Hakia, START and AnswerBus use natural language processing to understand documents. An evaluation of Hakia is presented by Tumer et al. (2009). They conclude that "semantic search performance of search engines was low for both keyword-based search engines and the semantic search engine [Hakia]". Similar Signorini and Imielinski (2009) evaluate Hakia and Cuil. They conclude, that semantic search and text matching needs still to be improved.

The approach of Cuil, Clusty and Carrot is to cluster search results. The systems aim to label clusters with phrases, which are offered as related queries. A detailed example for the query "diabetes causes" in the search engine Clusty is available in Figure 2.3. The example illustrates, that the clusters and labels contain relevant phrases, such as "Insulin", but the labels and proposed hierarchy are not always relevant and sound, as in the example with "Risk, Diagnosis", "Prevention", and "Risk factors". These topics are closely related but are separated and even on different levels in the result hierarchy created by Clusty. This is the main limitation of this approach, as clustering cannot recreate correct semantics and relations from text snippets alone.

Cuil, Clusty and Carrot are not semantic search engines in a strict sense, since these phrases are not part of an ontology or vocabulary. However, they do have the advantage of being generally applicable and Cuil offers definitions for phrases where available.

Ask.com uses its ExpertRank, an algorithm for computing query-specific communities and ranking in real-time, to identify relevant pages (Yang, 2006). They include structured knowledge to generate answers. BrainBoost is a meta-search engine. It uses the proprietary AnswerRank algorithm applying machine learning and natural language processing. It ranks answers extracted from the top web sites.

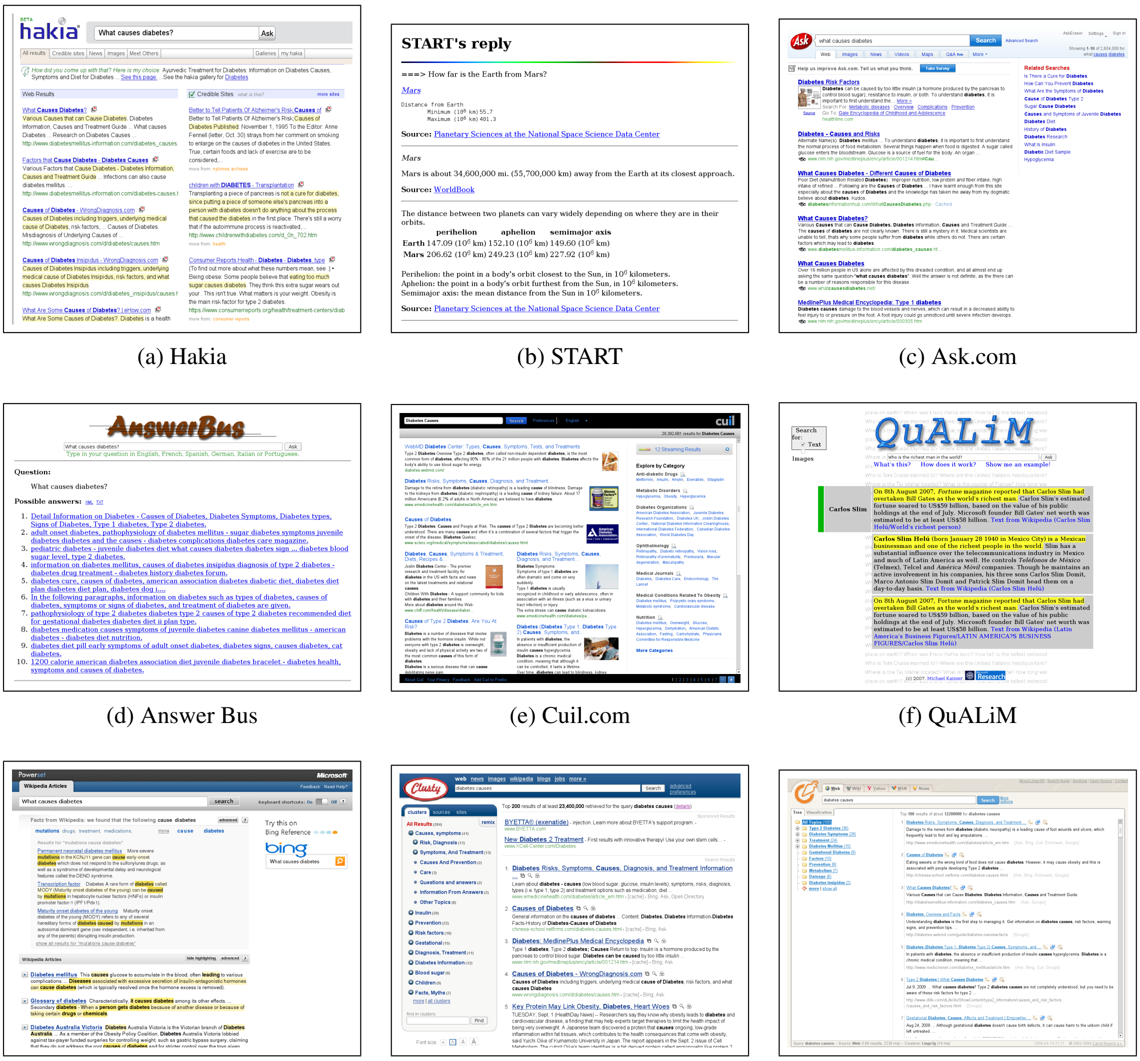
Wikipedia is a valuable resource to answer questions and hence some engines are specifically applied to it. PowerSet applies e.g., natural language processing to Wiki in a similar manner as Hakia. QuAliM (Kaisser, 2008) uses a pattern-based approach for sentence analysis. Semantic type checking for answers and a fallback mechanism with web search is implemented in QuAliM.

The advantage of natural language processing, as employed by PowerSet, is the chance to extract a rich set of facts from a given text. This advantage is achieved by using complex algorithm pipelines. These pipelines contain often CPU intensive steps, which make natural language processing a time consuming task. A second drawback is the systematic error accumulation in the different stages of the pipeline. As shown in Figure 2.4, PowerSet identifies for the query "What causes diabetes" relevant sentences, which talk in detail about the mutations in a specific gene. However, PowerSet identifies also irrelevant sentences talking about "Diabetes Australia" which is in this result an organization and not the disease.

[2] http://www.cuil.com/

[3] http://clusty.com/

[4] http://www.carrot-search.com/

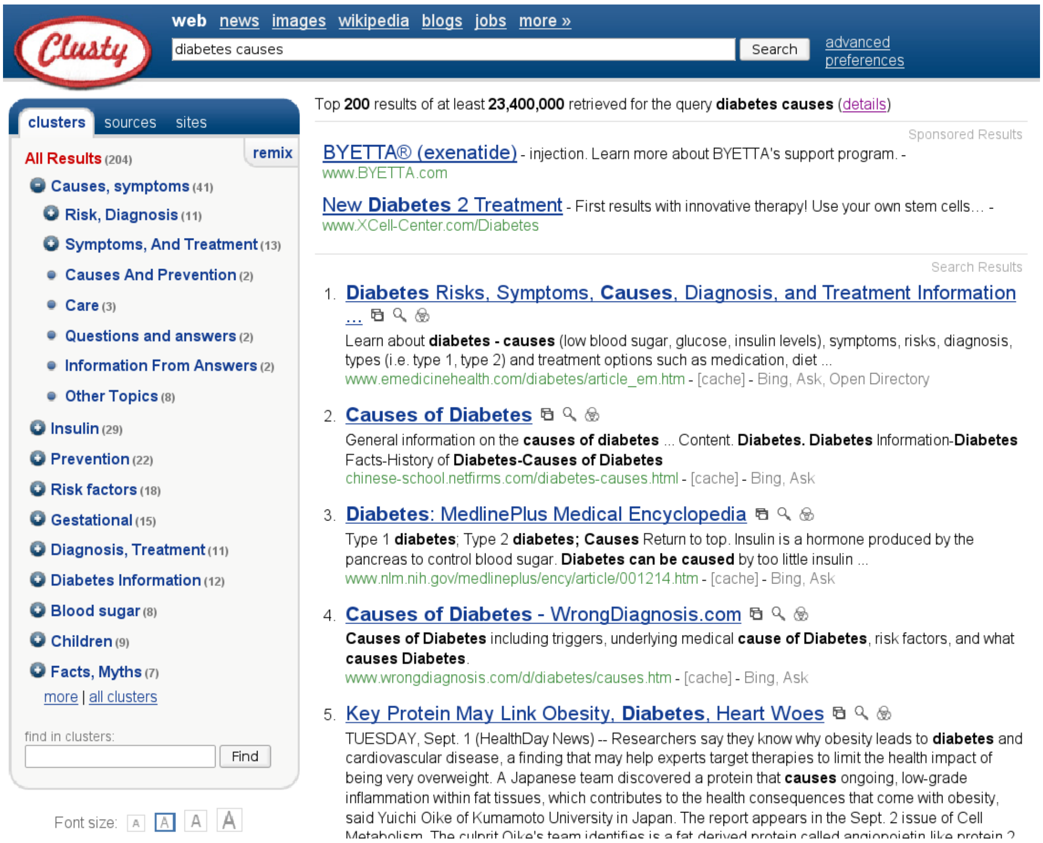


(a) Hakia (b) START (c) Ask.com

(d) Answer Bus (e) Cuil.com (f) QuALiM

(g) PowerSet (h) Clusty (i) Carrot

Figure 2.2: Web-based semantic search engines



Top **200** results of at least **23,400,000** retrieved for the query **diabetes causes** (details)

BYETTA® (exenatide) - injection. Learn more about BYETTA's support program. -
www.BYETTA.com

New Diabetes 2 Treatment - First results with innovative therapy! Use your own stem cells… -
www.XCell-Center.com/Diabetes

1. **Diabetes** Risks, Symptoms, **Causes**, Diagnosis, and Treatment Information ...
Learn about **diabetes - causes** (low blood sugar, glucose, insulin levels), symptoms, risks, diagnosis, types (i.e. type 1, type 2) and treatment options such as medication, diet ...
www.emedicinehealth.com/diabetes/article_em.htm - [cache] - Bing, Ask, Open Directory

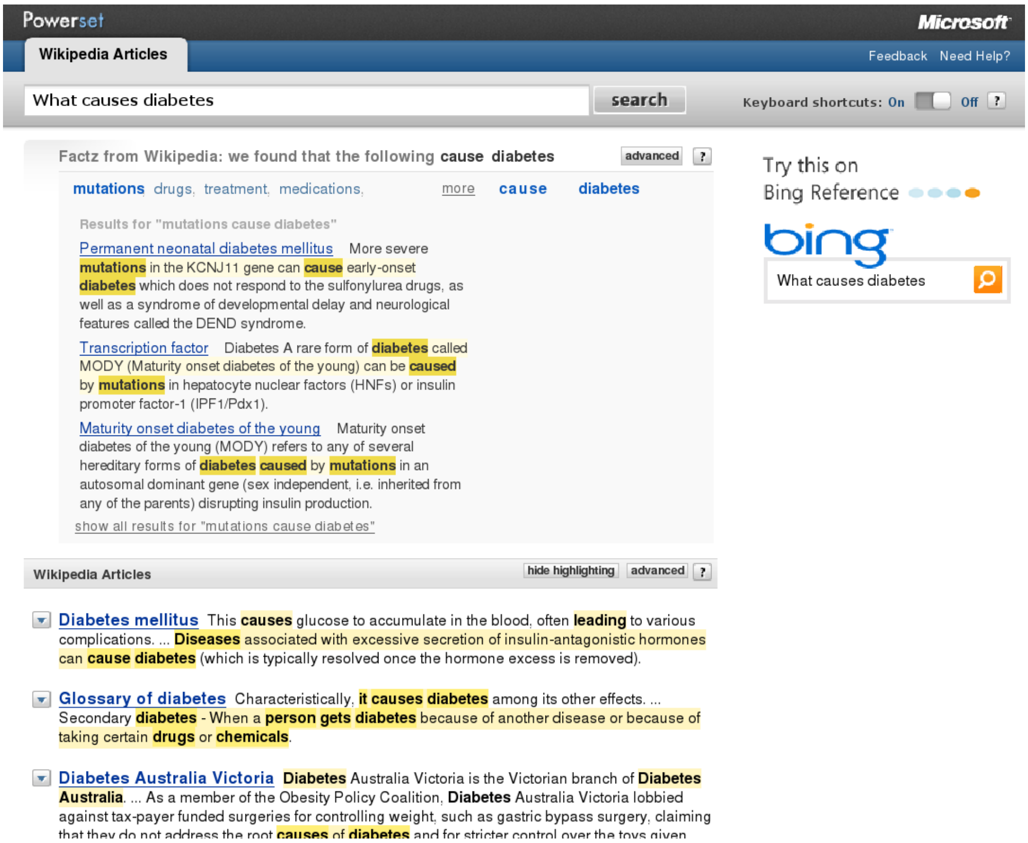
2. **Causes of Diabetes**
General information on the **causes of diabetes** ... Content. **Diabetes. Diabetes** Information-**Diabetes** Facts-History of **Diabetes-Causes of Diabetes**
chinese-school.netfirms.com/diabetes-causes.html - [cache] - Bing, Ask

3. **Diabetes**: MedlinePlus Medical Encyclopedia
Type 1 **diabetes**; Type 2 **diabetes; Causes** Return to top. Insulin is a hormone produced by the pancreas to control blood sugar. **Diabetes can be caused** by too little insulin ...
www.nlm.nih.gov/medlineplus/ency/article/001214.htm - [cache] - Bing, Ask

4. **Causes of Diabetes** - WrongDiagnosis.com
Causes of Diabetes including triggers, underlying medical **cause of Diabetes**, risk factors, and what **causes Diabetes**.
www.wrongdiagnosis.com/d/diabetes/causes.htm - [cache] - Bing, Ask

5. Key Protein May Link Obesity, **Diabetes**, Heart Woes
TUESDAY, Sept. 1 (HealthDay News) -- Researchers say they know why obesity leads to **diabetes** and cardiovascular disease, a finding that may help experts target therapies to limit the health impact of being very overweight. A Japanese team discovered a protein that **causes** ongoing, low-grade inflammation within fat tissues, which contributes to the health consequences that come with obesity, said Yuichi Oike of Kumamoto University in Japan. The report appears in the Sept. 2 issue of Cell Metabolism. The culprit Oike's team identifies is a fat derived protein called angiopoietin like protein 2

Figure 2.3: Screen shot for Clusty with example query “diabetes causes”



Factz from Wikipedia: we found that the following **cause diabetes**

Results for "mutations cause diabetes"

Permanent neonatal diabetes mellitus More severe **mutations** in the KCNJ11 gene can **cause** early-onset **diabetes** which does not respond to the sulfonylurea drugs, as well as a syndrome of developmental delay and neurological features called the DEND syndrome.

Transcription factor Diabetes A rare form of **diabetes** called MODY (Maturity onset diabetes of the young) can be **caused** by **mutations** in hepatocyte nuclear factors (HNFs) or insulin promoter factor-1 (IPF1/Pdx1).

Maturity onset diabetes of the young Maturity onset diabetes of the young (MODY) refers to any of several hereditary forms of **diabetes caused** by **mutations** in an autosomal dominant gene (sex independent, i.e. inherited from any of the parents) disrupting insulin production.

show all results for "mutations cause diabetes"

Diabetes mellitus This **causes** glucose to accumulate in the blood, often **leading** to various complications. ... **Diseases** associated with excessive secretion of insulin-antagonistic hormones can **cause diabetes** (which is typically resolved once the hormone excess is removed).

Glossary of diabetes Characteristically, **it causes diabetes** among its other effects. ... Secondary **diabetes** - When a **person gets diabetes** because of another disease or because of taking certain **drugs** or **chemicals**.

Diabetes Australia Victoria **Diabetes** Australia Victoria is the Victorian branch of **Diabetes Australia**. ... As a member of the Obesity Policy Coalition, **Diabetes** Australia Victoria lobbied against tax-payer funded surgeries for controlling weight, such as gastric bypass surgery, claiming that they do not address the root **causes** of **diabetes** and for stricter control over the toys given

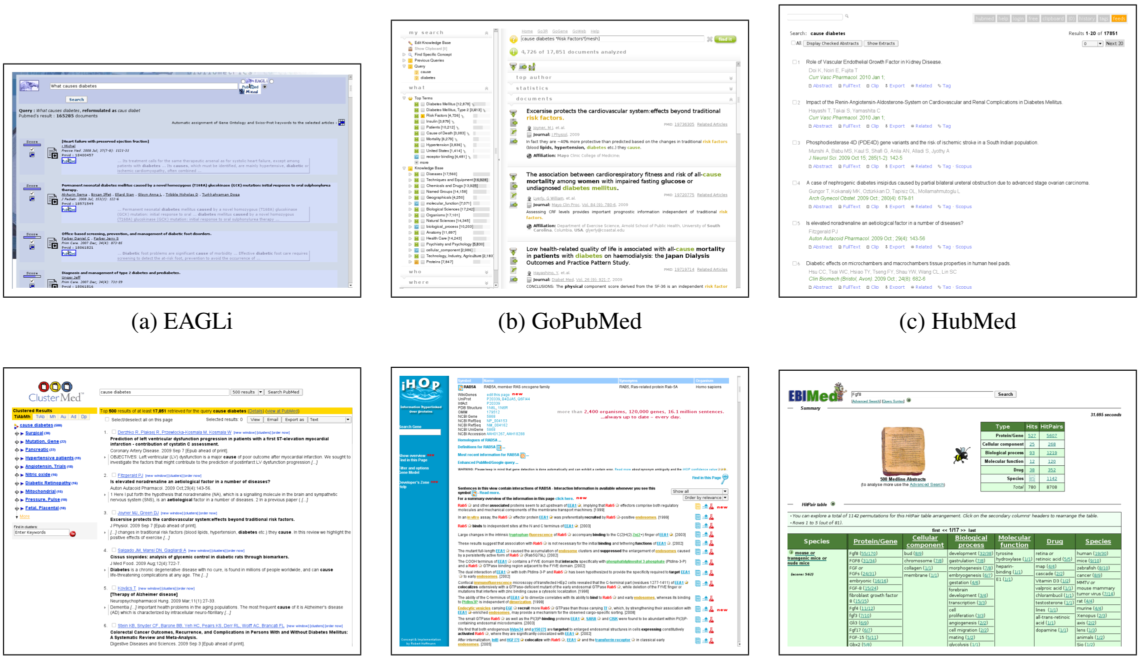
Figure 2.4: Screen shot for PowerSet with example query “What causes diabetes”

The above mentioned tools are intended to be general and as a result they generally do not well cover the biomedical domain. Searching, for example, for a protein such as Fgf8, PowerSet and Hakia do not offer an answer for relevant model organisms. They offer information on the protein, but are not able to find zebra fish as a model organism.

2.1.3 Literature-based semantic search engines

Engines such as askMedline, EAGLi (Gobeill et al., 2007), GoPubMed, HubMed, ClusterMed, IHOP, EBIMed , XplorMed, Textpresso and Chilibot address specialized biomedical domain by processing biomedical literature in full text (Textpresso) or abstracts as available in the PubMed literature database. See Figure 2.5 for a selection of screen shots. Due to the focused domain and literature database, these engines can employ background knowledge and more sophisticated query mechanisms. For instance, EAGLi and askMedline process natural language questions as input for the search.

An important approach for semantic search in the PubMed database is the employment of ontologies to classify, search for, and filter articles. The algorithms to match the ontology concepts in an article use text mining techniques. The advantage of using ontologies is that they offer a structured model with relations and definitions, as opposed to clustering methods. In comparison to RDF and structured statements, they still offer a limited capability for reasoning, e.g., induction of subsumption hierarchies. Ontology-based text mining uses algorithms, which can be less CPU-intensive than natural language processing. However, the drawback is that ontologies are normally not designed for text mining, thus the labels and synonyms of concept may not appear literally in text, see also Section 2.5 for more details. Furthermore, only the concepts modelled in the ontology can be matched. A problem for all approaches working on text are ambiguous textual labels, for details on disambiguation see Section 2.5.1.



(a) EAGLi

(b) GoPubMed

(c) HubMed

(d) ClusterMed

(e) IHOP

(f) EBIMed

Figure 2.5: Literature-based semantic search engines

GoPubMed
The semantic search engine GoPubMed supports answering biomedical questions (Doms and Schroeder, 2005). GoPubMed retrieves PubMed abstracts for your search query, detects ontology concepts from GO and MeSH and allows the user to browse the search results by exploring the ontologies and displaying only papers mentioning selected terms, their synonyms or descendants. It includes information about authors, journal, dates and locations. Furthermore, it provides statistical analysis for ontological concepts and all PubMed documents and individual search results.

HubMed
The search engine HubMed (Eaton, 2006) is direct front end to PubMed. It offers tools for the citation management of found PubMed articles. It also provides options for expanding the query or clusters the results in categories. This is all based on the MeSH terms directly provided by PubMed. If there are no MeSH concepts available for an article, these features do not work, because no term matching is done by HubMed itself. This is usually the case for the more recent articles. As an alternative, they offer a tagging system where you can add your own tags to an article.

Vivisimo ClusterMed
ClusterMed does not use existing ontologies, but clusters documents hierarchically, although it distinguishes between categories like title and abstract, authors, affiliation, or publication date. From document clusters, it derives representative terms. This automated hierarchy generation inevitable merges concepts of different nature, as the algorithm is only guided by the given documents, thus missing a lot of background knowledge a human uses in the creation of an ontology. Since Vivisimo clusters documents on the fly there is a limit to its scalability.

iHOP
The tool iHOP (Good et al., 2006) uses genes and proteins as hyperlinks between sentences and abstracts. It converts the information in PubMed into one navigable resource. The navigation along the gene network allows for a stepwise and controlled exploration of the information space. Each step through the network produces information about one single gene and its interactions.

EBIMed
The search engine EBIMed (Rebholz-Schuhmann et al., 2007) identifies associations between proteins/genes, cellular components, biological processes, molecular functions, drugs and species. The textual evidence in form of supporting sentences is cited. The frequency is used to sort the relations and providing the user with a quick overview of a search result set of PubMed documents.

XplorMed
(Perez-Iratxeta et al., 2007) filters PubMed results by the eight main MeSH categories and then extracts topic keywords and their co-occurrences. Abstracts can be retrieved for co-occurring keywords. The topic keywords are single words, usually occurring with a high frequency. Thus multi word concepts such as “Stem Cell” are not proposed as keyword. Currently XplorMed has a limited scalability and searches are restricted to 500 documents.

Textpresso for C. elegans
(Müller et al., 2004) has been developed as part of the Wormbase effort. It offers currently about 100 concepts such as allele, anatomy, association, characterization, clone, comparison, consort, developmental stage, disease, drugs, effect, entity feature, gene, involvement, life stages, mutants,

nucleic acid, organism, pathway, phenotype, purpose, regulation, reporter gene, restriction enzyme, sex, spatial relation, strain, time relation, transgene, transposon, vector and including also a subset of GeneOntology concepts. It searches only abstracts and full text articles relevant for C. elegans. Textpresso does not offer an ontology tree for the exploration of a result set.

Chilibot
With Chilibot (Chen and Sharp, 2004) it is possible to search PubMed abstracts for specific relationships between proteins, genes, or keywords. The user enters a list of two or more genes or other keywords. It uses a sentence-based approach to create a relationship graph to visualize different types of relationships, such as stimulation or inhibition. The generation of new hypotheses is possible based on this network and the guilt-by-association principle.

2.1.4 Summary

Currently, there are already many existing semantic search systems. They include semantics on different levels. For instance, the support and usage of ontologies differs with respect to the levels of expressiveness. Literature search engines comprise a lot of semantic knowledge compared with other search engines. This is possible due to the corpus. It is completely known and in form of a structured database, allowing to employ more sophisticated approaches. The most lightweight systems, such as Clusty or Carrot, employ only a minimal amount of background knowledge or ontologies. They use clustering algorithms to build the structured search result. An overview of all mentioned systems and discussed features is presented in Table 2.1.

From the user perspective, the overview shows that the current RDF search engines only differ in the presentation and limited number of indexed RDF statements. The web-based and Wikipedia-based systems using natural language processing, such as Hakia or PowerSet, offer no support for ontologies and reasoning. The literature-based systems provide such advanced semantic features, but are limited to their database.

RDF based semantic search can employ reasoning for question answering and hence deal with very complex queries (e.g. DBpedia can answer the query "All soccer players, who played as goalkeeper for a club that has a stadium with more than 40.000 seats and who are born in a country with more than 10 million inhabitants"). However, to date there is little RDF-based structured knowledge available in comparison to free text. At the same time all structured approaches suffer from the problem of the "unreasonable effectiveness of data" (Halevy et al., 2009). If the data is large enough it will cover all possible queries. Put simply, if the above DBpedia query is of so much interested, someone will put it together with the answer into a freetext document, so that an approach without RDF and reasoning can find the answer, too.

Besides the RDF-based approach, there are three main other approaches, NLP, ontologies, and clustering. They differ regarding three aspects: processing time, general applicability, and inference capabilities. NLP-based approaches require a lot of processing time, but are domain independent and general applicable. They only support inference in a limited form. Ontology-based approaches require little processing time, but are domain dependent and difficult to generalize. There key advantage is the ability to draw inferences. Clustering-based approach require little to medium processing time, are domain independent, but do not support inference well.

In comparison of all systems, there is no other semantic search system that combines general web search, biomedical relevant entities (protein and gene names) and ontologies, such as GeneOntology or MeSH. We will develop such an engine in Chapter 3.

2.2 Question Answering

Evaluating semantic search, Questions Answering (QA) can provide a challenging task and benchmarks. The form of questions can range from simple retrieval tasks to more complex scenarios. The most challenging task of Questions Answering is to provide a concise answer to a natural language question. This differs from the document retrieval (e.g., traditional web search) as now sentences or facts and not the relevant documents are the expected result.

The first well-known systems for question answering were presented in the early 1960's. There was for instance the "BASEBALL"-system (Green et al., 1963). It was an expert system for questions related to the sport baseball. In 1965 a review lists 15 systems (Simmons, 1965), which handle natural language input to various data back-ends (databases or texts from encyclopedias). An update (Simmons, 1970) with the second generation systems categorizes the systems in the following five categories: conversation machines, fact-retrieval systems, mathematical word-problem processors, natural language text processing and miscellaneous (e.g., usage of triplets extracted from a large document collection combined with logic). With the intensive research and theoretical foundations of computer linguistics during 1970's and 1980's more complex QA systems have been developed, such as the Berkeley Unix Consultant (Wilensky et al., 1989). Since the late 1990's, the field and interpretation of the questions answering is driven by competitions in the context of large conferences, such as the Text REtrieval Conference (TREC) QA track or commercial success. With TREC-QA the technologies have been focused for generating short answers to factual questions (e.g., "Who is the president of the United States?"). The introduction of the internet as information resource has shaped this process and introduced new challenges and turned the most systems from closed to open-world question answering system. For instance, the new data provided new challenges with regards to the amount or imprecision.

The general architectures of a question answering machine uses techniques from several fields including natural language processing, information retrieval, information extraction, template matching, summarization, and natural language generation. These techniques may be applied in different stages of the answering process, for instance during the pre-processing of the data or the evaluation of the user query.

According to Andrenucci and Sneiders (2005) there are three main approaches for question answering: Natural Language Processing, Information Retrieval and Template-based Question Answering. The later applies pattern matching to databases or web resources. The intelligence of the system is the set of rules and question templates. The second approach is based on Information Retrieval (IR). Traditional the task of IR is to find relevant documents for a given query. With the development towards passage retrieval the usage as question answering system becomes feasible. Both approaches may benefit from a large corpus. These corpora present information in a redundant fashion. This effect may be used to simplify the query strategies and provide the option for re-ranking.

The first option is based on classical Natural Language Processing. NLP is strongly related to computational linguistics. For such systems it is common to convert an input text to a formal representation. This representation can be logic-based, a semantic network, conceptual diagrams or frame-based. This conversion or computer-based understanding is usually at least a two step process. Starting with a parser a parse tree is generated using rules and lexicons. This parse tree is then transformed in the intermediate format with a semantic interpreter. The interpreter tries to assign meaning to the parse tree. It can employ semantic rules and a world model (e.g., ontology). Given the input, which is now associated with a meaning a query for the knowledge base is created and executed. The result from this query is then often transformed again to natural language and then presented to the user. This class of systems relies on a pre-structured database.

With the WWW as important information source, unstructured text has become the main focus as main data source. For handling large collections of texts "shallow NLP" has been established. It moves from the understanding (semantic analysis) of text in traditional NLP to extracting text chunks for matching patterns or entities. These text chunks are then presented as answers. For example take the question: "Who received the Nobel price in medicine in 2008? The "who" indicates a search for a person entity. A sentence containing such an entity and the association with the keywords "Nobel price", "medicine", "2008" would be considered as a valid answer (Srihari and Li, 1999).

Both the IR and NLP-based approaches are often used in competitions for Question Answering. The most prominent one for English questions is the TREC QA track[5]. It started in 1999 (TREC-8) and ended with TREC 2007. For non-English, there is the NTCIR Project[6]. The project ran QA in the competitions NTCIR-3 (2002), NTCIR-4 (2004) and NTCIR-5 (2005). Although it is mainly Japanese, they also provide English translation for the questions.

In the following, key aspects of question answering are detailed. The questions, as input for questions answering, are typed using different classification schemes. Also the influence of question, with respect to the expected answer, is presented. Next the strategies are presented, on how to implement a question answering system. Thereafter, the TREC QA as key competition is presented in detail.

2.2.1 Types of Questions and Answers

Pomerantz (2005) provide a quite comprehensive review for the different classification systems for questions and possible expected answers. They use five question types of taxonomies. These are called as follows (1) *Wh*-words, (2) Subjects of questions, (3) The functions of expected answers to questions, (4) the forms of expected answers to questions, and (5) types of sources from which answers may be drawn.

"The Five W's" is a simple and common classification of questions in the English language. However, questions are not necessarily phrased using a *wh*-word and statements phrased using a *wh*-word are not necessarily questions.

There are many classification schemas, which organize entities according to their subjects. This idea can also been applied to classify questions using existing and special purpose classification schemas.

Other question taxonomies use the answer to the question as classifier. The idea is that it's important to understand what type of information is sought for. For instance for the question "How high is the Mont Blanc?" a numerical answer or more precisely a measurement quantity is expected. Graesser et al. (1994) developed a theoretical model of question asking behavior. Their taxonomy is additionally divided into classes that require short versus long answers.

The forms of expected answers and types of sources from which answers may be drawn can be seen as complementary. They handle two aspects for the task of answering a question: (1) the exact definition on what the expected answer is and (2) where to find the answer.

Beside this formal approach of classifying into one of the five proposed categories, there exist special purpose or mixed variants. For instance, Li and Roth (2006) define a two-layered taxonomy. The proposed hierarchy contains 6 coarse classes and 50 refined subclasses.

Another special purpose taxonomy is proposed by Ely et al. (2000). They provide a taxonomy of generic clinical questions. It consists of 69 generic questions types, classified into a four level hierarchy. The first level splits the questions according to diagnosis, treatment, management, epidemiology and non-clinical questions. Their work is based on 1101 questions about patient care and involved 103 randomly selected family doctors from the state Iowa (United States of America) for classifying the questions.

As a last example for the many possible options for the classification of questions there is a classification by Tomuro and Lytinen (2001). It defines 12 question types:

- DEF – definition – What does "reactivity" of emissions mean?

[5] `http://trec.nist.gov/data/qamain.html`
[6] `http://research.nii.ac.jp/ntcir/index-en.html`

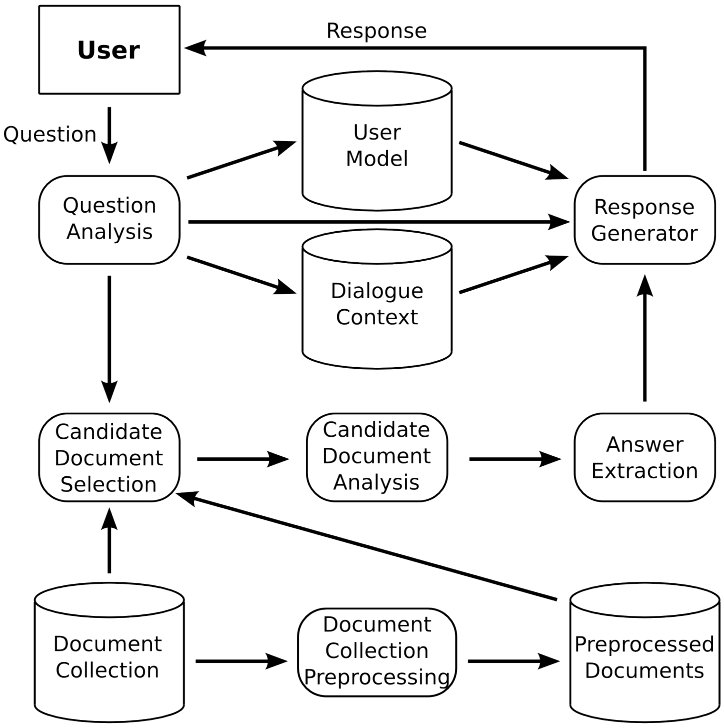


Figure 2.6: Generic Architecture for a Question Answering System (Hirschman and Gaizauskas, 2001)

- REF – reference – What do mutual funds invest in?
- TME – time – What dates are important when investing in mutual funds?
- LOC – location
- ENT – entity – Who invented Octane Ratings?
- RSN – reason – Why does the Moon always show the same face to the Earth?
- PRC – procedure – How can I get rid of a caffeine habit?
- MNR – manner – How did the solar system form?
- DEG – degree
- ATR – atrans – Where can I get British tea in the United States?
- INT – interval – When will the sun die?
- YNQ – yes/no – Is the Moon moving away from the Earth?

The classification of questions is a basic task to select the best answering strategy in a question answering system. It is part of the question analysis and answer extraction as described in the following section.

2.2.2 Answering Strategies

Hirschman and Gaizauskas (2001) provide a generalized architecture for many Question Answering systems (Figure 2.6). They derived this architecture from the system competing in TREC QA track, but it can be also applied for other Questions Answering applications. The system consists of six parts:

1. *Question Analysis* – The input from the user has to be interpreted or parsed into the internal format. This may include the dialogue context in form of previous questions or other user specific information (user model). The classification of questions into question types (see Section 2.2.1) is employed to provide the system an expected answer type. This could for instance be done with a machine learning approach (Li and Roth, 2006).

2. *Document Collection Preprocessing* – Documents or semi-structured data may be pre-processed to improve the handling as a knowledge source. This may involve time-consuming algorithms using possible advanced NLP, entity recognition or semantic annotation. The preprocessing is often required to provide an interactive/real-time behavior for the question answering system.

3. *Candidate Document Selection* – Even after preprocessing it is often not feasible to process all documents for a given question. Thus a selection takes place to create a relevant subset. This may involve Information Retrieval methods, for example Boolean queries or fuzzy queries with relevance ranks. Additionally a shortening of documents to relevant sections or passages may take place.

4. *Candidate Document Analysis* – Optional step, which analyzes the candidates and provides information for the answer extraction in the next step. This may include for instance additional entity recognition, sentence or noun-phrase tagging, multi-word term recognition and dependency analysis or relationship extraction.

5. *Answer Extraction* – Selection of the best matches for the question. This may include for instance the keywords, expected answer type, synonyms or semantic relations. Such constraints vary from system to system.

6. *Response Generator* – Depending on the application this may range from a simple list of short answers, or sentence snippets with links to the original documents or fully developed interactive answer with options for exploring the answers and evidence and justifications with links to related answers and the inclusion of feedback options to the system.

This architecture is well suited for systems using semantic annotation with ontologies or Natural Language Processing techniques. This is true for the light-weight approaches and the more in-depth linguist methods of NLP. Such systems are often quite complex and required tedious configuration and training of components to achieve good results.

With the growth of the Web in size and also importance, it became apparent that question answering on the web might require a different approach. This resulted in systems such as MULDER (Kwok et al., 2001) and AskMSR (Brill et al., 2001). AskMSR, for instance, relies only on the snippets from existing web search results and does not employ any ontological resources or named-entity recognizers. This machine explores the feature, that the Web contains the information in a redundant fashion. Thus this new approach has been dubbed redundancy-based approach. In later evaluation it was shown, that this approach is very well suited for factoid-type questions as they were used in the TREC-QA competition. A more recent version of AskMSR called Aranea has been implemented by Lin (2007). The authors also discuss the limits of this approach. They consider the philosophy “data is all that matters” as introduced by Banko and Brill (2001) and its implications on the algorithms. Given enough data simple counting may perform the same as any machine learning applied. However this approach is limited by the assumption the most popular answer is the correct answer. This assumption is not always true. After benchmarking their system they reformulate the statement to: “the more data, the better, but not too much garbage.”

The data driven approach to question answering has also been recognized to be an option for the inclusion in normal search engines. For instance a Google employee proposed to create a large repository of facts. These facts should be distilled off-line in a preprocessing step using a lightweight extraction method. They demonstrate this idea with temporal entities and regular expressions for finding these entities (Paşca, 2007). They only store small textual "nuggets" of information.

The methods presented so far have been mainly applied to open domains. The alternative is question answering in restricted domains. From the historical perspective most early QA systems started with restricted domains. This was usually due to the technical and theoretical limitations of the NLP and computational resources. Today there are also fields suited for restricted domain question answering system as proposed by (Mollá and Vicedo, 2007) For example such fields are:

- Interfaces to machine-readable technical manuals
- Front-ends to knowledge sources
- Help desk systems in large organizations

The main difference between open and restricted domain is the existence of specific information resources for the restricted domain QA. These information resources, which may include specific ontologies and databases, are used for the knowledge base. If a sufficient database exist, the QA can be focused on natural language interfaces to databases, see Androutsopoulos et al. (1995) for more details. The combination of free-text and ontologies has been demonstrated by Zajac (2001). They employ semantic annotation of the question and the source text.

Interesting for restricted domains is also the combination of heterogeneous sources of information. For instance Lin (2002) proposes to combine databases and internet-based sources. If no answer is available in the selected databases, an additional step with a corpus of documents is introduced. Also the combination of different databases is no simple task. In general there are two options: the federated approach and the integrative approach. Both have been studied in the context of federated search and data-warehousing for data bases. Independent of the approach, the goal of restricted domain QA is to provide the option for answering more complex questions (Ceusters et al., 2003). A typical field for domain specific systems with potentially complex questions is the medical area. For instance there is the MedQA system (Lee et al., 2006). But for such domains with vast expert knowledge, there are also alternative approaches to question answering. This can be Expert Systems or in the medical context Clinical Decision Support Systems. These systems may employ Case-Based-Reasoning, Bayesian Networks, Neural Networks, Rule-Based Systems, Logical Conditions or Causal Probabilistic Networks.

One aspect of Question Answering is that an answer may not exist or it is not available in the used corpora and other data sources. For instance, consider questions posed by physicians and a fixed biomedical corpus. Ideally a question filtering should determine whether or not a posed question is answerable. Yu and Sable (2005) developed an approach to classify clinical questions. They use supervised machine-learning and a corpus of 200 clinical questions that have been annotated by physicians to be answerable or unanswerable. They report that their best system is a probabilistic indexing system that achieves 80.5% accuracy.

Next to the dedicated Question Answering systems, the modern internet search engines have begun to integrate Question Answering into their engine. For instance Google and MSN/Bing include the option to pose questions directly in the search field. A review by Roussinov et al. (2008) sees them on a good path for question answering, but the specialized internet-based question answering system such as AskJeeves, BrainBoost, AnswerBus are still a bit better.

Finally, besides all of the automated approaches, Google, Yahoo! and Microsoft use or have used humans to answer questions in their services Google Answers (now retired), Yahoo! Answers[7] and MSN Live Search QnA (retired).

[7] http://answers.yahoo.com/

2.2.3 Competitions

To assess the proposed question answering approaches and implemented systems benchmarks are needed. These benchmarks are often part of competitions, challenges or conference series. The most well known conference series in this context is the Text REtrieval Conference (TREC)[8]. It is organized by the National Institute of Standards and Technology (NIST)[9], an agency of the U.S. Department of Commerce. Additionally the Defense Advanced Research Projects Agency (DARPA, historically also called ARPA), a military-based research agency has been a co-sponsor. The conference series started in 1992 and in 2009 it has its 18^{th} installment. It currently has three main tracks:

- **Chemical IR Track**: large scale evaluations on chemistry data sets in order to promote the research in chemical IR in general and chemical patent IR in particular. Test collection composed of over 100,000 full text chemical patents and 45,000 research papers from the Royal Society of Chemistry, UK (Lupu et al., 2009).

- **Entity Track**: The overall aim of this new track is to perform entity-related search on Web data. These search tasks (such as finding entities and properties of entities) address common information needs that are not that well modeled as ad hoc document search (Balog et al., 2009).

- **Web Track**: The TREC Web Track explores and evaluates Web retrieval technologies. Currently, the Web Track uses a new billion-page ClueWeb09 collection. It includes both a traditional ad hoc retrieval task and a new diversity task. (Clarke et al., 2009)

Next to the currently running tracks, there are tracks that have been discontinued. There is for instance the Genomics Track. The purpose of the Genomics track was to study retrieval tasks in the specific domain of Genomics data. This domain is constructed broadly to include not just gene sequences but also supporting documentation such as research papers, lab reports, etc. The Genomics track last ran in TREC 2007. Other past TREC tasks are:

- Enterprise Track: enterprise search

- HARD Track: High Accuracy Retrieval of Documents (last TREC 2005).

- Interactive Track: user interaction with text retrieval systems (last TREC 2003)

- Novelty Track: find new non-redundant information (last TREC 2004)

- Question Answering Track

- Robust Retrieval Track (last TREC 2005)

- SPAM Track (last TREC 2007)

- Terabyte Track: scaling of methods for larger data sets (last TREC 2006)

Depending on the research focus there are further corpora. With a focus on cross language question answering and document retrieval, there is the Cross-Language Evaluation Forum (CLEF)[10] or the NTCIR Project[11]. The main bottle neck in providing corpora and running competitions are the time and funding requirements. For instance the TREC Genomics Track was not discontinued because the problem is solved, but because at the time there was no further funding available.

[8] http://trec.nist.gov/

[9] http://www.nist.gov/

[10] http://www.clef-campaign.org/

[11] http://research.nii.ac.jp/ntcir/

The different competitions, especially TREC, provide benchmarks for the evaluation of question answering systems. Furthermore, the corpora, questions, and expected answers of these benchmarks can also be used as an evaluation tool for a semantic search. Next, we will describe ontologies, as they are an important concept for the implementation of semantic search. Ontologies can be used to model and share the concepts and relations of the background knowledge.

2.3 Ontological Background Knowledge

A fundamental aspect for the work of researchers is the need to share knowledge. In the beginning this was often done without the help of a controlled vocabulary or nomenclature. This is in particular applicable for the biomedical area and life sciences. There are many genes and proteins that have multiple names or identifier. An example is Hnrpa1 which is also known as Tis, Fli-2, heterogeneous nuclear ribonucleoprotein A1, helix-destabilizing protein, single-strand-binding protein, hnRNP core protein A1, HDP-1, and topoisomerase-inhibitor suppressed. But there are also names such as Cleopatra, Ariadne, Groucho, Lost in Space, Brokenheart, Hairy, Superman and many more. Of course there have also been efforts to standardize names or at least to reach a consensus for naming. For instance in the context of yeast research and for human genes there are widely used standards, even if they are not always adhered to in literature.

Similar issues arise, if the task is to annotate genes and their function within the categories biomedical process, molecular function, and cellular components. You can find that

- Cellulose 1,4-beta-cellobiosidase is also known as exoglucanase,
- superoxide-generating NADPH oxidase as cytochrome B-245,
- thiamin as vitamin B1,
- pyrexia as fever,
- heme as haem, and
- Apoptosis as cell death.

The aim of ontologies is to reduce this problem. They include concepts, synonyms and their relation ships.

One prominent example for a widely used ontology is the GeneOntology (Ashburner et al., 2000). In the beginning it was developed for the annotation of the fruit fly genome. Later the GeneOntology was adapted and expanded for mouse and other genomes and covers now biomedical processes, molecular functions, and cellular components. It uses two kinds of relationships to model the dependencies between the concepts: `is-a` and `part-of`. Both play a pivotal role in biomedical ontologies (Schulz et al., 2006). Today the GeneOntology is also part of the Open Biomedical Ontology (OBO) effort (Smith et al., 2007), which houses over 60 ontologies covering many areas of interests. This includes anatomy, chemical compounds, development, experimental conditions, phenotype, taxonomy and more.

The second example are the Medical Subject Headings (MeSH). The MeSH thesaurus is developed by the U.S. National Library of Medicine (NLM). Its main purpose is to provide an index for the articles, books and other media in the National Library of Medicine. It tries to cover all relevant topics for the medical area. This includes disease and anatomy, but also other branches like geographic locations and experimental techniques.

There are other medical ontologies, e.g., GALEN (Rector et al., 2003), SNOMED and Unified Medical Language System (UMLS) (Humphreys et al., 1998; Bodenreider, 2004). An overview of all presented Ontologies is available in Table 2.2. The Unified Medical Language System (UMLS) has a different approach. It tries to integrate as much relevant ontologies or taxonomies as possible. The UMLS consists of three parts: a meta thesaurus, a semantic network and the specialist lexicon. Whereas the meta thesaurus represents the concepts including the synonyms, the semantic network corresponds to categories and the specialist lexicon acts as a kind of index.

geneontology.org	Ontology with $\geq$20.000 terms on biomedical processes, molecular functions and cellular component
nlm.nih.gov/mesh	Medical Subject Headings created by the U.S. National Library of Medicine, taxonomy with $\geq$150.000 terms
opengalen.org	formal medical ontology, with $\geq$70.000 terms
snomed.org	commercial medical ontology, which contains $\geq$350.000 terms
nlm.nih.gov/research/umls/	Unified Medical Language System created by the U.S. National Library of Medicine, contains $\geq$1.000.000 terms
obofoundry.org	Open Biomedical Ontology, collection of over 60 specialized biomedical ontologies

Table 2.2: Ontologies for the biomedical field

A non-trivial aspect is the design and later on the evolution of ontologies. With many thousands concepts and definitions how does one keep it all, including the relations, consistent. Although this starts with the question: How is consistence defined in the first place? The GeneOntology follows an informal approach. The transitive closure still has to hold. This means, if a concept A `is-a` B and B `is-a` C then A `is-a` C has to be true. These inferred redundant relationships are not kept directly in the ontology. This helps to ease the maintenance of the ontology as corrections, modifications and additions only need to check if their direct relations are still valid.

Even though this consistency definition is a pragmatic solution, there are more formal approaches. One such idea is the usage of description logic to formally define concepts and their relations. This was used for instance in the GALEN and SNOMED ontologies. The advantage of the formal definitions is the chance to automatically check for inconsistencies in the ontology. Imagine that one adds the new fact heparin `is-a` glycosaminoglycan, but it was not yet stated that heparin biosynthesis `is-a` glycosaminoglycan biosynthesis. Because of the formally defined relations and concepts, this additional relation can be inferred with this new fact in the knowledge base.

All ontologies represent a view on selected aspects of the world. During the modeling process for an ontology, the knowledge engineer has to decide what and in which detail to include in the ontology. This often includes abstraction or simplifications. Depending on the goals and modeling premises of an ontology, similar facts may be modeled in different ways.

Different goals and premises may be the reason to build a custom ontology. A typical scenario is to model selected aspects in greater detail than in existing ontologies. But it has to be considered that building an ontology is a time consuming process. In a study by Studer and Sure (2006) the average size of an ontology was 830 entities and an average duration of 5.3 person months to create the ontology. It was also stated that the reuse of existing entities was in average around 50%.

The re-usage of existing ontologies is advised, but brings also new tasks to handle. If two ontologies cover a similar aspect the corresponding concepts should be matched. The ontology matching, alignment or integration has to employ background knowledge and approximation, as stated by Harmelen (2006). This area is an active field of research with different proposed approaches such as corpus-based (Wächter et al., 2006), graph-based (Thanh Le and Dieng-Kuntz, 2007) or correspondence patterns and an alignment ontology (Scharffe et al., 2008).

Ontologies are an important tool to share knowledge. They formalize a model, including concepts, synonyms, and their relations, for a specific domain, they are a specification of a conceptualization (Gruber, 1993). If modelled appropriately, ontologies provide specialist knowledge and the capability to reason with the concepts using the defined relationships. These features allow to use ontologies as background knowledge for semantic search.

2.4 Data and Text Sources

The basis for semantic search and question answering is the information source. The information can be provided in many different formats and ways. This varies from the peer-reviewed articles with supplementary data published in a printed journal to a highly specialized database with a search front-end or from open access repositories to content provided on personal web pages. The accessible content format may range from simple text over structured and parsable files to binary content, e.g., PDF-documents, images or movies.

Research interest in the biomedical domain is highly diverse. It is cross-disciplinary and comprises interests from basic research to drug development for a patient. This diversity leads to a wide range of potential relevant information sources. This may be for instance:

- electronic publications from the web with the latest research and findings
  - literature databases such as MEDLINE[12], PubMed Central (PMC), Scopus, Web of Science
  - open access publishing, e.g., PLOS[13], BioMedCentral (BMC)
  - pre-print servers
  - conference proceedings
- specialized bio databases for different topics, like
  - genome databases
    - general, e.g., Entrez Gene, Ensembl, EMBL Nucleotide Sequence Database, Genome Reviews
    - specific, e.g., organism specific like Saccharomyces Genome Database (SGD), WormBase or FlyBase, Mutations (Sequence variation database project) or Parasites (Parasite Genome databases), OMIM (human genotypes related phenotypes)
  - proteins UniProt (Swiss-Prot/TrEMBL), InterPro
  - 3D protein structures like the PDB
  - pathways and networks e.g., KEGG, IntAct, Reactome
  - and many databases more
  - chemical compounds and drugs: PubChem, Rote Liste, Drugs@FDA
- patents provided by the patent offices (e.g., USPTO, EPO) or other service like freepatentsonline.com
- institutions and their provided service, like the National Center for Biotechnology Information (NCBI), European Bioinformatics Institute (EBI) or European Molecular Biology Laboratory (EMBL)
- semi-informal source like Wikipedia or discussion forums

[12] Medical Literature Analysis and Retrieval System Online

[13] Public Library of Science

- local data repositories

In this rich and diverse bouquet of information source, it is often hard to keep track of changes. For databases, there is for instance the Nucleic Acids Research (NAR) journal with its reoccurring special issues on databases; the NAR volume 36 contains 183 articles about new or updated databases. In contrast 19% of the databases are no longer available under their published URL (Wren, 2008). Usually the databases offer their own user interface, import, export, or data format.

Another aspect for the different information sources is the quality and trust in the provided information. For the many important bio-databases, like the PDB, there are either handled by government-funded organizations or large research consortia. They often use a review process to ensure a certain quality standard of the data. Smaller and specialized databases typically describe their approach in peer-reviewed paper. The paper enables the scientist to evaluate, if the used methods to create the database fit their needs. For scientific publications the peer-review process is currently the standard procedure. This includes the publishing companies, open access journals, and scientific conferences and their proceedings. For high-key journals and events with many submissions the acceptance rate can be very low and only very mature manuscripts are accepted.

Next to the high-key journals, there exist many other journals. Literature databases combine the publications of several sources. One of the benefits of such databases is the capability to search the content and not each journal individually.

Due to the subscription-based business model of the content provides many of the databases and journals only offer the title, abstract and authors for a publication. This lead may lead to the situation, that one can not retrieve the full text of your own publication without additional costs. An alternative is the idea of open access publishing. There the costs of publishing are paid by the author and it is guarantied that the publication is always freely available. One of the motivations for open access is also, that these papers can be easily accessed and hopefully are more cited.

The advantage of free access has been recognized by some funding agents. For instance the NIH Public Access Policy ensures that the public has access to the published results of NIH funded research. They requires that scientists submit their final peer-reviewed journal manuscripts that arise from NIH funds to the digital archive PubMed Central.

An exception to review process is the pre-print server. On such servers you can find manuscripts that still have to be peer-reviewed. The advantage is the faster access of new ideas. With the pre-print access the time for the review, which takes weeks, months, sometimes even years, is saved. A popular example is the arXiv[14] pre-print server for the field of physics. In fact it is so common, that some physics journals even accept the arXiv e-print numbers for the submission of publications to the peer-review process. The Nature Publishing Group started in 2007 Nature Precedings[15]. Its goal is to be a free electronic repository for pre-publication research and preliminary findings, especially for biology, medicine, chemistry and earth science.

An attempted hybrid approach of combining peer-review with early and open access is followed by the PLoS ONE Journal[16]. It proclaims, that it will publish all papers that are technically sound, which is checked via peer-review. A judgment about the importance is done post publishing by the readership.

Wiki systems, where the users collaboratively add, modify or delete content, became increasingly popular. They offer a simple way to share knowledge. The most popular is Wikipedia. It provides with its ≈ 3.4 million English entries a rich source of diverse information. Because of the open nature of wikis and in particular Wikipedia, there are some drawbacks. It may be biased, contain inconsistencies or be incomplete (Clauson et al., 2008). For citing as source of information in a scientific article or using it as reliable source of information, it is rated second best (Stillman-Lowe, 2008; Lacovara, 2008).

[14] http://arxiv.org/

[15] http://precedings.nature.com/

[16] http://www.plosone.org

The source discussed so far are all more or less available in the web. But often information is also stored locally. This can range in the simplest case from a bunch of files on the local hard disk to intranets with wiki or other knowledge management systems storing papers, presentations and data series. Local copies of some important databases are not uncommon. This may be done for convenience in terms of speed or response time or for security and confidentiality concerns.

When dealing with an information system all of these sources have at least to be considered. The decision of using or excluding may depend on several aspects, but often there are time and money as the limiting factors.

2.4.1 Document and Content Types

Information, including the web or patents, can be provided in various formats. This can range from text-related formats, binary-files to database schemas and dumps. It can be multi-medial, which can include texts and images as more traditional media or audio and video. The last media types became increasing popular since the bandwidth of the connection available to the individual user developed from some 56kBit per second with a modem to several MBit per second with technologies such as DSL[17]/ADSL[18], Data Over Cable, Fiber-to-Home or similar. The basis for the web is still the HyperText Markup Language (HTML). This markup language supports structural markup, like headings, links, etc. Today's web browser support many extensions and supplements for HTML. This includes for instance embedded videos or script languages like JavaScript. In the HTML5 Standard[19] there are now tags to handle audio and video content without additional plug-ins such as the Adobe Flash Player provided by Adobe Systems[20].

In the area of the scientific research and publishing the vast bulk of content is provided in form of texts with images (publications) and databases. Most of the databases offer a front end. This front end renders database content into a textual representation, often HTML with help of tables or div-tags to structure the text. The publications themselves can be obtained usually as HTML or Portable Document Format (PDF) files. Other formats to consider are the different file-formats used in popular word processing and desktop office applications. Many of such application use or have used in the past binary and proprietary formats to store the content. With the requirement of portability, long term stability and interoperability, open standards for document-files have emerged (Shah et al., 2008).

The most prominent example is the Open Document Format for Office Applications (OpenDocument, ODF, ISO/IEC 26300:2006) and the Office Open XML (also referred to as OOXML, OpenXML, or Open XML, ISO/IEC 29500:2008). With both formats the storage of office documents has moved from binary to XML. The OpenDocument format is the result of a long process of standardization of multiple vendors. It was originally developed by Sun; the standard was developed in the Organization for the Advancement of Structured Information Standards (OASIS) consortium. Its most prominent usage is the OpenOffice suit and variants such as StarOffice.

The OpenXML format was developed by Microsoft (market leader for office products) as replacement for the old binary formats in Microsoft Office 2003. Later on it was submitted as standard to the Ecma International (ECMA-376, December 2006) and was recognized by the International Organization for Standardization (ISO) as standard (December 2008). The discussion, whether the OpenXML is a good open standard, is still ongoing. Some discussion points are that the standard is just too big. It has a specification of over 7000 pages. As result, it is possible that only Microsoft can provide a full reference implementation. Furthermore, the possible infringement of Microsoft patents with OpenXML is unclear. The compatibility with Open Source licenses such as GPL is currently in question. On the technical site, for instance, the extensibility

[17] Digital Subscriber Line

[18] Asymmetric Digital Subscriber Line

[19] `http://www.w3.org/TR/html5/`

[20] `http://www.adobe.com/products/flashplayer/`

and usage in other office programs is limited. The global flags defined in OpenXML are currently designed with older Microsoft-Office formats in mind.

The Portable Document Format (PDF) has established itself since the mid 1990-ties as the standard for exchanging mostly read only documents. Originally a proprietary format, PDF was released as open standard in July 2008. The PDF format is a container format for encapsulating the fixed layout of a document. That includes text, fonts, images and vector graphics. The format combines a PostScript subset with font-embedding and storage structures with data compression. A file in PDF cannot guarantee long-term reproducibility. For this use case the *ISO 19005: PDF/A* standard has been introduced. PDF/A is a subset of the Adobe PDF Reference 1.4. It excludes for example transparency, sound and movie actions. PDF/A also requires some optional elements to be implemented, e.g., embedded fonts. PDF/A-1a (Level A Conformance) denotes full compliance with the currently approved PDF/A Standard ISO 19005-1: Part 1. This entails, for instance, the preservation of a document's logical structure and content text stream in natural reading order (Tagged PDF). This is used for text-extraction and in practical applications, such as screen readers.

The correct identification of the content and file type and, if applicable, character encoding is the basis for text extraction. In an ideal case the text extraction should also recover the logical structure of the input. Title, headings or summaries such as abstracts should be identified. This additional knowledge can help to rank the content for importance. Especially in scientific publications and abstracts contain a condensed version of the content.

2.4.2 Web search

Most of the relevant information for current research is nowadays available in the web and may be retrieved in an electronic form. To find the information keyword search is currently the de facto standard. For web search the most used sites are Google, Yahoo! and Microsoft. With the market leader Google for web search, the verb to google has established itself as a synonym for keyword search on the web.

The main purpose of keyword-based search is the fast identification of relevant documents in the web. For this the user chooses keywords, submits it as query to the engine. Today the usual result format is the paginated result list, with 10 or a similar number of results per page. Each result document is presented with the title, a short extract of the document containing the keywords (snippet) and a URL. Which documents are presented as most relevant to a query depends on the ranking algorithm employed be the search engine (see also section 2.6). If the user is not satisfied with the results, the user changes the keywords and searches again. This process may continue till the user deems to have found the relevant documents.

Because of the popularity, the keyword-based search has been extended for other media types such as images, audio or videos. Often this search options also use textual feature. This features consist mostly of the descriptions found around the media, tags and meta-data (e.g., ID3-tags[21] in MP3 audio files). The content-based search of images (McDonald and Tait, 2003; Liu et al., 2005; Datta et al., 2006, 2008), videos (Amir et al., 2003; Shao et al., 2008) is an open research topic.

A different approach for retrieving relevant web sites is a web directory. Such directories offer links to other web sites and use categories and sub-categories to sort and filter their link database. These catalogs can be either maintained automatically by submissions or by recommendations of users. There might be a review/editing process for ensuring quality and scope aims. The scope of web directories can range from general to highly specialized listings, e.g., for business and e-commerce. A directory implementation can range from a link list on a private homepage up to Open Directory Project with more than 4.6 million entries. Further examples for directories are available in Table 2.3. Meta-search engines are an alternative to single search engines or directories. Meta-search engines use several sources to search. They accumulate the results for the user to one result set. Examples for meta-search can be found in Table 2.3.

[21] http://www.id3.org/

type	name	URL
directories	Open Directory Project	`http://www.dmoz.org/`
	World Wide Web Virtual Library (VLIB)	`http://vlib.org/`
	Yahoo! Directory	`http://dir.yahoo.com/`
	vfunk	`http://www.vfunk.com/`
meta-search	Mamma	`http://www.mamma.com/`
	Ixquick	`http://www.ixquick.com/`
	SurfWax	`http://www.surfwax.com/`
	Agent55	`http://agent55.com/`
	windseek	`http://www.windseek.com/`
	Dogpile	`http://www.dogpile.com/`

Table 2.3: Examples for web directories and meta-search engines

Today's search market is a diverse field. Although the majority uses still Google there exist many alternative and complementing systems. There are vertical search engines, with specialization in nearly any field, e.g., news, blogs, geographic locations and maps. Any medical/biomedical, scientific search system or portal can be seen as a vertical search system. Many systems do not just offer a simple search. They offer additional content and features to attract and help users. Such features include spell checking, auto-completion, or keyword proposal for the refinement of a search.

2.4.3 Automated Content Retrieval

The ability to retrieve the content of a source is the basis for working with the data. The retrieved data may be used, directly for evaluations, e.g., data processing pipelines. Alternatively, the retrieved data can be the basis for building an index in the search infrastructure. The most straight forward technique to retrieve data is to fetch the content via a transfer protocol, e.g., HTTP or FTP and work with a local copy. For databases this could be a simple plain-text database dump or other exported formats. If this is not available, other retrieval methods are required. For linked content, a strategy for retrieving the content is to follow the links to explore the document space. The automated process of traversing linked content and fetching related content is called crawling. The program for this task is called a crawler or spider. The most common scenario for a crawler is the web.

2.4.3.1 Web Crawling

The Goal of a crawler is to create a content copy of a resource. The copy should be as complete and as recent as possible. The word crawling illustrates the idea behind this approach. It follows the links from resource to resource and stops to analyze the content for new links. It crawls the trail of links. The main area of application is the web crawl. A web crawler fetches web sites using the HTTP protocol. The algorithm for a generic web crawler may consist of the following steps:

```
REQUIRE: set of seed URLs
  initialize link database
  prioritize URLs
  create a fetch queue
  REPEAT
    fetch the URLs
    extract the content and URLs
    update link database
    add new URLs to fetch queue,
 UNTIL (empty fetch queue OR time limit OR space limit)
```

Although the algorithmic idea is fairly simple, there are a lot of practical issues to consider. This includes implementation issues but also runtime related problems (Castillo, 2004, 2005). For example there are network, DNS, HTTP, content and web server related issues. The network connection is foundation for retrieval of content. For smaller crawling efforts a normal 10 MBit connection can be sufficient. For large scale efforts a 10 GBit connection can be easily saturated. In an estimation (Hawking, 2006a) the web had a size of 400 TeraBytes. Including some overhead, the data for a full crawl requires more than 10 days to transfer using a 10 GBit link.

Another step for fetching content is the resolving of host names to IP addresses. If the local DNS server fails or is temporarily unavailable due to a high load, web sites might be falsely marked as not existent. The DNS implementation must be able to recover from malformed and wrong records. Similar the HTTP implementation has to cope with web servers not handling accept headers or ranges requests. Web servers may send response with missing or malformed headers or wrong dates. A server might be temporarily unavailable, so requests have to be retried several times.

Politeness and Legal Matters

Politeness, for a crawler is required. This ranges from providing a proper user-agent, and contact information (E-Mail) in the request to adherence of standards, like the Robots Exclusion Protocol. The “robots.txt” is a descriptor for crawlers. It is defined in the HTML 4.01 specification, Appendix B.4.1[22]. With the robots.txt it is possible to allow or restrict the access to a web site. Another way, to specify that a crawler should ignore content, are special HTML meta-tags. These tags define for instance the caching strategy. Also the number of request to a server should be monitored. Too many requests may slow down the answer or crash server, but it may also be interpreted as an attack, such as a denial-of-service attack. Non-compliance to such standard may lead to consequences such as blocking your IP-addresses or in worst cases to legal measures. If damage was inflicted, for example, a down-time in a web shop, it may be coupled with a claim for indemnification.

When dealing with content from the web the legal situation is usually not straight forward. The intellectual property laws for instance, such as copy rights for images or publication and usage licenses, may vary from country to country. For image search it is custom to provide a short preview images, a thumbnail. Depending on the laws this practice can be okay, e.g., through the fair-use practices know to the U.S. trademark law. But the same practice might be an infringement. For instance, there was a non-final court decision in Germany that Google must get permission from the owner or creator of an art work before displaying it. The OLG Hamburg found, that thumbnails are reproductions (OLG Hamburg 26.09.2008 - Az.: 308 O 248/07).

A similar picture can be found regarding illegal content. This can affect content related to political (propaganda), sexual (protection of minors), crime (identity or credit card data), software (warez, movies, MP3) areas. For instance Google did maintain a different index for China, to adhere to the local laws.

[22] http://www.w3.org/TR/html4/appendix/notes.html#h-B.4.1.1

Speed

If each HTTP request takes one second to complete – some will take much longer or fail to respond at all – a crawler can fetch 86,400 pages per day. At this rate, it would take 634 years to crawl 20 billion pages. Thus crawling is done in parallel using hundreds or thousand crawling machines. Additionally each crawling machine employs a high degree of internal parallelism, by using hundreds or thousands of threads issuing HTTP requests and waiting for the responses.

One idea to reduce the crawl size and duration is to crawl only a relevant subset. This lead to the idea of focused crawling (Chakrabarti et al., 1999) and incomplete crawling. One rational for incomplete crawling is that, in both theory and practice, a crawler needs to download just a few levels. Usually no more than 3 to 5 clicks away from the start page are sufficient, if the crawler wants to reach 90% of the pages that users actually visits (Baeza-Yates and Castillo, 2004). Also there are techniques to evaluate the importance of a whole web site by using sample web pages (Gonzlez et al., 2006).

Focused crawling is an idea for implementing vertical search engines. At best the crawler will only visit relevant pages. Some examples are language or region specific crawls, crawling for digital libraries (Bergmark et al., 2002) or topical crawling (Menczer et al., 2004). Topical crawlers rely on machine-learning or other classification techniques to evaluate the importance of web site (Pant and Srinivasan, 2005). Depending on the importance the crawler decides if the links extracted should be also crawled (Ehrig and Maedche, 2003).

Distributed Crawling

As mentioned before crawling has to be done multi-threaded and distributed on more than one machine. The partitioning in parallel executable work chunks is obviously done via the URLs. Each machine can write the fetched data locally. In a post processing step the individual content is merged into a big result set.

This pattern is also known as the Map-Reduce-Pattern. It was made popular by a publication from Google (Dean and Ghemawat, 2004, 2008). They explain how their map-reduce-framework and algorithmic scaffold proved to be highly versatile and easy to work with.

The computation takes a set of input key/value pairs, and produces a set of output key/value pairs. The user expresses the computation as two functions: map and reduce.

$$\begin{array}{lrcl} \text{map} & \texttt{(k1,v1)} & \rightarrow & \texttt{list(k2,v2)} \\ \text{reduce} & \texttt{(k2,list(v2))} & \rightarrow & \texttt{list(v2)} \end{array}$$

Map, written by the user, takes an input pair and produces a set of intermediate key/value pairs. A map-reduce-framework groups together all intermediate values associated with the same intermediate key I and passes them to the reduce function. The reduce function, also written by the user, accepts an intermediate key I and a set of values for that key. It merges these values together to form a possibly smaller set of values. For the calculation the input data is segmented, e.g., via a hash of the key `k1`. Each data chunk is processed by a worker and each worker writes the intermediate data in a local file. For load balancing, there are more chunks than worker. A master schedules the working packages and coordinates the execution. Each chunk can be computed independently, so the required synchronization is the status of a package and which workers are ready. The same goes for the reduce step. The workers, usually less then in the first step, access a chunk of intermediate data and write their own result file. For an illustration please see Figure 2.7.

To deal with failures of hardware and software the master checks if the workers are alive and responsive. If this is not the case, the master restarts the failed jobs. To guarantee the integrity of the data the framework has to provide atomic commits for results. The worker state can be saved with a check point concept.

To facilitate the easy transmission of data files between distributed (e.g., multiple hosts) workers a distributed file system is often part of the framework. The access patterns for files in a map reduce environment are always very similar. File reads and file writes are streamed, not random

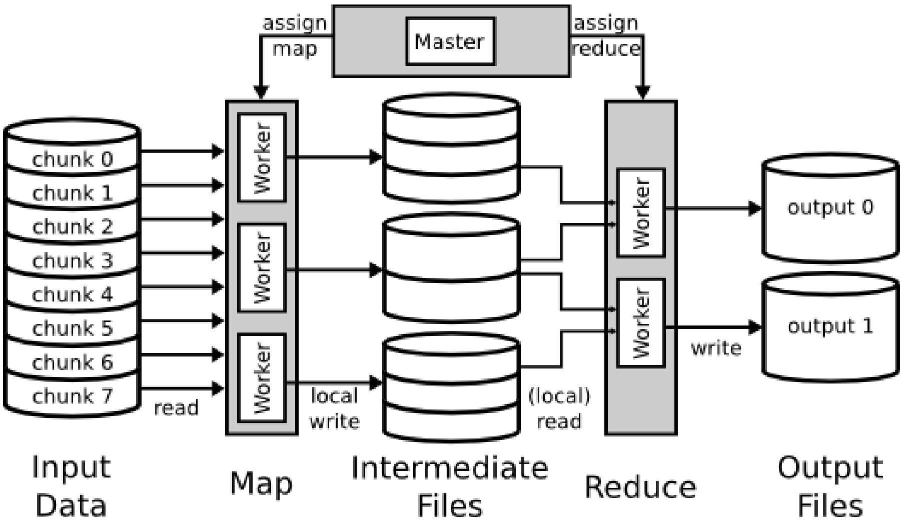


Figure 2.7: MapReduce – Schematic overview

access. Files are only written once and provide only append operations. Thus,provide the file system implementation can rely on the local file systems implementations. The only need is to add the streaming capability to different hosts and a directory where files are located. To add failure resistance the directory and the data should be replicated. For practical reasons files are handled in blocks. The directory handles the association of blocks and replicated blocks to files. As a single point of failure, the directory must be as robust as possible. The use of transactions for changes and fail-over concept with replicated states is nearly mandatory.

An important performance optimization is the principle of locality. The goal is to place the worker as close to the data as possible. This means, for instance, create a reduce worker on the same host, where the majority of the data blocks of intermediate data file is on local disk. This minimizes the access time of files with the minimal amount of network usage.

Many of the mentioned requirements and enhancements are intended for an environment with many hosts, e.g., clusters with more than 1000 machines, where host failures are frequent. This is especially the case for clusters using of-the-shelf hardware. Google's programmers find their MapReduce system easy to use: more than ten thousand MapReduce programs have been created. An average of one hundred thousand MapReduce jobs are executed on Google's clusters every day, processing a total of more than twenty PetaBytes of data per day (Dean and Ghemawat, 2008).

Map-Reduce Examples A good and simple demonstration example program for map-reduce is a distributed word or token count. The inputs for the examples are files containing the texts. When using a map-reduce frameworks the user could write code similar to the following pseudo code:

```
map(String key, String value)
  // key: file name
  // value: text from file
  Iterator i = tokenize(value);
  for each token t from i
     EmitIntermediate(t, 1);
```

```
reduce(String key, Iterator values)
  // key: a token
  // values: a list of counts
  int result = 0;
  for each c from values
     result += c;
  Emit(result);
```

The `map` function extracts the tokens from the text. It then emits for each token a value pair <token, count>. In this simple case the count is constant of one per token. The framework sorts and groups the intermediate keys. The `reduce` function sums up all the counts for a token.

Other examples for map-reduce are as follows (Dean and Ghemawat, 2004):

Distributed Grep The map function emits a line if it matches a given pattern. The reduce function is an identity function that just copies the supplied intermediate data to the output.

Count of URL Access Frequency The map function processes logs of web page requests and outputs <URL, 1>. The reduce function adds together all values for the same URL and emits a <URL, total count> pair.

Reverse Web-Link Graph The map function outputs <target, source> pairs for each link to a target URL found in a page named "source". The reduce function concatenates the list of all source URLs associated with a given target URL and emits the pair: <target, list(source)>.

Term-Vector per Host A term vector summarizes the most important words that occur in a document or a set of documents as a list of <word, frequency> pairs. The map function emits a <hostname, term vector> pair for each input document (the hostname is extracted from the URL of the document). The reduce function is passed all per-document term vectors for a given host. It adds these term vectors together, throwing away infrequent terms, and then emits a final <hostname, term vector> pair.

Inverted Index The map function parses each document, and emits a sequence of <word, document ID> pairs. The reduce function accepts all pairs for a given word, sorts the corresponding document IDs and emits a <word, list(document ID)> pair. The set of all output pairs forms a simple inverted index. It is easy to augment this computation to keep track of word positions.

The success of Map-Reduce has lead to open source implementations like Hadoop[23] or Phoenix (Ranger et al., 2007). The open source search engine Nutch[24] and its distributed crawler implementation (Moreira et al., 2007) use the Hadoop framework.

Although Map-Reduce is popular, there are other systems for the fetching of URLs, like a distributed queue. For instance Marin et al. (2008) propose high performance priority queues for parallel crawling.

2.4.3.2 Web Services

An alternative for crawling the content are web services (WS). They facilitate the communication from program-to-program (Gottschalk et al., 2002). The advantage for content retrieval is the option to skip the step of parsing data files intended for human readers (eyeball web).

For the definition of web service, there is a wide range of possibilities. They range from generic and all-inclusive to specific and restrictive. For instance, there is the strict definition of the W3C. They define a web service as follows[25]: “A Web service is a software system designed to

[23] `http://hadoop.apache.org/core/`

[24] `http://nutch.apache.org/`

[25] `http://www.w3.org/TR/ws-gloss/#webservice`

support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP-messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards."

A simpler definition just assumes that Web services are Web application programming interfaces (API) that can be accessed over a network (e.g., the Internet). The service call is executed on the remote system hosting the requested services. In the simplest case a web site, URL or CGI-script request could be seen as a simple web service call.

A Web service definition can encompass many different systems, but in common usage the term refers to clients and servers that communicate over the HTTP protocol used on the Web. Currently the services encompassed by those criteria can be classified in two categories: "Big" Web Services and RESTful Web Services.

"Big" Web Services

This class of web services follows the W3C definition and standards very closely. This includes the usage of SOAP as communication and message format, WSDL as service description language and UDDI based name and directory services/servers (Alonso et al., 2004). The common syntax for web services specifications is XML, with data structures and formates described as XML documents. The communication including the required binding and mapping to a transport protocol, should be flexible enough to support a wide variety of protocols, such as TCP/IP, HTTP, or SMTP.

SOAP: Simple Object Access Protocol The interactions between service requester and provider are based on SOAP. The SOAP standard[26] defines how to organize information using XML in a structured and typed manner, in particular it specifies the following:

- A message format for one-way communication, describing the packaging of information into an XML document.
- The usage conventions for SOAP messages with RPC interactions, including RPC invocation and reply with a SOAP message.
- The rules for processing SOAP messages. The resolution of XML elements and the reaction to errors in form of missing, incomplete or wrong elements.
- An example description how to bind SOAP to a transport layer (HTTP, SMTP).

SOAP is a stateless and one-way protocol. It ignores the semantics of the message. The interaction between the caller and callee has to be encoded within the SOAP document. SOAP is designed for loosely-coupled applications using one-way asynchronous messages for interaction. Communication patterns such as request-response or more complex such as two-way synchronous messaging and PRC has to be implemented by the underlying system or middleware.

WSDL: Web Services Description Language WSDL[27] is an XML-based language that provides a model for describing Web services. It provides a machine-readable description of the operations offered by the service. A WSDL specification consists of two parts: (1) abstract part and (2) concrete part. The later (2) defines services as collections of network endpoints, or ports. A port is defined by associating a network address with a reusable binding. The abstract part is separated from their concrete use to facilitate reuse of definitions. The abstract part (1) consists of port type definitions, which are collections of related operations. Each operation is defined by an exchange of messages. The message is the basic building block for a service. By default,

[26] `http://www.w3.org/TR/soap/`

[27] `http://www.w3.org/TR/wsdl`

the typing is the same type system as the XML Schemas. If required, the typing system can be specified explicitly and thus changed.

WSDL is not a requirement of a SOAP endpoint, but it is often a prerequisite for automated client-side code generation in many Java and .NET SOAP frameworks. Some industry organizations, such as the Web Services Interoperability industry consortium (WS-I), mandate WSDL in their definition of a Web service. The WS-I Basic Profile is a specification that provides interoperability guidance for core Web Services specifications such as SOAP, WSDL, and UDDI. The profile uses Web Services Description Language (WSDL) to enable the description of services as sets of endpoints operating on messages.

UDDI: Universal Description Discovery and Integration Web services are meaningful only if potential users may find information sufficient to permit their execution. The focus of Universal Description Discovery & Integration (UDDI) is the definition of a set of services supporting the description and discovery of (1) businesses, organizations, and other Web services providers, (2) the Web services they make available, and (3) the technical interfaces which may be used to access those services. Based on a common set of industry standards, including HTTP, XML, XML Schema, and SOAP, UDDI provides an interoperable, foundational infrastructure for a Web services based software environment for both publicly available services and services only exposed internally within an organization.[28] The information available in a UDDI registry can categorized into three components:

- **White pages** – Listings of organizations, contact information, services provided by the organizations; find web services for a given business

- **Yellow pages** – Classification of organizations/companies and web services in a predefined or user specific taxonomy; find services in a category

- **Green pages** – Information for the invocation of a given web service.

The categories were derived from a similar structured telephone directory.

Current UDDI search mechanism can only focus on a single search criterion, such as business name, business location, business category, or service type by name, business identifier, or discovery URL. For a business solution, it is very normal to search multiple UDDI registries and then aggregate the returned results. The aggregation employ filter and ranking methods. The first big player in federated web service discovery was IBM in 2001 with the Business Explorer for Web Services (BE4WS).

RESTful Web Services

More recently, REpresentational State Transfer (RESTful) Web services have been regaining popularity, particularly with Internet companies. These also meet the W3C definition, and are often better integrated with Hypertext Transfer Protocol (HTTP) than SOAP-based services. They do not require XML messages or WSDL service-API definitions, although an improved support for RESTful web services has been integrated into the WSDL 1.2/2.0 specification.

The REST architecture was originally introduced for building large-scale distributed hypermedia systems. It is based on four principles (Pautasso et al., 2008):

- *Resource identification through URI.* A RESTful Web service exposes a set of resources which identify the targets of the interaction with its clients. Resources are identified by Uniform Resource Identifiers (URI)[29], which provide a global addressing space for resource and service discovery.

[28] www.oasis-open.org/committees/uddi-spec/doc/spec/v3/uddi-v3.0.2-20041019.htm

[29] RFC 3986 – `http://tools.ietf.org/html/rfc3986`

- *Uniform interface.* Resources are manipulated using a fixed set of create, read, update, delete operations: `PUT`, `GET`, `POST`, and `DELETE`. `PUT` creates a new resource, which can be then deleted using `DELETE`. `GET` retrieves the current state of a resource in some representation. `POST` transfers a new state onto a resource.
- *Self-descriptive messages.* Resources are decoupled from their representation so that their content can be accessed in a variety of formats (e.g., HTML, XML, plain text, PDF, JPEG, etc.). Meta data about the resource is available and used, for example, to control caching, detect transmission errors, negotiate the appropriate representation format, and perform authentication or access control.
- *Stateful interactions through hyperlinks.* Every interaction with a resource is stateless, i.e., request messages are self-contained. Stateful interactions are based on the concept of explicit state transfer. Several techniques exist to exchange state, e.g., URI rewriting, cookies, and hidden form fields. State can be embedded in response messages to point to valid future states of the interaction.

The strength of RESTful web services is the usage of existing and well known standards (HTTP, XML, URI, MIME). The required infrastructure, such as HTTP clients and servers, is available for all major platforms (programming language, operating system or hardware). The lightweight infrastructure (minimal tooling requirements, inexpensive hardware) provides a reduced adoption barrier.

The two approaches of "big" and RESTful web service differ in the number of architectural decisions that must be made and in the number of available alternatives. The discrepancy between freedom-from-choice and freedom-of-choice explains the complexity difference perceived. However there are significant differences in the consequences for the resulting development and maintenance costs. RESTful web services are well suited for basic, ad hoc integration scenarios, "big" web services are more flexible and address advanced quality of service requirements commonly occurring in enterprise computing (Pautasso et al., 2008). To support this, the "big" web service stack can include further protocols for security (WS-Security), reliability (WS-ReliableMessaging, WS-Reliability), or transactions (WS-AtomicTransaction, WS-BusinessActivity, WS-CAF). This stack can be abbreviated with WS-*, as most of the standards start with this prefix.

Similar Efforts to Web Services

Other approaches with nearly the same functionality as web services are Common Object Request Broker Architecture (CORBA), Microsoft's Distributed Component Object Model (DCOM) or SUN's Java/Remote Method Invocation (RMI) and other Middleware systems. A More basic effort is XML-RPC. It is a remote procedure call protocol which uses XML to encode its calls and HTTP as a transport mechanism (St. Laurent et al., 2001). XML-RPC is a predecessor for SOAP based messaging.

JSON-RPC is a remote procedure call protocol encoded in JavaScript Object Notation (JSON, Crockford 2006). It is a very simple protocol (and very similar to XML-RPC). It defines only a handful of data types and commands. In contrast to XML-RPC or SOAP, it allows for bidirectional communication between the service and the client, treating each more like peers and allowing peers to call one another or send notifications to one another. It also allows multiple calls to be sent to a peer, which may be answered out of order. As transport protocol HTTP and sockets are supported. One main application for JSON-RPC is the communication between a web browser and a server for dynamically loading web site content.

2.4.3.3 Semantic Web Services

Web services are an approach to provide remote procedure calls in a heterogeneous environment such as the web. A traditional web service takes a set of input parameters, calls the web service and receives a result. The input and output is often typed as primitive data types (strings, numbers) and the description of what the service does is either a short text or just encoded in the function name. The standards WS-* or REST define only the syntax of a web service call, but not the semantics. The semantic annotation of a web service is the basic idea of semantic web service. For instance, take a web service for the common bioinformatics task of multiple sequence alignment. With a normal web service the input would usually be a list of strings and the output would be a single string or xml document with the alignment. In contrast, the semantic version describes the input as a list of typed sequences with the same type, e.g., amino acid sequences for proteins or nucleotide sequences for DNA/RNA. The result will be annotated as a sequence alignment of the same type.

With this enhancement the machine interoperability can be improved. The main goal of the semantic annotation of web services is to facilitate a more meaning full and automatic discovery, selection, invocation and possible composition of web services. There have been several proposed ideas on how to formalize and standardize this semantic description, for example OWL-S, WSMO + WSML, WSDL-S or SAWSDL.

OWL-S

OWL-S (formerly DAML-S) is an ontology of services that makes these semantic functionalities possible. OWL-S was submitted to the W3C in 2004 (Martin et al., 2004). The OWL-S ontology has three main parts: the service profile for advertising and discovering services; the process model, which gives a detailed description of a service's operation; and the grounding, which provides details on how to interoperate with a service, via messages. The OWL-S follows a layered approach. It uses the Ontology Web Language (OWL) framework as basis for syntax and formalizing semantics.

WSMO + WSML

WSMO or Web Service Modeling Ontology is a conceptual model for relevant aspects related to Semantic Web Services (de Bruijn et al., 2005a). It provides an ontology-based framework, which supports the deployment and interoperability of Semantic Web Services. The WSMO has four main components: goals, ontologies, mediators, and web services. The Web Service Modeling Language or in short WSML (de Bruijn et al., 2005b) provides the formalization for syntax and semantics for the WSMO. WSML uses logical formalisms such as description logics, first-order logic and logic programming. WSML has a human-readable syntax, XML and RDF syntax. Furthermore, there is a mapping for a common subset between WSML ontologies and OWL ontologies.

The authors of the WSMO W3C submission describe the reasons for this alternate standard proposal as follows: “One proposal for the description of Semantic Web services is OWL-S. However, it turns out that OWL-S has serious limitations on a conceptual level and also, the formal properties of the language are not entirely clear (Lara et al., 2005). For example, OWL-S offers the choice between different languages for the specification of preconditions and effects. However, it is not entirely clear how these languages interact with OWL, which is used for the specification of inputs and output.”

WSDL-S

The WSDL-S (Akkiraju et al., 2005) is a W3C submission that takes a different approach to attaching semantic information to a web service. It assumes that formal semantic models relevant to the services already exist. With this assumption, it is possible to maintain the model outside of WSDL documents. The models can be referenced from the WSDL document using the WSDL extensibility elements. To describe these models any language such as OWL-S, WSMO or even UML could be used. The semantic information should include definitions of the precondition, input, output and effects of web service operations.

The advantage of WSDL-S are *upward compatibility*, in contrast to OWL-S with its fixed service and ontology description, *independence of ontology representation language* and *easier upgrade path* with the reuse of existing WSDL service descriptions.

SAWSDL
Recognizing the advantages of the WSDL-S submission the W3C developed and finalized Semantic Annotations for WSDL and XML Schema, SAWSDL as a recommendation (Farrell and Lausen, 2007). Additionally it was included in the **WSDL 2.0** recommendation (Chinnici et al., 2007). SAWSDL defines three extensibility attributes to WSDL 2.0 elements. The *modelReference*, that allows the association between a WSDL component and a concept in some semantic model. The *liftingSchemaMapping* and *loweringSchemaMapping* allow mappings between semantic data and XML.

2.4.4 Summary

Information, especially also for semantic search, is available from various sources and formats. There is explicit data in form of tables or databases and information in form of free text, as in scientific articles and web pages. The texts are stored in different document formats and content types making a text extraction step necessary. The extraction of text from a binary-encoded document can be a challenging task, as, for example, with PDF documents.

Web search is the current standard to navigate the rich information accessible on the web. Many users identify relevant information through a web search, relying on the ranking of the search engine as quality indicator.

The basis for processing the information — not only for search engines — is the automated content retrieval of data and text. For the web as information source, crawling is the most common approach for automated retrieval. Crawling, especially on the whole web scale, is a time and resource intensive task. The size and speed requirements demand a distributed approach. Here, the Map-Reduce-Pattern has been shown to be a successful solution to implement a scalable and high-performance web crawler.

As an alternative to crawling, there are web services, as they offer a direct programmatic interface for automated retrieval and other functional calls. To improve the programmatic usage of web services, the standardized semantic annotation of web services offers a way for more detailed description of the required parameters and result.

Next, we will describe the text mining and entity recognition techniques, which can be applied to the texts, extracted from the different information sources.

2.5 Text Mining and Entity Recognition

Semantic search engines need to extract information from their used information sources. Text mining and entity recognition techniques are suitable approaches to match ontology concepts and named entities in text. Here, we introduce the relevant ideas and algorithms to implement these two approaches.

2.5.1 Finding Ontology Concepts in Text

Structured background knowledge, such as ontologies, are an important source of information (see Section 2.3). The annotation of documents with concepts from the background knowledge is a challenging task. If done by hand with human curators, it is a very time consuming and thus costly step. Additionally the semantic gap between unannotated documents and already available annotations can only be lessened with an automated support.

The goal of automated approaches is to provide related concepts for a given text. With text mining the basic principle is to match the label of ontology concepts with the text. The majority of the current ontologies have been designed to annotate data or to be used as classification schemes. But originally, they are not intended for text mining. Therefore the identification of ontology concepts in free text remains a challenging task. For instance, an assessment for extracting GeneOntology concepts revealed performances around 20% success rate only (Ehrler et al., 2005). The difficulties of automating manual annotation is evident from the fact that only as few as 15% of manually annotated concepts appear literally in the associated abstracts. Text mining, in particular for the biomedical domain, uses various techniques and algorithms. There is natural language processing, information retrieval and machine learning, to identify the relevant concepts (Jensen et al., 2006). They all have to deal with the following groups of problems.

Ad-hoc Variations of Names

To begin with, terms in vocabularies and labels of concepts in ontologies appear in many, slight or severe, variations in natural language texts:

- orthographic: IFN gamma, Ifn-γ
- morphological: Fas ligand, Fas ligands
- lexical: hepatitic leukaemia, liver leukemia
- structural: cancer in humans, human cancers
- acronyms/abbreviations: MS, Nf2
- synonyms: neoplasm, tumor, cancer, carcinoma
- paragrammatical phenomena/typographical errors: cerevisae, nucleotid

Some of the terms or concepts encountered in texts are rather ad-hoc creations, which cannot be found in any background knowledge.

Synonymity of Ontological Concepts

As mentioned before, concept labels in a vocabulary or ontology might not appear literally in a text, but authors rather use synonyms for the same concept. First of all, this may complicate normal searches:

- When searching for “digestive vacuole”, results should also contain texts that mention “phagolysosome”;
- mentioning of “ligand” refer to the concept “binding”;
- an “entry into host” might occur as an “invasion of host”.

In the Plant ontology for example, many synonyms exist for the same structure in different species. “Inflorescence” is referred to as “panicle” in rice, and as “cob” in sorghum, and “spike” in wheat, for instance. Some labels are also intra-ontology synonyms, for example the label “eye” in AnoBase can refer to the eye spot or the adult compound eye.

Ambiguity of Ontological Concepts

Concepts can have a very specific meaning in biomedical research, but in other contexts the textual labels mean other things. Examples are “development”, “envelope”, “spindle”, “transport”, and “host”. Protein names such as “Ken and Barbie”, “multiple sclerosis” or “the” that resemble common names, diseases, or common English words are especially hard to disambiguate. The same problems arise from drug names like “Trial” or “Act”.

This is no problem for human annotators, but it is a challenge to automated methods, which identify ontology terms in text. Classical approaches to word sense disambiguation use co-occurring words or terms. However, most treat ontologies as simple terminologies, without making use of the ontology structure or the semantic similarity between terms. Another useful source of information for disambiguation can be metadata available for a text.

The challenge for the biomedical domain is the rapid growth of the literature in terms of new words and their senses, with the situation getting worse with the use of abbreviations and synonyms. This illustrates the exact need in the case of the biomedical domain; the development of statistical approaches that utilize “established knowledge” (like thesauri, dictionaries, ontologies and lexical knowledge bases) and require no or only some parsing of the text in order to perform the correct annotation.

Word sense disambiguation (WSD) deals with relating the occurrence of a word in a text to a specific meaning, which is distinguishable from other meanings that can potentially be related to that same word (Schuemie et al., 2005). WSD is in general a classification problem: given an input text and a set of sense tags for the ambiguous words in the text, assign the correct senses to these words. Sense assignment often involves two assumptions:

1. within a discourse, for example a single document, the word is only used in one sense (Gale et al., 1992),
2. words have a tendency to exhibit only one sense in a given collocation of neighboring words (Yarowsky, 1993).

As shown in Table 2.4, WSD algorithms can be distinguished as supervised, unsupervised, or using established knowledge (Schuemie et al., 2005; Agirre and Edmonds, 2006). In the biomedical domain researchers have focused on supervised methods (Hatzivassiloglou et al., 2001; Liu et al., 2004; Gaudan et al., 2005; Pahikkala et al., 2005) and using established knowledge (Schijvenaars et al., 2005; Humphrey et al., 2006; Hakenberg et al., 2008; Farkas, 2008) to perform gene name normalization and resolve abbreviations. According to the recent BioCreAtIvE challenge, the former problem can be solved with up to 81% success rate (Hakenberg et al., 2008) for human genes, which are challenging with 5.5 synonyms per name (therefore many genes are named identically).

Resolving ambiguous abbreviations achieves higher success rates of close to 100%, as the task is less complex when long forms of the abbreviated terms are in the document (Gaudan et al., 2005). The above approaches use cosine similarity (Schijvenaars et al., 2005), support vector machines (SVM, Gaudan et al. 2005; Pahikkala et al. 2005), Bayes, decision trees, induced rules (Hatzivassiloglou et al., 2001), and background knowledge sources such as the Unified Medical Language System (UMLS) (Bodenreider, 2004), Medical Subject Headings (MeSH) (Nelson et al., 2001), and the Gene Ontology (GO) (Ashburner et al., 2000). Two approaches use metadata, such as authors (Farkas, 2008) and Journal Descriptor Indexing (Humphrey et al., 2006). Most of the unsupervised approaches so far were evaluated outside the biomedical domain (Schütze and Pedersen, 1995; Schütze, 1998; Pedersen and Bruce, 1998; Purandare and Pedersen, 2004; Yarowsky, 1995; Dorow and Widdows, 2003; Mihalcea, 2004), with the exception of Widdows et al. (2003), who used relations between terms given by the UMLS for unsupervised WSD of medical documents and achieved 74% precision and 49% recall. Another approach that uses the UMLS as background knowledge for WSD is that of Leroy and Rindflesch (2005), who compared

	publ.	Data	Background knowledge	Approach	Experiment	Accuracy
Established Knowledge	Schijvenaars et al. (2005)	gene definition & abstract vector	5 human gen. dbs & MeSH	cosine similarity	52,529 Medline abstracts, 690 human gene symbols	92.7%
	Humphrey et al. (2006)	free text	UMLS, Journal Descriptors	Journal Descriptor Indexing (JDI)	45 ambiguous UMLS terms (NLM WSD Collection)	78.7%
	Hakenberg et al. (2008)	Medline abstracts	BioCreative-2 GN lexicon & text, EntrezGene, UniProt, GOA	motifs from multiple sequence alignments	BioCreative-2 GN challenge	81%
	Farkas (2008)	Medline abstracts	list of gene senses, EntrezGene	inverse co-author graph	BioCreative GN challenge	97%P
Supervised	Hatzivassiloglou et al. (2001)	XML tagged abstracts, positional info, PoS	-	naive Bayes, decision trees, inductive rule training	protein/gene/mRNA assignment: 9 million words (mol. biol. journals)	85%
	Ginter et al. (2004)	text	-	word count, word cooc	-	86.5%
	Liu et al. (2004, 2002)	Medline abstracts	UMLS terms	UMLS term cooc	35 biomedical abbreviations	93%P
	Gaudan et al. (2005)	abbreviations in Medline abstracts	-	SVM	build dictionary, use for abbreviations occurring with their long forms	98.5%
	Pahikkala et al. (2005)	gene symbol context (n words +/-)	-	SVM	-	85%
Unsupervised	Schütze and Pedersen (1995); Schütze (1998)	document	-	LSA/LSI, 2^{nd} order cooc	170, 000 documents, 1013 terms (TREC-1) (Wall Street Journal)	⇑ 7–14%
	Pedersen and Bruce (1997)	word cooc, PoS tags	WordNet	average link clustering	13 words, ACL/DCI	73.4%
	Pedersen and Bruce (1998)				Wall Street Journal Corpus	
	Purandare and Pedersen (2004)	-	-	1st, 2nd order context vectors (coocs within 5 positions)	24 Senseval-2 words, Line, Hard, Serve corpora	44%
	Yarowsky (1995)	text	few tagged data, WordNet	co-training, collocations	12 common Engl. words × 4000 instances	96.5%
	Mihalcea (2004)	-	-	co-training & majority voting	Senseval-2 generic English	⇑ 9.8%
	Dorow and Widdows (2003)	-	WordNet	noun coocs, Markov clustering	-	-

Table 2.4: Algorithms for Word Sense Disambiguation

the results from a naive Bayes classifier and other algorithms (decision tree, neural network). They conclude that different senses in the UMLS could contribute to inaccuracies in the gold standard used for training, leading to varied performance of the WSD techniques. Another approach by Dorow and Widdows (2003) is based on a graph model representing words and relationships (co-occurrences) between them and uses WordNet (Fellbaum, 1998) for assigning labels.

Interestingly, most of the above approaches consider the background knowledge sources as terminologies, without taking into account the taxonomic structure or the terms' semantic similarity (Rada et al., 1989; Sussna, 1993; Resnik, 1995; Lin, 1998; Lord et al., 2003; Azuaje et al., 2005; Schlicker et al., 2006; del Pozo et al., 2008). This gap is filled by Alexopoulou et al. (2009). They introduce three approaches using ontologies with inference and semantic similarity and the use of metadata to solve the problem of WSD for ontological concepts. The use of ontologies and metadata can improve results for WSD.

Stemming and Missing Words

Some aspects for finding terms in text refer to the actual processing of natural language and appear rather technical. Very often, words will appear in different forms, such as "binding" and "binds". These refer to the same concept, which can be solved by resolving words to their stem ("bind"). However, the analogous reduction of "dimerisation" to "dimer" is more questionable. The former talks about the process, the latter about the result. A similar example is "organization", where a transformation into "organ" is invalid.

Texts contain additional words that are missing in the ontological term. This happens, for instance, when a text contains further explanations that describe findings in more detail. An example is "tyrosine phosphorylation of a recently identified STAT family member" that should match the ontology term "tyrosine phosphorylation of STAT protein." In general, matching is allowed to ignore words such as "of", "a", "that", "activity", but obviously not "STAT". Additional background information on term variations is needed to know that a "family member" can refer to a protein.

Formatting of terms represents another source for potential matching errors. Concepts in ontologies contain commas, dashes, brackets, etc., which require special treatment. In "thioredoxin–disulfide" the dash can be dropped, whereas in "hydrolase activity, acting on ester bonds" the clause after the comma and therefore the comma is important, but unlikely to appear as such in text. Concepts containing additions such as "(sensu Insecta)" may have important contextual information, but are also less likely to appear in text.

Ontology Specific Issues

Concept overlaps – Some concepts can overlap in their labels or synonyms: in many cases there is a difference between the written document and the intended meaning of the document. Unfortunately, researchers do not have strict and formal ontologies or nomenclatures in their minds when composing a scientific article; in most of the cases they might use a parent concept to refer to a child concept, or vice-versa. For example, many people are treating the MeSH terms *cardiovascular disease* and *coronary artery disease (CHD, CAD)* the same, although the latter is a child of the first.

Descriptive labels – In most of the cases, the labels in an annotation ontology cannot be used directly for text mining, often due to their explanatory nature. For example, it is unlikely that the Gene Ontology term "cell wall (sensu Gram-negative bacteria)" will appear as such in a text. Concepts like "positive regulation of nucleobase, nucleoside, nucleotide and nucleic acid metabolism" and "dosage compensation, by inactivation of X chromosome" are almost complete sentences and are also unlikely to be found as such in a text.

Ambiguity – Ambiguity results either from identical abbreviations for different concepts, or, in general, tokens that can refer to terms that might or may not be of our interest. An example of an ambiguous *abbreviation* is “CAM” that can stand for “constitutively active mutants”, “cell adhesion molecule”, or “complementary alternative medicine”. The second category of ambiguities – and the most difficult to handle – is that of terms that (in the context of anatomy) can refer to different species. An example of such ambiguities is “embryo”, which can be a chicken, mouse, human, or even zebra fish embryo.

2.5.2 Entity Recognition

Although finding ontological concepts in free text is important, there are many additional relevant things to find in the text, for instance:

- proteins
- genes
- species
- mutations

The task to find these entities is called entity recognition. The identification of ontology terms can be seen as a sub species of the more general task of entity recognition. As a consequence many of the techniques and problems described above are also valid for entity recognition (Manning et al., 2008). In the case of protein and gene name identification, there are some other difficulties (Hakenberg et al., 2007). One challenge is the increased ambiguity and synonymity of names. Often the gene name and the protein are used as synonyms by the authors or a gene has the same name in different organisms. Another task is to deal with the number of entities one can find. As an example, the UniProtKB/TrEMBL protein database contains over $4,500,000$ entries. The real number to match is even higher as one has to integrate all the synonyms and variants of the protein names and genes. For the case of identification on which species an article talks about, a reoccurring problem is, that the species is sometimes never mentioned in the text. For mentions of point mutations one has to recognize the mutations and the related proteins to have a useable result (Lee et al., 2007; Baker and Witte, 2006; Winnenburg et al., 2009). Moreover many cases of ambiguities and missing concepts can only be resolved, if any information available from the text is used. For example, a matched gene name corresponds to a list of several proteins. To reduce the list of candidates, one can try to find the species or a point mutation in the text and than verify if they match with any of the candidate proteins.

The importance of entity recognition and their relations has been acknowledged by the scientific community (Spasic et al., 2005; Jensen et al., 2006). There have been efforts to establish benchmarks and competitions to advance the research. Examples for this are the “bioentity recognition task at JNLPBA” (Kim et al., 2004) or the “Critical Assessment of Information Extraction in Molecular Biology” (BioCreAtIvE) (Hirschman et al., 2005). In the BioCreAtIvE II for the gene mention task the best systems (Hakenberg et al., 2007) could achieve a precision of 78.9% and a recall of 83.3% as an example for the current state of the art.

2.5.3 Algorithmic principles for concept and entity recognition

In the following, we will introduce different algorithms to implement concept and entity recognition. There are the dictionary, rule-based, and alignment-based algorithms. As one of most basic function for text analysis, different implementation of string matching algorithms are presented, including hashing, automata, and tree-based approaches.

Dictionaries

Dictionaries are the most basic form to identify concepts in text. They use a fixed list of concepts and identify the corresponding occurrences of the concept labels in text. This can be done very efficiently, for example by constructing Finite State Automata (Hakenberg et al., 2008). Unfortunately the simplicity of a fixed dictionary can decrease the quality of the matching, e.g., reduce precision and recall. This can be attributed to missing variation and unknown concept labels. As reported by Hirschman et al. (2002) this can range from 16 to 69% of missed concepts. Furthermore, a dictionary approach may need an additional disambiguation step. Depending on the implementation, it has been reported that the correct identified concepts with their corresponding senses can go as low as 2 to 7% (Hirschman et al., 2002).

Rules

Rules allow to describe an infinite number of variations. For instance, McDonald (1996) uses rules in three stages. The first stage is to identify candidates with a dictionary and lexical hints. In the second stage the candidates are classified context-based using the surrounding words. For the third stage, abbreviations resolution for identified concept is applied. Rules can also be applied for handling morphological modifications. This is demonstrated by Ananiadou (1994) with a system of layered rules.

For the best results its necessary to combine rules with other approaches. Fukuda et al. (1998) achieved good results by applying first a rule-based approach to finding core-terms with five rules. Secondly, the core-terms are connected, using other rules and a part of speech tagger.

More recently Hakenberg et al. (2008) use rules to expand their dictionary by generating variations of gene and protein names after splitting them at visual gaps. They identify a visual gap for example in the gene name “BRAC1” between the “C” and the “1” because of the change from letters to numbers. This allows for variations like “BRAC-1” or “BRAC 1”. Additionally arabic numbers can be interchanged with roman numbers, thus accepting also “BRAC I” as a valid gene name.

In general, rule-based systems often need to be adapted for each domain or research aspect. This can be done by hand or with (semi-)automated approaches. Caporaso et al. (2007) use a bootstrapping approach to generate *Regular-Expressions* rules for identifying protein-point-mutations in scientific literature.

Alignment

An alternative way of handling unknown variations of term labels is to employ alignment algorithms. The basic operations of an alignment are four operations: match, substitution, deletion and insertion. In combination with a scoring scheme the task of an alignment is to find a combination of operations with the best score in the search space. Well known algorithms for this task are, for instance, the global sequence alignment (Needleman and Wunsch, 1970), the local sequence alignment (Smith and Waterman, 1981), and the Basic Local Alignment Search Tool (BLAST). BLAST provides improved local sequence alignment and relevance scores (Altschul et al., 1990).

For text mining these alignment algorithms can be applied on different levels. It is used for characters or digits (Tsuruoka and Tsujii, 2003), words (Doms, 2004) or in a translated version. For the translation approach the text and concept labels are translated to nucleotides. In a second step the BLAST algorithm is used to find the best alignments (Krauthammer et al., 2000).

String Matching Algorithms

The basis for all the above mentioned algorithms is an efficient matching of Strings. For dictionaries the task is to find direct occurrences. Rules and patterns also rely on the matching of textual fragments. Similar, the alignment works on position information of tokens.

The annotation task requires to match a list of Strings $d = d_0, d_1, d_2, \ldots, d_k$ with a given text String s. The String s is typically longer than the Strings in the list d. The goal is to find all matching strings d_j in s. Most traditional string search algorithms[30] such as Knuth-Morris-Pratt (Knuth et al., 1977) or Boyer-Moore (Boyer and Moore, 1977) use one item d_i and match it against s. For matching a dictionary, this results in a matching algorithm with a complexity that depends on the number of entries in the list d and the complexity of each comparison. To address this complexity issue, the algorithms have to be designed in such a way, that they check all list items d_i at once. With this design the lower bound for the complexity is length of the string s. There are multiple options for implementing such an algorithm. These algorithms usually create a data structure from the list d to facilitate the efficient lookup. The complexity and memory requirements for the creation and storing of the data structures are the trade-off for the reduced runtime during the actual search. There are different possibilities to create such a data structure, which are described in the subsequent paragraphs.

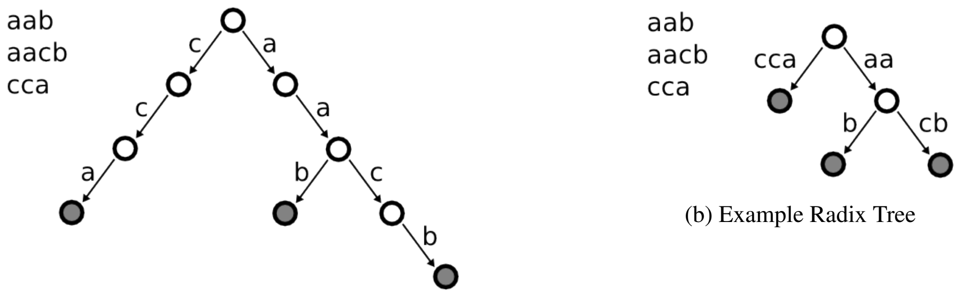
Hashing The first option is to simply use a hash code based method. Given a collision-free hash function a hashmap can provide a lookup with a constant complexity ($\mathcal{O}(1)$). The main complexity contribution of the hash function is the function complexity and how often it is used. Given that the hash function uses all characters in the string, the lower bound is the length of the string $|s| = l$. For the task of finding all matching substrings of s in the list, this ends up with a total complexity of $\mathcal{O}(l^2)$. But a hashmap is not the only option for hashing and string search. The Robin-Karp (Karp and Rabin, 1987) algorithm uses a hash function to reduce the number of character comparisons.

Finite Automata A different approach is to treat the task as a pattern matching problem. For a given dictionary in the simplest case, this results in a non-deterministic finite state automaton (NFA). For an efficient matching this automaton has to be transformed into a deterministic finite state automaton (DFA). This can be done with a powerset construction. Unfortunately this step potentially generates from n initially states 2^n new states. This space requirement during the construction may be a limiting factor.

Trie An alternative representation of automata are trees. To avoid the limiting step of a powerset construction, it is the goal to construct trees with deterministic branching conditions. Trees especially for string search in dictionaries are called tries. This word was created from the context of re***trie***val. A trie is a character-based prefix tree. In general, tries can be as fast as a hashmap or faster. Instead of calculating a hash key, the trie is traversed. This allows for an early termination and a possible speedup compared to a hashmap. Consider the following example: Given a dictionary d with three entries $d_0 = aab$, $d_1 = aacb$, $d_2 = cca$ and the query $q = accb$. For an illustration of the resulting trie see Figure 2.8a. In the case of the hashmap the hash function calculates the hash key using the whole length $|q| = 4$ and does the lookup. In contrast for the trie the traversal of the tree is a follows. In the first step the first character of $q[0] = a$ is used to select the next node. In the second step the second character $q[1] = c$ has no next node and the lookup in the trie terminates without checking the remaining characters in q.

Radix Tree This data structure radix tree is a trie with the additionally constraint, that it merges nodes with only one child. This has the effect that, in contrast to the very memory intensive trie, the radix tree has a smaller memory footprint. But it requires also a slightly more complex node selection algorithm during the traversal of the tree. For an example of an radix tree see Figure 2.8b. The radix tree originally was introduced as PATRICIA (**P**ractical **A**lgorithm **T**o **R**etrieve

[30] `http://www-igm.univ-mlv.fr/~lecroq/string/`



(a) Example Trie

(b) Example Radix Tree

Figure 2.8: Example for the Trie and Radix Tree. The data structures contain the dictionary $d = aab, aacb, cca$

Information **C**oded in **A**lphanumeric) by Morrison (1968) and as crit bit tree by Gwehenberger (1968).

Prefix search, as done by trie and radix tree, is a task that is also used in context of routing of packages, for instance with the internet. For an efficient routing, it is required to identify the best matching prefix in a lookup table of routing addresses. A good review on the different algorithms, including radix tree, is presented by Waldvogel et al. (2001). Furthermore, the Radix Tree as a data structure is also available in the Linux kernel and can be used for such tasks (Corbet, 2006).

Suffix Tree Another specialization for string search with trees is the suffix tree. It is a radix tree containing all suffixes from a string (Weiner, 1973; McCreight, 1976). With this proposed data structure it is possible to solve many string search problems. The biggest known issue with suffix trees was to construct the tree itself. This is solved by Ukkonen (1995). The idea is to construct the suffix tree online and not by adding all suffixes (Giegerich and Kurtz, 1997). For the usage of suffix trees in conjunction with a dictionary or multiple patterns, the suffix tree has been extended to the Generalized Suffix Tree (Chi and Hui, 1992; Bieganski et al., 1994). It can handle multiple source strings in one suffix tree. Multiple pattern string search using suffix tree like data structures have also been proposed by Aho and Corasick (1975) and Commentz-Walter (1979).

2.5.4 Summary

Ontologies contribute to the background knowledge for semantic search. The automatic identification of ontology concepts in text with text mining applies different algorithms to match the concept labels and text. For this task the properties of text with different types of variations, synonyms, and ambiguity and ontology issues with naming conventions and descriptive labels are discussed. Furthermore, entities are an additional information extraction target and require adaptation with respect to synonyms, abbreviations, ambiguity and the large amount of entities.

The algorithmic approaches to implement ontology-based text mining and entity recognition are presented. This includes dictionaries, rules, and alignment. An in-depth description for the fundamental task of string matching is given, including a motivation for the advantages of tree-based string matching compared to hashing or automata, as an implementation for rules. Later in Section 3.1.3, several of these string matching algorithms are compared with each other as part of the GoWeb implementation and evaluation.

2.6 Indexing and Ranking for Information Extraction

Text search and the retrieval of relevant documents is the task of any search engine. To avoid the complexity of scanning each document for each query, the well established technique of *indexing* is used. An index is a data structure to facilitate lookup with complexities below $\mathcal{O}(n)$. For example a constant $\mathcal{O}(1)$ complexity for retrieving the list of relevant documents for a given term. But querying can also be carried out without an index. For typical data and queries, building an index requires about 10 – 50 times the cost of a string search. If queries are rare, or the data is extremely volatile, it is reasonable to use a document-oriented querying strategy.

Given the size and growth of document collections such as a web crawl, file-server or literature repositories, it is necessary to sort the possible hits to a relevance scheme, e.g., statistical similarity. This *ranking* allows the search engines to return only a limited set of results. There are several indexing and ranking techniques, which are presented in this section.

2.6.1 Indexing Techniques

Taking all of these factors into account, implementers of search engines must design their systems to balance a range of technical requirements:

- effective and fast resolution of queries
- use of features: query term proximity, anchor strings and URL terms
- minimal use of other resources (disk, memory, bandwidth)
- scaling to large volumes of data
- handle change in the set of documents
- advanced features: boolean or phrase querying

Many different types of index have been described. The most efficient index structure for text query evaluation is the inverted file: a collection of lists, one per term, recording the identifiers of the documents containing that term.

For querying of larger collections, inverted files are not the only technology that has been proposed. The two principal alternatives are suffix arrays and signature files (Zobel et al., 1998). However, given the probabilistic indexing method and fuzzy query of signature files, there are no situations in which signature files are the method of choice for text indexing even for exact-match Boolean style searching (Zobel and Moffat, 2006).

2.6.1.1 Inverted File Index

An inverted file index consists of two major components (Hawking, 2006b; Zobel and Moffat, 2006). The first component is the $searchstructure$ or $vocabulary$. It stores for each distinct word t

- a count f_t of the documents containing t and
- a pointer to the start of the corresponding $invertedlist$.

The vocabulary should be as complete as possible. All terms should be indexed, including numbers. This is especially important for web site collections. Any visible component of a page might be used as a query term, including elements, e.g., as parts of an URL, like domains. Additionally stop words have to be considered for phrase queries. They may be ignored for simple bag-of-words queries.

The second component of the index is a set of inverted lists. Each list stores for the corresponding word t the matching documents containing t. The list contains the document as identifiers d and the associated set of frequencies $f_{d,t}$ of terms t in document d.

The lists are represented as sequences of $\langle d, f_{d,t} \rangle$ pairs. As described, this is a *document level* index in that word positions within documents are not recorded. Together with an array of W_d values (stored separately), these components provide all the information required boolean and ranked query evaluation.

Searching and Ranking with an Inverted File Index

Ranking using an inverted file is illustrated in Figure 2.9. The first step is the query analysis and the extraction of the relevant search terms. Once the terms are identified, the algorithm is initialized with an array A of N partial similarity scores, the accumulators. Then, for each term t, the accumulator A_d for each document d mentioned in t's inverted list is increased by the contribution of t to the similarity of d to the query. Once all query terms have been processed, similarity scores S_d are calculated by dividing each accumulator value by the corresponding value of W_d. Finally, the r largest similarities are identified, and the corresponding documents are returned to the user.

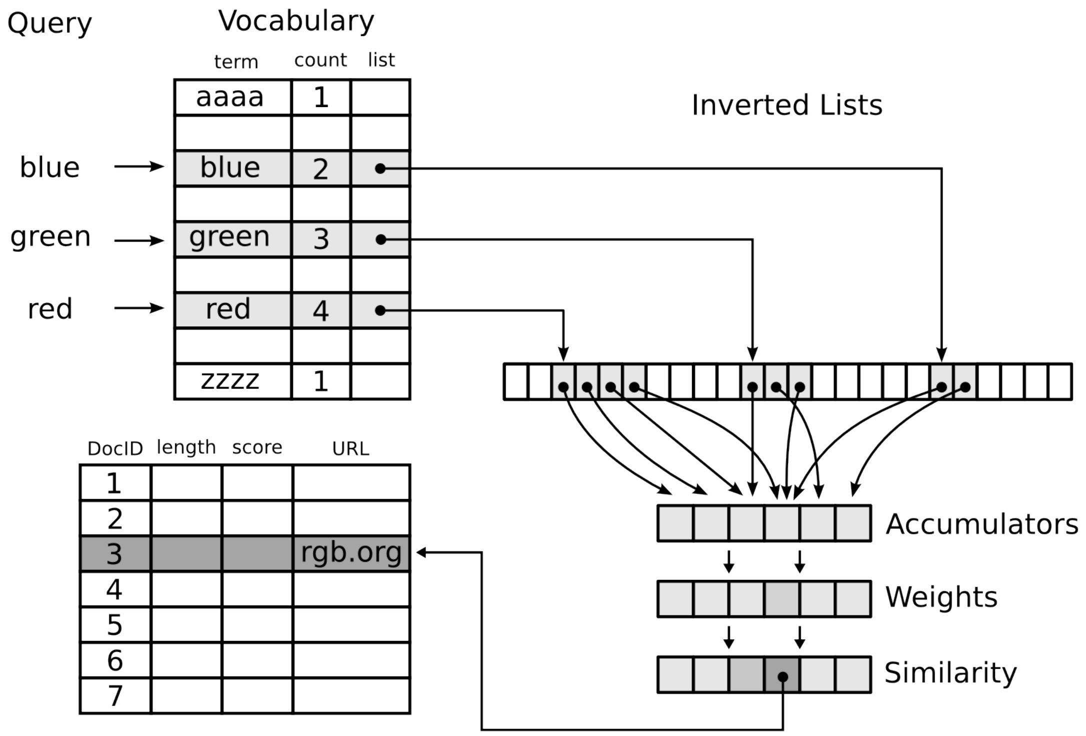


Figure 2.9: **Inverted File Index** — Components and general workflow for searching and ranking with an inverted file index

Indexing Word Positions

A document level index can be extended to include word positions. The frequency $f_{d,t}$ represents the number of occurrences of t in d. It is straightforward to modify each entry to include the $f_{d,t}$ ordinal word positions p at which t occurs in d. The result is a word-level inverted list containing pointers of the form $\langle d, f_{d,t}, p_1, \ldots, p_{f_{d,t}} \rangle$. The stored position might be either a character-based or word-based count. A word-based count allows for easier adjacency checks, as the word length is not needed in this representation.

Word positions can be used in a variety of ways during query evaluation, for example phrase queries. The word positions can also be used in bag-of-word queries, for example, using a similarity measure that makes use of adjacency or proximity mechanisms.

Index Creation

The creation of an inverted file index is a two step process. The first step is scanning. During the text scan of each input document, for each term in a document the indexer writes a data record to a temporary output file. Such a data record contains a document number, a term number and position information (for a term level index). This file will naturally be in document number order.

The second step is the inversion. The indexer sorts the temporary output file into term number order. For each term a list containing the information of corresponding document is created. In the end the indexer also records the starting point and length of the lists for each entry and adds this information to the term dictionary.

In-Memory Inversion There are two passes through the documents. A first pass collects term frequency information, sufficient for the inverted index to be laid out in memory in template form. A second pass then places pointers into their correct positions in the template, making use of the random-access capabilities of main memory. The advantage of this approach is that almost no memory is wasted compared to the final inverted file size, since there is negligible fragmentation. In addition, if compression is used, the index can be represented compactly throughout the process.

It is also possible to extend the in-memory technique to data collections where the index size exceeds memory size. This is done by laying out the index skeleton on disk, creating a sequence of partial indexes in memory. Each partial index is transferred in a skip-sequential manner to a template, which has been laid out as a disk file. With this extended method, and making use of compression, indexes can be built for multi-gigabyte collections using around $10 - 20$MB of memory beyond the space required for a dynamic vocabulary.

Sort-Based Inversion Other index construction methods are based on explicit sorting. In a simple form of this approach, an array or file of $\langle t, d, f_{d,t} \rangle$ triples is created in document number order, sorted into term order, and then used to generate the final inverted file. With careful sequencing and use of a multi-way merge, the sort can be carried out in place on disk using compressed blocks.

Merge-Based Inversion For larger volumes of data, the cost of keeping the complete vocabulary in memory is increasingly significant. Eventually, the index must be created as a composite of smaller chunks. The chunks are created using one of the previous mentioned techniques. This algorithm follows the Map-Reduce pattern introduced in Section 2.4.3.1.

For merge-based inversion, documents are processed in memory until a fixed capacity is reached. For this chunked processing, the intermediate data structure of the inverted list needs to be able to grow, when new information about a term is appended. This can be achieved using dynamically resizable arrays. When the memory limit is reached, the data structure is flushed to disk. This flush includes the vocabulary and the inverted lists. The lists are sorted in lexicographic order. This allows an efficient merge in the last step. After each flush the entire memory is cleared, including the vocabulary. Each working chunk starts with an empty configuration. After finishing all work chunks, the last step is to merge all the intermediate data files. The merge creates the final vocabulary on the fly. Typically the final inverted index is smaller as vocabulary information is no longer duplicated. So the final inverted lists can be represented more efficiently.

Distributed Inverted Indices

To distribute an inverted file index there are two possibilities for partitioning the data.

The *first* and simplest distribution regime is to partition the index via splitting the document collection. Each part can be handled independently on a machine. A local index is built for each sub collection. To handle a query, a server sends the query to each sub collection/machine. There it is evaluated against the local index. The server has after this the task of merging the subsets of answers to a global one. This can be implemented with a global comparable similarity score. Such a score might not be trivial, as some similarity measures use global frequencies or document counts. If they are calculated using only a sub collection, these scores may not be comparable.

The advantages of a document partitioned system are that document updates or new documents can be minimized (limited to one machine) and the computationally expensive parts of the search are distributed equally across all of the machines.

The *second* strategy is term partitioning. In a term-partitioned index, the vocabulary is used to create index slices for distribution. Each slice contains the full information about a subset of the terms. Because of this feature it is possible to handle a query, only with the relevant subset of the slices. This approach has the advantage that during the query handling the disk seeks and transfer operations are minimized. The disk seeks are minimal, because the inverted list of a term is still stored contiguously on a single machine rather than in fragments across multiple machines.

The drawback of term partitioning is that the disk transfer operations involve more data and the majority of the processing load falls to the coordinating machine. This single machine can become a bottleneck.

The inverted file index is an time and space efficient way to provide a full text index. The data structure can use compression algorithms and use pre-calculated static ranks to minimize the disk access during the search. The index is scalable, as the index itself can be distributed and the index creation can employ the Map-Reduce-Pattern to handle large amounts of data, such as a web crawl. Next, we will describe the suffix array as an alternative indexing approach.

2.6.1.2 Suffix Arrays

The Suffix array is an array of integers giving the starting positions for the suffixes of a string in lexicographical order (Baeza-Yates and Ribeiro-Neto, 1999). The array can be used as an index to quickly locate every occurrence of a substring within the string. The task is to find all suffixes that begin with the substring. Due to the lexicographical ordering, these suffixes will be grouped together in the suffix array. To traverse efficiently in the array binary search can be employed. In the simplest case this leads to a complexity of $\mathcal{O}(m \log n)$ time (m is the length of the substring). An optimization is to avoid redoing comparisons. This can be achieved by using extra data structures, which contain the information about the longest common prefixes of suffixes. This can help to reduce the complexity to $\mathcal{O}(m + \log n)$.

One way of interpreting the resultant array of pointers is that they represent a combined vocabulary and inverted index of every suffix string in the original text. The suffix array access provided access directly to the bytes instead of ordinal document identifiers.

For large-scale applications, suffix arrays have significant drawbacks. The pointer array is accessed via binary search, so compression is not an option. For a word-aligned suffix array, a 4-byte pointer is needed for each 6 bytes or so of text, and the underlying text must also be retained. In total, the indexing system requires around 170% of the space required by the input, all of it memory-resident (Zobel and Moffat, 2006).

Another drawback is that there is no equivalent of ranked querying. All but simple stemming regimes are also problematic. But suffix arrays offer increased string searching functionality. They offer the chance to support complex patterns, such as wild card patterns. Suffix arrays are best equipped to handle and accelerate grep-style pattern matching. The speedup is achieved by spending more for memory resources with pre-calculated suffix positions.

An important side note is that many of the more complex query options can be also provided with an inverted file index. To implement this, the vocabulary is either indexed with a suffix array spanning the vocabulary strings or a secondary inverted index created. This secondary index can use character bigrams or trigrams that comprise the vocabulary terms (Zobel et al., 1993) for fuzzy search.

Both indexing techniques, inverted file index and suffix array, are efficient retrieval tools. The inverted file index is well suited for Boolean keyword search, even with large scale data sets, whereas the suffix array supports more complex patterns, but requires a large memory overhead. The later is therefore better suited for smaller tasks, such as auto-completion for the user based on a dictionary of the existing terminology. Next, we will introduce static similarity measures for ranking. Such global importance measures can be used to optimize an index, as mentioned for the inverted file index.

2.6.2 Static Similarity Measures for Ranking

Given the extensive amount of information on the World Wide Web, a typical short query of one or two keywords submitted to a search engine can easily retrieve tens of thousands web pages. Ranking the returned web pages, such that the useful ones appear in the top of the ranked list, is a critical task in the Web information retrieval (IR). In this case traditional IR content analysis and similarity measures are often not adequate. Example for traditional algorithms are the cosine similarity within the vector space model (Salton et al., 1975; Singhal et al., 1996), the Okapi BM25 and BM25F (Robertson et al., 1996), and probabilistic approaches based on language models (Manning and Schütze, 1999; Croft and Lafferty, 2003). They are not well equipped for the retrieval in this case, because the query is too short, web pages are created with varying qualities, web structure on a local site is not taken into account, and many other reasons.

This lead to the research of using information implicitly contained in the hyperlink structure of the web. Two popular ranking algorithms among the early developments are the PageRank algorithm (Page et al., 1997; Brin and Page, 1998) which is used in the search engine Google, and the Hypertext Induced Topic Selection (HITS) algorithm (Kleinberg, 1999). HITS has been used by the search engine Yahoo! in their ranking schemes. The HITS measure makes the distinction between hubs and authorities and computes them in a mutually reinforcing way. PageRank considers the hyperlink weight normalization and the equilibrium distribution of random surfers as the citation score. The ground braking idea of a HyperRank (Marchiori, 1997) had been presented in 1996 at WWW6, one year before the PageRank paper, but was never as successfully marketed as the PageRank.

Online Page Index Calculation (OPIC) is an incremental algorithm, designed to be calculated while crawling. As each new link is seen, it increments the score of the page it links to. OPIC is thus much simpler and faster to calculate than PageRank. It also provides a good approximation of PageRank, but prioritizes better when crawling than PageRank. Crawling using an incrementally calculated PageRank is not as good as OPIC, because OPIC crawls sooner the pages with higher PageRank (Abiteboul et al., 2003). The OPIC algorithm is used in the open source search engine Nutch[31].

Online Page Index Calculation (OPIC)

The OPIC algorithm is one of many variants and possible optimizations for the classic PageRank algorithm. The main idea can be described as follows. For each page or node there are two values. The first is the *cash*. This value is initialized by distributing an equal amount to each node. For n nodes this will be $\frac{1}{n}$. The cash of a node records during the calculation the recent information discovered about the page. It is the sum of the cash obtained by the page since the last time it was visited. The second value recorded is the *history* (*credit*) of the page, the sum of the cash obtained by the page since the start of the algorithm until the last time it was crawled. The cash is typically stored in main memory, whereas the history may be stored on disk.

[31] `http://nutch.apache.org/`

Now if a page i is retrieved by the crawler, there are no additional costs to retrieve the pages it points to (out links). Then the cash history is updated or created. The next step is to distribute this cash to the out links. Thereafter the cash of the page i is reset to zero. This is sufficient to create an approximation of the importance.

Detailed Algorithm The implementation of the OPIC algorithm uses two vector data structures $C[1 \ldots n]$ for the cash and $H[1 \ldots n]$ for the history. The vectors can be arbitrarily initialized. The history is, in the simplest version of OPIC, a single number. For more complex versions this can be extended to an array of fixed length per history entry. For efficiency purposes it is assumed that C can be kept in the main memory. The history H is stored on disk and each history entry can be loaded with costs of a single disk access. In order to optimize the computation of $|H| = \sum_i H[i]$, a variable G is introduced so that $G = |H|$ at each step. The algorithm for updating the cash and history is shown in Figure 2.10.

```
OPIC: On-line Page Importance Computation
 for each i let C[i] := 1/n ;
 for each i let H[i] := 0 ;
 let G:=0 ;
 do forever
 begin
   choose some node i ;
   %% each node is selected
   %% infinitely often
   H[i] += C[i];
   %% single disk access per page
   for each child j of i,
     do C[j] += C[i]/out[i] ;
   %% Distribution of cash
   %% depends on L
   G += C[i] ;
   C[i] := 0 ;
 end
```

Figure 2.10: OPIC update algorithm (Kleinberg, 1999)

After each update of the vector the new importance can be calculated for each page k with $(H[k] + C[k])/(G + 1)$. The proposed algorithm for the update does not impose any requirement on visiting order of the graph as long as each node is visited infinitely often. In general a greedy strategy (take pages with high cash values first) converges much faster, than a random approach with equal probability for each page.

If the assumption holds that the cash of children is stored in main memory, no disk access is necessary to update it. At the time a node is crawled and updated, the list of the out links children is directly available from the document. A distributed variant of the OPIC can be implemented quite easily. For this, the URLs are distributed to a cluster of machines using a hash function. Then each machine can handle and update the costs and history of its own set of pages. If cash has to be transferred to URLs not known to the machine, it uses the hash function to determine which machine is responsible. Then it transfers the cash to the machine, e.g., using a buffered network call.

The properties of the OPIC algorithm make this approach an ideal candidate for focused crawling. Because it operates in an online-fashion and the results are immediately available, the crawler can use them to focus crawling to the most relevant pages. Furthermore, since it's not required to store the link matrix, it requires less storage resources, CPU, memory and disk access than the standard algorithms. Similar OPIC is advantageous for continuous crawling as the update costs are low compared to approaches where each page is read only once.

Further approaches

Besides the three presented approaches for calculating a meaningful ranking, there exist many more proposed methods. There is the CleverRank (Wang, 2002) and the InormRank/SnormRank. They provide an option for unifying the approaches from the PageRank and HITS (Ding et al., 2002). Opposed to using linear systems as in HITS and PageRank, it is also possible to apply non-linear dynamical systems for web searching and ranking (Tsaparas, 2004).

There are time aware authority rankings with T-Rank (Berberich et al., 2004) or temporal ranking (Jatowt et al., 2005). The FlexiRank includes syntactic properties and other classifiers to improve search results for user-requested types of pages (Mukhopadhyay and Biswas, 2005). There is the work of Cho and Adams (2003) with a focus on page quality and the search for an unbiased Web ranking. It includes a quality estimator to reduce the rich-get-richer phenomenon and promote new and high-quality pages. Furthermore, Hawking et al. (2004) evaluate the usage of links from external web sites as a ranking factor for internal/local web search.

Another idea is to use machine learning for estimating a static ranking. For example, the RankNet algorithm (Richardson et al., 2006) uses features that are independent of the link structure of the Web. The RankNet features include the frequency at which users visit Web pages.

The weight of keywords may be included. The weight can be computed from the elements, keywords and anchors (K-elements). A linear combination of the hyperlink structure and the weight of keywords can be used to rank web pages (Lai et al., 2006).

2.6.3 Summary

For fast and efficient indexing, there are techniques such as the inverted files and suffix arrays. The data structures and algorithms of inverted files are best-equipped for indexing of large data sets, as they support distribution and can be implemented with the Map-Reduce-Pattern. The suffix array provides more complex query options than the inverted file, but requires more memory.

In large data sets, such as the web, there is a requirement to assess the importance of an individual web site. This score can be used for the prioritization in the crawling queue or the ranking of search results. For the web, the most well known algorithms for such a static similarity score are the PageRank and HITS. They use the network structure in form of the hyperlinks between pages to calculate the score. An open implementation of a simplified PageRank is the OPIC-algorithm and is used in the search engine Nutch.

CHAPTER 3

GOWEB – SEMANTIC WEB SEARCH FOR THE LIFE SCIENCE

The key contribution of this chapter is the combination of established keyword-based web search with semantic search technologies. The result is the semantic search engine GoWeb, which combines both approaches using ontologies and entity recognition techniques.

GoWeb is an internet search engine based on ontological background knowledge. It helps the user to browse a potentially long list of search results according to the categories provided by the used biomedical ontologies GO and MeSH. It identifies named entities, such as protein and gene names, persons, company names, and contact information in form of E-Mail and telephone numbers.

To efficiently annotate web search results with ontology concepts and named entities a new algorithm using the radix tree data structure is introduced. The system is available as a web application, which requires a scalable implementation using parallelization and distributed services. The web interface provides categories and semantic filter for fast and convenient retrieval of relevant results. For each search result the semantic filter are induced from the annotated concepts and entities and the structure of the background knowledge.

GoWeb is evaluated on the technical level with regards to the runtime and memory consumption of the annotation and entity recognition algorithms. With regard to contents the system is evaluated using three benchmarks, answering questions for three research tasks in the biomedical domain. Furthermore, the positive contribution of the semantic filtering feature and web interface are shown.

The first research problem addressed in this chapter is the combination of keyword-based search with semantic search (*Open Problem 1*). Only the combination provides the advantage of both approaches and presents the best results. Modern keyword web search engines contribute ease of use with a nearly complete and up-to-date web crawl. The semantic technologies provide the structured background knowledge for the induction of relevant results.

Web search engines are currently the most commonly used tool for search. To address the special purpose needs of life science research, appropriate background knowledge, entity recognition algorithms and names entities are selected. This addresses the *Open Problem 2* as proposed in the Motivation (Chapter 1).

3.1 Algorithm

The search is executed using the Yahoo! Search BOSS service (Yahoo! Inc.). The result of a submitted search is a list of textual extracts from web documents, called snippets. Next, GoWeb uses entity recognition techniques to annotate the snippets with concepts from the background knowledge.

The algorithm for the identification of ontological concepts in text is based on the radix tree, see Section 2.5.3 for an introduction. For the identification of protein and gene names we use the approach by Hakenberg et al. (2007), which achieved the best results in the gene identification task of BioCreAtIvE 2 (Critical Assessment of Information Extraction systems in Biology) in the year 2007. GoWeb also employs entity recognition algorithms for person and company names, and additional information such as e-mail addresses and telephone numbers. Further entity recognition services can be integrated into GoWeb. Currently the OpenCalais service (ClearForrest, 2008) can be used to identify additional entities.

3.1.1 Choosing an Algorithm for Annotation

The task of annotating snippets for an interactive web application has different requirements and parameters than traditional literature search, cf. the GoPubMed system (Doms and Schroeder, 2005), and Section 2.1.3. The input for the annotation consists of snippets, which are short, automatically extracted text blocks. Snippets can be irregularly structured or incomplete.

For GoWeb the goal is to annotate up to 1000 snippets for several entity types within sub-second response time. Furthermore, the algorithm must be able to handle large dictionaries with millions of entries. For practical reasons such as hardware requirements, it is preferable to achieve this with reasonable memory consumption.

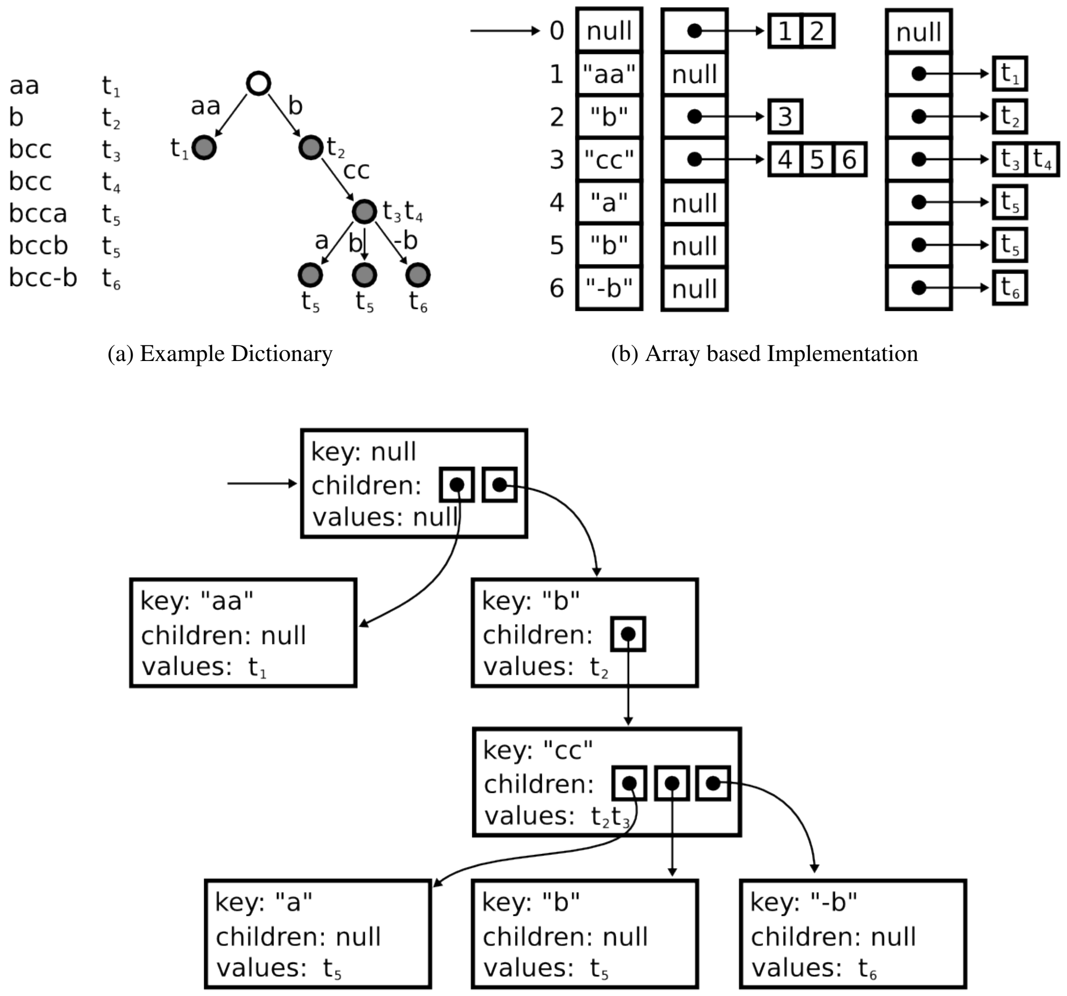
The matching of concepts and snippets is done using string-based methods. As discussed in Section 2.5.3, there are several options for string matching. The automatons are a good option, if the rules are known and the transformation of the non-deterministic automaton (NFA) to a deterministic one (DFA) can be done within main memory. For large dictionaries this is not possible. The data structure trie[1] is the fastest option, but has very high memory overhead. The radix tree is a good compromise between memory consumption and search speed. Suffix trees are not required. The additional feature to find all suffixes of a given dictionary is not required. For the matching this is even counter-productive. This drawback applies especially for the non-informative and short suffixes.

3.1.2 Implementing a Radix Tree for Annotation

The radix tree in the GoWeb system must support two operations. First, the insert operation of a text label and its associated identifier, second the search. The data structure does not need a delete operation, as the dictionaries and ontologies can be considered as static. As the GoWeb system is implemented in Java, there are two options for implementing a tree. The first is building a tree with pointers using object references. The second is using arrays. The later idea is to use an object of arrays instead of an array of objects.

The difference between the two implementations is the number of objects the information is stored. The array-based implementation is more memory efficient, as there is no overhead for class information in every node. In contrast, to access the information for a node the array-based implementation has to access three large arrays. This is not required for the case of objects, since this information is local to the object. An example for both types of implementation is shown in Figure 3.1. The example dictionary contains labels and identifiers for 6 terms $\{t_1, t_2, \ldots, t_6\}$. The

[1] trie = tree data structure of re***trie***val



(a) Example Dictionary

(b) Array based Implementation

(c) Object Tree based Implementation

Figure 3.1: Radix tree data structure example

resulting radix tree is shown in Figure 3.1a. Using the object-based approach this results in a tree with seven nodes (see Figure 3.1c), whereas the number of arrays is constant (see Figure 3.1b).

The search in both variants has to be adapted in order to accommodate longer strings. This means the search function has handle also prefixes of the query string. A prefix matching a dictionary entry is to be considered a valid match. For example, assume the dictionary as shown in Figure 3.1a, and let the query string $q =$ "bcc-b␣aa". The longest match from the dictionary is in this case term t_6 with label "bcc-b". This assumes that the space character ␣ is a delimiter between words. This is a modification from the original radix tree algorithm (see Section 2.5.3). In the case of an incomplete match for a node, it additionally checks if the next character is a delimiter. If yes, then a valid prefix is found and handled as possible result.

A second modification of the algorithm is to consider not only the longest match but also shorter matches. This allows, for instance, a simple back tracking. Given a set of two delimiters $\{␣, -\}$, there are three possible prefix matches for q: t_3, t_4 with label "bcc" and t_6 with "bcc-b". The required change is implemented in the search function. The search stores in this case a list of all matches during the traversal of the tree.

A single query using the radix tree can identify only the relevant prefixes at one starting position. But as shown in the previous example, there is at least one more possible match. For practical reasons the search must also return the starting position of a match in the query. The result is a collection of pairs $\langle t_i, pos\rangle$. To find all valid pairs, the radix tree has to be used multiple times with all suffixes from the query. To optimize this and reduce the number of suffixes, the delimiters $\{\text{␣}, -\}$ are employed. For example, with query q and all matches this will result to the following:

$$\begin{aligned}
\texttt{"bcc-b␣aa"} &\Rightarrow \{\langle t_3, 0\rangle, \langle t_4, 0\rangle, \langle t_6, 0\rangle\} \\
\texttt{"b␣aa"} &\Rightarrow \{\langle t_2, 4\rangle\} \cup \{\langle t_3, 0\rangle, \langle t_4, 0\rangle, \langle t_6, 0\rangle\} \\
\texttt{"aa"} &\Rightarrow \{\langle t_1, 6\rangle\} \cup \{\langle t_3, 0\rangle, \langle t_4, 0\rangle, \langle t_6, 0\rangle, \langle t_2, 4\rangle\} \\
&\Rightarrow \{\langle t_3, 0\rangle, \langle t_4, 0\rangle, \langle t_6, 0\rangle, \langle t_2, 4\rangle, \langle t_1, 6\rangle\}
\end{aligned}$$

Depending on the application and the used dictionary, it may not be required to extract all matches. If only the longest matches are required and no overlapping matches are allowed, a reduction of steps can be achieved. In the aforementioned example this results to the following:

$$\begin{aligned}
\texttt{"bcc-b␣aa"} &\Rightarrow \{\langle t_6, 0\rangle\} \\
\texttt{"aa"} &\Rightarrow \{\langle t_1, 6\rangle\} \cup \{\langle t_6, 0\rangle\} \\
&\Rightarrow \{\langle t_6, 0\rangle, \langle t_1, 6\rangle\}
\end{aligned}$$

With this radix tree-based search, it is possible to find dictionary entries without the explicit need for tokenization. The delimiters are used during the search and for minimizing the number of suffixes. To minimize the number of suffixes, suffixes are only used, if and only if they start directly after a delimiter or at index zero. This can be also used to skip multiple delimiters. Furthermore, suffixes shorter than the shortest entry in the dictionary can be skipped.

The time complexity during search depends solely on the length $l = |q|$ of the search string q. Additional, the number of hits h affects the runtime complexity. The overall time complexity is $\mathcal{O}(l^2 + h)$. The space complexity for radix tree is constant during the search. Only the numbers of hits h influences the space complexity during runtime.

3.1.3 Runtime Assessment for the Annotation Algorithms

GoWeb requires a fast annotation procedure. In this direction, the behavior of the speed and memory requirements is measured using a benchmark. The benchmark consists of the following parts:

- 100 iterations, with average runtime (average), minimal runtime (minimum), maximal runtime (maximum), and the memory used for the search data structure (memory)
- per iteration: 1000 search results (title and snippet) from yahoo search result
- Dictionaries: Gene Ontology and Medical Subject Headings
- Machine-Hardware: Intel Core 2 Quad CPU Q9550 @ 2.83GHz (4 Cores) with 16 GByte Ram and as operating system a Linux x86_64 Debian with a 2.6.26-2 kernel

The benchmark measures the runtime for six different implementations. The two radix implementations (see Section above), a search trie, a suffix array (cf. Section 2.6.1.2), and as comparison two additional annotators. One using an automaton and the other employing the alignment approach as used in GoPubMed (Doms, 2004).

The results of the runtime assessment are shown in Table 3.1. Additionally, the effect of using different versions of Java is given. The memory consumption listed is the memory for the data structure itself and not the whole Java process. This information was not available for the automaton and alignment-based measurements.

The results of the benchmark show the advantages and trade-offs for choosing the radix tree and implementation option. The Trie is the fastest option (minimum) but the memory consumption

	average [second]	minimum [second]	maximum [second]	memory [MByte]
Radixtree	0.054	0.053	0.109	106
Radixtree Array	0.059	0.057	0.135	75
Trie	0.052	0.031	1.966	795
Suffix Array	0.899	0.888	0.993	27
Automaton	1.203	1.065	1.467	—
Alignment	13.072	12.868	14.493	—

(a) Java version: java-1.5.0-sun-1.5.0.17

	average [second]	minimum [second]	maximum [second]	memory [MByte]
Radixtree	0.041	0.039	0.124	106
Radixtree Array	0.049	0.046	0.135	76
Trie	0.043	0.019	1.876	782
Suffix Array	0.697	0.690	0.771	27
Automaton	1.196	1.085	1.640	—
Alignment	11.884	11.661	13.032	—

(b) Java version: java-6-sun-1.6.0_16

Table 3.1: Runtime times (average, minimum, and maximum) and memory consumption for six compared annotation algorithms, using two different Java virtual machines, in a benchmark experiment that annotates search results with GO and MeSH concepts.

is very high. Compared to the array-based radix tree implementation, the trie uses 10 times as much memory for this scenario. An annotator using suffix arrays is the most memory efficient. Due to the usage of binary search in the suffix array, it is not as fast as the trie-based variants.

The selection between the two radix tree implementations entails a traditional trade-off decision between execution time and memory consumption. More precisely, the array-based implementation is 18% (7 milliseconds) slower but consumes 28% (30 MBytes) less memory. But this difference is not as large as the speedup by using a more efficient (more recent) java virtual machine.

The implementations based on the automatons and alignment are not directly comparable, as they have more expressive search patterns. Automatons offer support for full regular expressions. The alignment provides the features of gaped alignment and deletion handling with a scoring function. Their performance is indicative of the cost pertaining to upgrading into more complex algorithms for annotation. The radix tree is a good choice for large dictionaries. Some of the features of rules and patterns, such as variants, can also be incorporated into the tree-based annotators. However, in this case a major drawback pertains to the fact that these changes need to be conducted on the source code level. This is not as flexible and maintainable as rules. The main advantage of the radix tree over the automatons and alignment is the overall better scalability. This feature can be mainly attributed to fact that tries do not need the powerset construction to create a deterministic finite automaton. Compared to alignment, the radix tree offers a dictionary independent runtime complexity.

3.1.4 Description of the Entity Recognition Algorithm

The radix tree can be used to find ontology labels in text. Furthermore, it can be also used to find other entities in text. In GoWeb it is used to find person names and companies.

To identify as many person names in text as possible, a large library of names is required. For GoWeb this library is extracted from the PubMed records of author names. The advantage of these records is that they are very comprehensive for the last names. This is not the case with the first names, where many records contain them in form of initials. To complement this, a list of popular first names was added. Using both dictionaries for first and last names, an annotator was constructed. It utilizes two radix trees, one for each dictionary, and checks if a valid name is found. For this, the back-tracking feature of the radix tree is useful. For example, if a last name is also a first name. The person name annotator is a use case, where a construction of an automaton failed due to memory constraints. In contrast, both radix tree consume 410.0 MBytes in Memory for $205,872$ first names and $1,391,262$ last names. The average time for annotating 1000 snippets is about 250 ms.

In contrast to person names, the available data for companies is not that comprehensive. Some collections, especially the ones for business intelligence, are only commercially available. However, there are non-commercial collections which can be used to create a company name dictionary. For instance there is the Open Directory (DMOZ) and Freebase (Bollacker et al., 2008). The drawback is that the noise in form of very general or short and misleading names is increased in these collections. Fortunately in Freebase, which enables semantic information, the semantics can be used to minimize the noise by filtering, after using the provided additionally information. Thus, in GoWeb the company dictionary consists mainly of Freebase companies with a slightly filtered input, with $\approx\ 73,000$ labels.

The complexity in dictionary size is not as large in the case of other entities, such as E-Mail addresses and telephone numbers. Both entity types have clear patterns. These patterns can be expressed with regular expressions. As mentioned earlier automatons are an efficient way to implement regular expressions. In GoWeb we use the Brics-library (Møller, 2009). It is a very fast Java library (Mascord, 2005). For the E-Mail patterns the RFC 5322 (Resnick, 2008) is the main reference. Additionally there are variants to handle obfuscated addresses, for example `name(at)domain(dot)com`. For telephone numbers, there are patterns-based on the country specific conventions. This includes variations for country, local area codes, and grouping of number blocks.

3.1.5 Co-Occurrence-Based Filter

The identified entities and keywords are the basis for the co-occurrence-based semantic filtering mechanism of GoWeb. The filter uses the `part-of` and `is-a` relationships from GO and the tree structure of MeSH. These relations are used to induce the relevant search result for each concept from the background knowledge. The induction result is also used to select important concepts. These top concepts are selected for the entire background knowledge and for each sub category. The selection of top concepts includes the occurrence frequency, the hierarchy level and, if available, a global frequency from a pre-analyzed corpus.

3.2 Architecture

The architecture of GoWeb is a client-server model. The user web browser is a client that communicates with the web server via HTTP. The server uses a three-tier architecture with service oriented aspects. The top layer, the presentation tier, comprises the rendering of the internal data structures to HTML or similar. The logic tier includes the application logic, including the annotators and the inducer as the main task of the application. The data tier contains the ontological background knowledge and the preprocessed dictionaries such as protein and gene names. The service oriented aspect is the usage of web services, such as the Yahoo! Search BOSS service and the OpenCalais service. Additionally the architecture provides internal hooks to employ additional services for the distributed annotation. This is done for the gene and protein name annotator. The task in this case is distributed to multiple hosts.

The workflow for GoWeb can be described as follows. The user submits a query through the search form on the GoWeb website to the server. The server preprocesses the query. The preprocessing entails the extraction of application specific parts from the query and a translation to the API query syntax. Before the preprocessed query is send, a query cache is checked. If the search result is not present or up-to-date, a new search request is send to the search service. With the current maximum of 1000 documents in GoWeb, these results are not retrievable in a single search request. This is a limit introduced due to the API specific requirements, e.g., the maximum number of result returned per request. For 1000 search results and with currently 50 results per request, this leads to overall 20 requests. To minimize the response time, GoWeb sends the results to the client as soon as the search service returns the first results. The first results are then annotated, highlighted (concepts and keywords), rendered and returned to the user. The remaining results are fetched and, if allowed by the API, this is executed in parallel as a non-blocking background task. The user can then browse the first results.

The next step is to annotate all search results. Again to reduce the response time, the annotation of results has to be done in parallel. Due to the several different annotators used, parallelism can be implemented in a straight forward manner, i.e., each process executing a different annotator. To mask the longer response times of the external annotation service OpenCalais, it is executed only in the case the user does an explicit request by clicking in the tree. For the OpenCalais service the usage of a cache is mandatory. Currently the OpenCalais-API has a limit of maximal 4 concurrent requests and maximal 4 requests per second.

As next step all the annotation information is used to induce a tree representation and the top concepts of the ontological background knowledge for the submitted query (result tree). This information is rendered and sent to the user-interface using AJAX (Asynchronous JavaScript and XML) technologies through a JSON (JavaScript Object Notation) based message format to reduce the required time and bandwidth. The JavaScript updates the tree in the browser. Again to minimize the required transfer time, only the first levels are transmitted. The deeper levels are treated with a lazy approach. They are only transferred during the exploration of the induced tree by the user. An overview of the GoWeb components and the workflow is available in Figure 3.2.

If the user selects a concept in the result tree by clicking it, a request is made to update the presented documents. This includes a filtering step of the result set and a re-ranking step. For an illustration see also Figure 3.3. The new ranking is based on the found concepts, keywords and the original ranking. A distance measure between keywords and selected concept is included for the re-ranking calculation.

Once the user decides to open a web page, GoWeb highlights the page with the keywords and concepts from the background knowledge. This is done with a proxy based process. The server checks if this page is annotatable, i.e. the content is HTML-based. Next, the GoWeb server fetches the site, analyzes the content, adds the annotations, and sends the result to the user. If the content is not processable by the proxy, the user is forwarded to the original content. To minimize format and layout changes introduced through the annotation, the process tries to preserve the original formatting. Furthermore, the web site annotator uses the HTML header to set the source to the original site. This allows for an unchanged loading of linked resources, such as images. However, the enrichment of a web page with additional with content from a different source hard in nature and touches legal aspects as well. This may be seen as Cross-Site-Scripting, which is sometimes connected to non-legal activities.

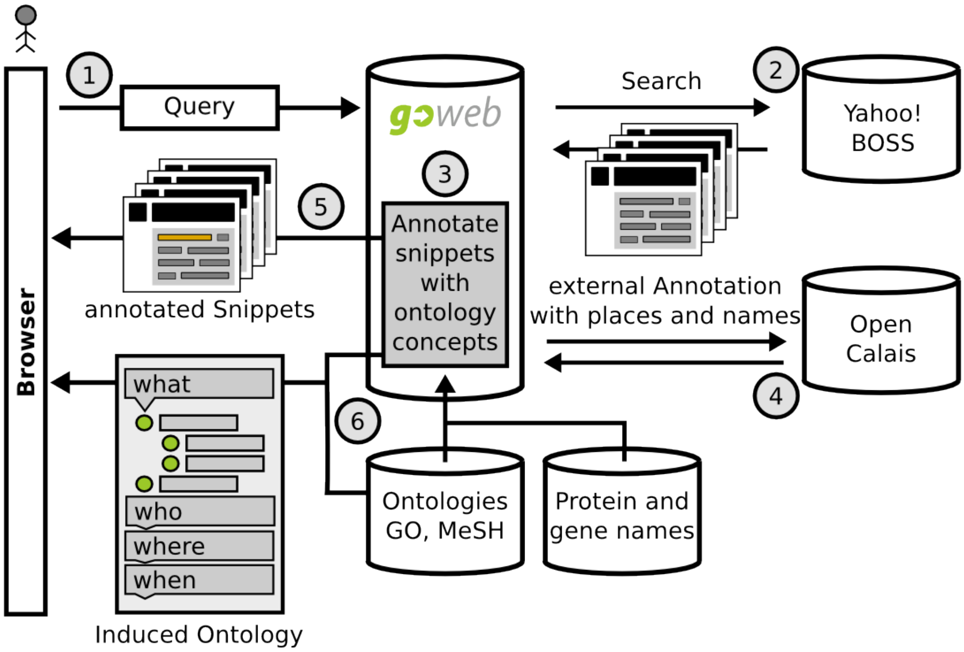


Figure 3.2: General workflow for GoWeb showing the main components and the interactions between the external services. The workflow starts with the user submitting a query via the search input field from the GoWeb page (1). The search request is parsed and transformed in a search for the external Yahoo! BOSS (2). The service returns a list of results, snippets. The textual content is annotated by GoWeb (3) and the additional external OpenCalais service (4). The search keywords and the identified entities form the annotation are highlighted in the search results. Then the results are rendered and sent to the browser (5). Based on the annotations and the ontology structure the tree representation is induced; top concepts are selected and sent to the browser (6).

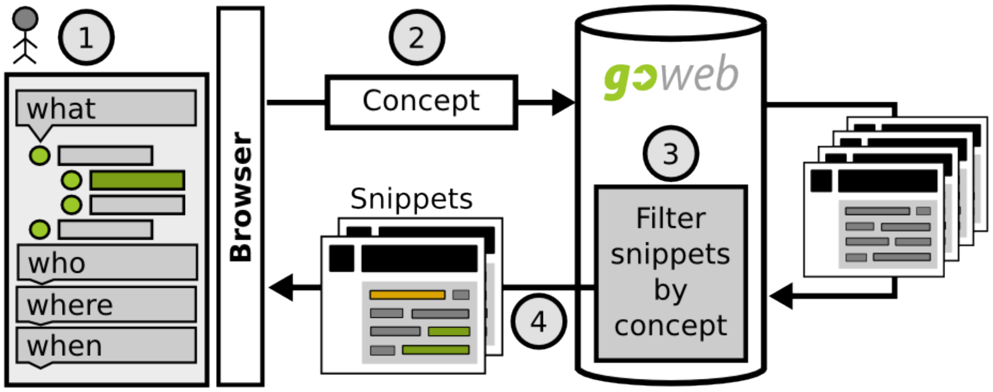


Figure 3.3: Workflow for a request containing a concept selected from the result tree in the user browser. When a user clicks on a concept in tree from the GoWeb website (1), the browser sends are request to update the search results (2). GoWeb filters all search results. If a result is annotated with the concept or an induced child of this concept, it is included in the new result list (3). Additionally the results are re-ranked and the highlighting of the selected concept and children is updated. The new result list is finally sent to the browser.

3.3 Evaluation: Answering Research Questions

The goal of GoWeb is to use ontologies and text mining in semantic web search to answer questions. Here we give some examples and evaluate the question answering capabilities of GoWeb on three benchmarks. Some typical questions of interest in the biomedical domain are for instance the following: *Which model organisms are used to study the Fgf8 protein? Which processes are osteoclasts involved in? What are common histone modifications? Which diseases are associated with wnt signaling? In which countries the chagas disease is most prominent?*

An answer to these questions can be found using GoWeb. For example Fgf8 is studied in Mice, Zebra fish; osteoclasts are involved in bone resorption; common histone modifications are Methylation and Acetylation; the wnt signaling pathway is associated with neoplasms like breast cancer, tumors or leukemia and chagas is most prominent in Brazil, Argentina, Bolivia and other Latin American countries.

The answers were directly obtained with GoWeb using simple keyword searches and the induced background knowledge. For example the answer to the first question can be found in the following way: first a user may submit the query `Fgf8`. The answer is directly shown as listed concepts in the organism's part of the background knowledge (see also Figure 3.4 and 3.5). To retrieve the corresponding search results the user may click on the organism. Answering the last question can be achieved with a very similar approach. The user may submit the query keyword `chagas` and open the *where* panel and the entities, in which information about countries resides. There is a list of identified countries, which are mostly located in Latin America. Once again a click on the country of interest in the tree retrieves the relevant articles.

The simple strategy of using keywords and filter with the induced background knowledge can be generalized to support semi-automated question answering. Next, we will demonstrate this, using three independent benchmarks. The three benchmarks use questions regarding genes and functions, symptoms and diseases, and proteins and diseases.

3.3.1 Genes and Functions

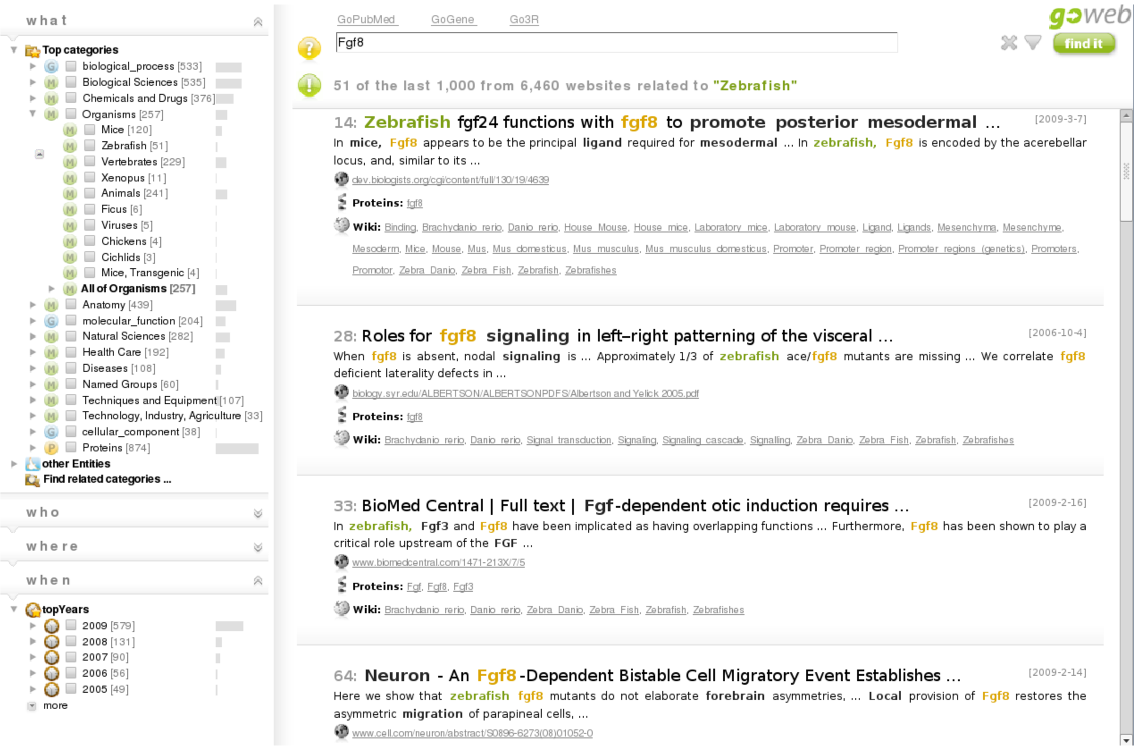
The first benchmark is based on the association of Genes and their functions. The BioCreAtIvE 1 – Task 2 (Blaschke et al., 2005) was a competition for text mining algorithms to find functional annotations in the form of GeneOntology (GO) concepts for genes in a given full-text corpus. This task is a key problem for annotators of many databases and a key question for biologists encountering a gene/protein they are not familiar with.

The test set for GoWeb currently contains all GO annotations and genes from the competition, which were identified with high confidence in the results. This yields a list of 457 gene names with a total of 1352 GO concepts. For example for "Rag C" there are 10 annotations: cytoplasm, small GTPase mediated signal transduction, RNA splicing, transcription, GDP binding, protein heterodimerization activity, small monomeric GTPase activity, heterotrimeric G-protein complex, protein binding, and nucleus.

For a test run GoWeb was given a gene name as query. Then it is checked whether the induced ontology tree contains the concepts corresponding to the expected functional annotations for the gene. For all 457 submitted names the search returned documents and the GoWeb system could identify GO concepts from these snippets. The results show that 58.1% (785 of 1352) of the benchmark concepts are contained in the tree (recall). The Top 10 concepts of the identified and not found concepts are listed in Table 3.2. As the original BioCreAtIvE task specified a corpus and not an internet wide search, as with GoWeb, precision is not applicable for this evaluation case.

3.3.2 Symptoms and Diseases

The second benchmark demonstrates the capabilities of GoWeb concerning the association of symptoms and diseases as carried out by general practitioners and medical researchers. It is based



Fgf8

51 of the last 1,000 from 6,460 websites related to "Zebrafish"

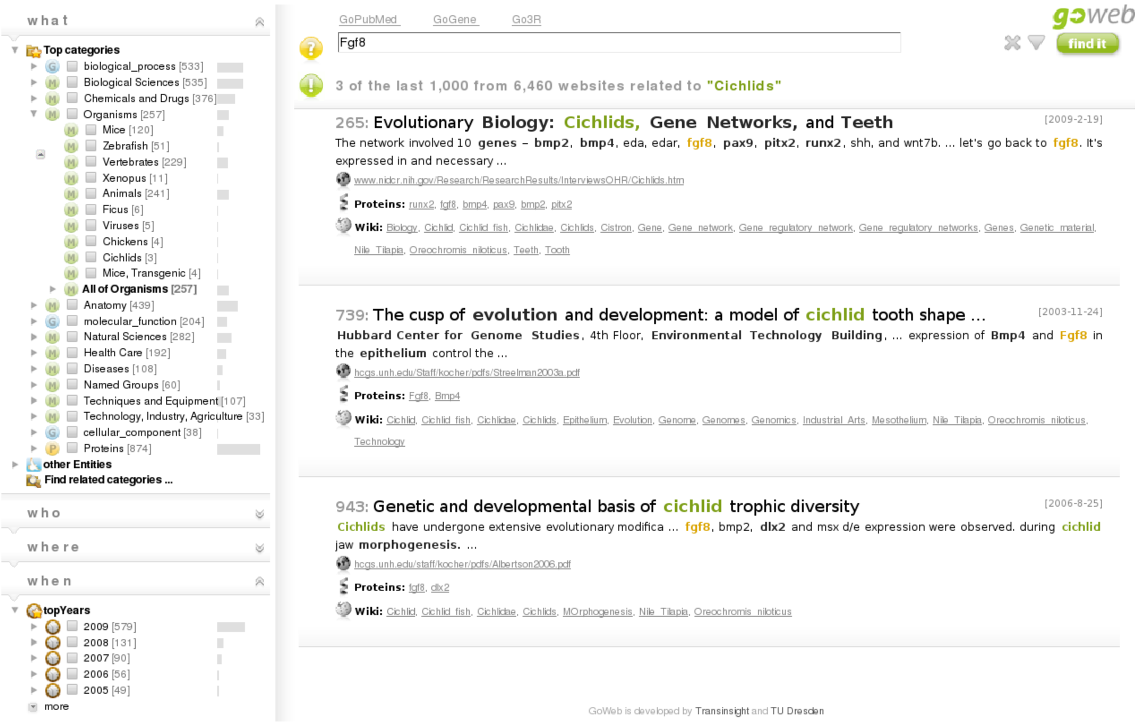
14: Zebrafish fgf24 functions with fgf8 to promote posterior mesodermal ... [2009-3-7]
In mice, Fgf8 appears to be the principal ligand required for mesodermal ... In zebrafish, Fgf8 is encoded by the acerebellar locus, and, similar to its ...
dev.biologists.org/cgi/content/full/130/19/4639
Proteins: fgf8
Wiki: Binding, Brachydanio rerio, Danio rerio, House Mouse, House mice, Laboratory mice, Laboratory mouse, Ligand, Ligands, Mesenchyma, Mesenchyme, Mesoderm, Mice, Mouse, Mus, Mus domesticus, Mus musculus, Mus musculus domesticus, Promoter, Promoter region, Promoter regions (genetics), Promoters, Promotor, Zebra Danio, Zebra Fish, Zebrafish, Zebrafishes

28: Roles for fgf8 signaling in left–right patterning of the visceral ... [2006-10-4]
When fgf8 is absent, nodal signaling is ... Approximately 1/3 of zebrafish ace/fgf8 mutants are missing ... We correlate fgf8 deficient laterality defects in ...
biology.syr.edu/ALBERTSON/ALBERTSONPDFS/Albertson and Yelick 2005.pdf
Proteins: fgf8
Wiki: Brachydanio rerio, Danio rerio, Signal transduction, Signaling, Signaling cascade, Signalling, Zebra Danio, Zebra Fish, Zebrafish, Zebrafishes

33: BioMed Central | Full text | Fgf-dependent otic induction requires ... [2009-2-16]
In zebrafish, Fgf3 and Fgf8 have been implicated as having overlapping functions ... Furthermore, Fgf8 has been shown to play a critical role upstream of the FGF ...
www.biomedcentral.com/1471-213X/7/5
Proteins: Fgf, Fgf8, Fgf3
Wiki: Brachydanio rerio, Danio rerio, Zebra Danio, Zebra Fish, Zebrafish, Zebrafishes

64: Neuron - An Fgf8-Dependent Bistable Cell Migratory Event Establishes ... [2009-2-14]
Here we show that zebrafish fgf8 mutants do not elaborate forebrain asymmetries, ... Local provision of Fgf8 restores the asymmetric migration of parapineal cells, ...
www.cell.com/neuron/abstract/S0896-6273(08)01052-0

Figure 3.4: **GoWeb Screen Shot**
GoWeb screen shot shown with example query `Fgf8` and selected concept “Zebra fish”. On the left, the semantic filters are shown, with the *where-what-who-when* panels. In the *what* panel, the GO and MeSH are shown in a tree representation. For this example the MeSH branch “Organisms” is expanded. The most relevant concepts in this branch are listed, for instance “Mice” and “Zebra fish”. The number of matching search results is given in brackets and is illustrated with a small bar chart. The bar indicates the fraction compared to the overall result set count. The wider the bar, the more often it occurs in the search result.
On the right side, the search results with the query field and the summary on top are shown. The search summary contains information about the current and overall number of search results. The individual search results are presented as a list. Each result has a title and a short text extract. In both, keywords and terms are highlighted. The number in front of the title represents the original result position.



3 of the last 1,000 from 6,460 websites related to "Cichlids"

265: Evolutionary Biology: Cichlids, Gene Networks, and Teeth [2009-2-19]
The network involved 10 genes – bmp2, bmp4, eda, edar, fgf8, pax9, pitx2, runx2, shh, and wnt7b. ... let's go back to fgf8. It's expressed in and necessary ...
www.nidcr.nih.gov/Research/ResearchResults/InterviewsOHR/Cichlids.htm
Proteins: runx2, fgf8, bmp4, pax9, bmp2, pitx2
Wiki: Biology, Cichlid, Cichlid fish, Cichlidae, Cichlids, Cistron, Gene, Gene network, Gene regulatory network, Gene regulatory networks, Genes, Genetic material, Nile Tilapia, Oreochromis niloticus, Teeth, Tooth

739: The cusp of evolution and development: a model of cichlid tooth shape ... [2003-11-24]
Hubbard Center for Genome Studies, 4th Floor, **Environmental Technology Building**, ... expression of **Bmp4** and Fgf8 in the **epithelium** control the ...
hcgs.unh.edu/Staff/kocher/pdfs/Streelman2003a.pdf
Proteins: Fgf8, Bmp4
Wiki: Cichlid, Cichlid fish, Cichlidae, Cichlids, Epithelium, Evolution, Genome, Genomes, Genomics, Industrial Arts, Mesothelium, Nile Tilapia, Oreochromis niloticus, Technology

943: Genetic and developmental basis of cichlid trophic diversity [2006-8-25]
Cichlids have undergone extensive evolutionary modifica ... fgf8, bmp2, **dlx2** and msx d/e expression were observed. during cichlid jaw **morphogenesis.** ...
hcgs.unh.edu/staff/kocher/pdfs/Albertson2006.pdf
Proteins: fgf8, dlx2
Wiki: Cichlid, Cichlid fish, Cichlidae, Cichlids, MOrphogenesis, Nile Tilapia, Oreochromis niloticus

GoWeb is developed by Transinsight and TU Dresden

Figure 3.5: **GoWeb Screen Shot - Selected Concept**
GoWeb screen shot shown with example query `Fgf8` and selected concept "Cichlids". Next to the top concept as answers there is for instance the concept "Cichlids" listed under "Organisms". When selected, GoWeb retrieves the matching snippets. For this example the result set is reduced to three articles, which were formerly on positions 265, 739 and 943. The snippets mention the usage of cichlids in the study of tooth and jaw morphogenesis. The user can learn more about cichlids through the concept definitions available in the tooltips, or the exploration of related links to Wikipedia.

Found (Top 10)			Not Found (Top 10)		
GO accession	count	term name	GO accession	count	term name
GO:0005515	90	protein binding	GO:0005624	12	membrane fraction
GO:0005634	29	nucleus	GO:0005515	7	protein binding
GO:0005737	17	cytoplasm	GO:0005653	7	perinuclear space (synonym of GO:0005641)
GO:0005488	17	binding	GO:0005524	5	ATP binding
GO:0005886	13	plasma membrane	GO:0042921	5	glucocorticoid receptor signaling pathway
GO:0016020	11	membrane	GO:0005545	5	phosphatidylinositol binding
GO:0007165	10	signal transduction	GO:0042804	5	protein homooligomerization activity (synonym of GO:0051260)
GO:0003700	10	transcription factor activity	GO:0016197	5	endosome transport
GO:0005624	9	membrane fraction	GO:0010033	4	response to organic substance
GO:0016021	8	integral to membrane	GO:0007050	4	cell cycle arrest
GO:0009986	8	cell surface	GO:0050656	4	3’-phosphoadenosine 5’-phosphosulfate binding

Table 3.2: Overview for the Top 10 found and missed GeneOntology Terms

on the study by Tang and Ng (2006), who used a set of 26 diagnostic cases published in the case records of the New England Journal of Medicine. The symptoms were used as keywords for the search. From the search results, they proposed a possible diagnosis. For example for the symptoms "fever, anterior mediastinal mass and central necrosis", they expected to find the diagnosis "Lymphoma". With their Google-based approach the proposed diagnosis was in 15 out of 26 (58%) cases correct. It also has to be remarked, that Tang and Ng is a controversial article (Taubert, 2006; Twisselmann, 2006; Wentz, 2006). One of the main issues was a possible wrong impression to the patients. It has to be clear that a search cannot replace the professional and trained diagnostic capabilities of a physician. Especially in the medical domain, web search results need to be handled with careful considerations.

In the experimental setup for GoWeb the same keywords as in the original paper were used. Each diagnosis has been mapped to the corresponding MeSH concept, if possible (see Table 3.4). During the experiment a query was given to the GoWeb system and the resulting induced background knowledge tree was evaluated. As an additional comparison for GoWeb we also applied this benchmark and experimental setup to the GoPubMed system (Doms and Schroeder, 2005).

GoWeb can provide the correct answer in 20 out of 26 (77%) cases. In 10 of these cases, the answer term is found directly in the top categories of the Diseases subtree of MeSH (see Table 3.5). With up to 10 categories per subtree, this equals to a top 10 ranking – or better – in these cases for the identified ontology concepts.

The cases 8, 10 and 18 are not marked as successful, although the results mention the searched concepts. But they all find only one article, the article Tang and Ng (2006) this analysis relies on. With GoPubMed an answer could only be found in 13 cases. GoPubMed searches only in scientific abstracts and does not include web contents such as clinical trails, general health pages, disease group pages, etc. For a comparative overview see Table 3.3.

Case	5	6	7	8	9	10	11	12	14	15	16	17	18	19	22	25
Google	√		√	√	√		√	√		√		√				
GoPubMed	√	√	√					√	√			√				√
GoWeb	√	√	√		√		√	√	√	√	√	√			√	√

Case	26	27	28	29	30	31	33	34	36	37	Count	Ratio
Google	√	√	√	√	√		√	√		√	16	62%
GoPubMed	√		√		√		√	√		√	13	50%
GoWeb	√	√	√	√	√		√	√		√	20	77%

Table 3.3: Comparison of Google, GoPubMed and GoWeb for symptoms and diseases benchmark

For example for the case study number 28 GoWeb finds 126 articles for the query "ANCA haematuria haemoptysis". Under diseases one can find the MeSH concept "Churg-Strauss Syndrome". A click on the concepts in the tree retrieves three snippets containing the concept. The resulting snippets are:

- **Laboratory imposed restrictions on ANCA testing – 63 (5): 594 – Annals of the Rheumatic Diseases**
  The laboratory has performed ANCA testing only when the request form indicated ... haematuria (requests from the renal/transplant unit), Churg-Strauss syndrome, ...
  `ard.bmj.com/cgi/content/extract/63/5/594`

- include haemoptysis (13% of patients), cystic bone lesions (4-20% of ... Some 70-75% of patients with Churg-Strauss syndrome have ANCA ...
  `hospitaldoctor.ie/hospital_doctor/pdfs/HOS_DOC_MARCH_APRIL_05.pdf`

Case	Diagnosis	Term
5	Infective endocarditis	Endocarditis, Bacterial
6	Linitis plastica with bowel obstruction	Linitis Plastica AND Intestinal Obstruction
7	Cushing secondary to adrenal adenoma	Cushing Syndrome
8	Osteoid osteoma	Osteoma, Osteoid
9	Hot tub lung secondary to M. avium	
10	Ehrlichiosis	Ehrlichiosis
11	Lymphoma	Lymphoma
12	Neurofibromatosis Type 1	Neurofibromatoses
14	Vasculitis	Vasculitis
15	Amyloid light chain	Amyloid
16	Pheochromocytoma	Pheochromocytoma
17	Acute chest syndrome	
18	Endometriosis	Endometriosis
19	Aspiration pneumonia and brain abscess (polymicrobial)	Pneumonia, Aspiration AND Brain Abscess
22	West Nile fever	West Nile Fever
25	Pylephlebitis	Phlebitis
26	HOCM	Cardiomyopathy, Hypertrophic
27	Creutzfeldt-Jakob disease (CJD)	Creutzfeldt-Jakob Syndrome
28	Churg Strauss	Churg-Strauss Syndrome
29	Dermatomyositis secondary to NHL	Dermatomyositis
30	Cat scratch disease	Cat-Scratch Disease
31	Cryoglobulinaemia	Cryoglobulinemia
33	MADH4 mutation (HTT + juvenile polyposis)	Telangiectasia, Hereditary Hemorrhagic
34	TENS	Epidermal Necrolysis, Toxic
36	MELAS	MELAS Syndrome
37	Brugada	Brugada Syndrome (Arrhythmia (1996-2006))

Table 3.4: Diagnosis and their corresponding ontology term matches

	Query	GoWeb	Count
5	Acute "Aortic regurgitation" depression abscess	Tree: Endocarditis, Bacterial	7 (1000)
6	oesophageal cancer hiccup nausea vomiting	Tree: Adenocarcinoma AND Intestinal Obstruction	2 (1000)
7	hypertension "adrenal mass"	Top categories: Cushing Syndrome	41 (1000)
8	"hip lesion" child	no, bmj article	258
9	HRCT centrilobular nodule "acute respiratory failure"	Finds the case studies this analysis relies on	15
10	fever bilateral "thigh pain" weakness	no, bmj article	500
11	fever "anterior mediastinal mass" central necrosis	Top categories: Lymphoma	66 (323)
12	multiple "spinal tumors" "skin tumors"	Top categories: Neurofibromatoses	21 (240)
14	"ulcerative colitis" "blurred vision" fever	Tree: Vascultits	2 (1000)
15	nephrotic syndrome "Bence Jones" ventricular failure	Top categories: Amyloidosis	20 (247)
16	hypertension papilledema headache "renal mass"	Tree: Pheochromocytoma	1 (31)
17	"sickle cell" pulmonary infiltrates "back pain"	Top5 snippet is ACS	1000
18	fibroma astrocytoma tumor leiomyoma scoliosis	no, bmj article	1 (47)
19	pulmonary infiltrates "cns lesion" OR "Central nervous system lesion"	no	87
22	CLL encephalitis	Tree: West Nile Fever	3 (1000)
25	"portal vein thrombosis" cancer	Tree: Phlebitis	9 (1000)
26	"cardiac arrest" exercise young	top categories: Cardiomyopathy, Hypertrophic	22 (1000)
27	ataxia confusion insomnia death	Tree: CJD	17 (1000)
28	ANCA haematuria haemoptysis	Top categories: Churg-Strauss Syndrome	3 (126)
29	myopathy neoplasia dysphagia rash periorbital swelling	Top categories: Dermatomyositis	4 (32)
30	"renal transplant" fever cat lymphadenopathy	Top categories: Cat-Scratch Disease	13 (322)
31	"buttock rash" "renal failure" edema	no	120
33	polyps telangiectasia epistaxis anemia	Top categories: Telangiectasia, Hereditary Hemorrhagic	33 (1000)
34	"bullous skin" "respiratory failure" carbamazepine	Top categories: Epidermal Necrolysis, Toxic	4 (25)
36	seizure confusion dysphasia lesions	no	1000
37	cardiac arrest sleep	Tree: Brugada Syndrome	3 (1000)

Table 3.5: Overview of the GoWeb results for the symptoms and diseases benchmark.

- **Churg-Strauss Syndrome - Patient UK**
  Pulmonary: asthma, pneumonitis and haemoptysis ... patients are perinuclear-ANCA (p-ANCA) positive (antimyeloperoxidase antibodies) ...
  `www.patient.co.uk/showdoc/40024815/`

The GoWeb system performs better than GoPubMed because the underlying search engine uses a larger repository of documents. Additionally, it can index the full text, if it is available on the web. The MEDLINE search for all PubMed based search engines, like GoPubMed, is only based on abstracts. This aligns with the fact that the MEDLINE search returns often none or only one article abstract. See Appendix A.1 for details on GoWeb searches and results, including textual evidences.

3.3.3 Proteins, Diseases and Evidences

Linking proteins and diseases is a key task for molecular biomedical researchers. The third benchmark for GoWeb is based on the questions from the TREC Genomics Track 2006 (Hersh et al., 2006). The results of TREC Genomics Track 2006 comprise a benchmark that focused on passage retrieval for question answering. It is based on full-text documents from the biomedical literature. For the year 2006 there were 28 questions. With GoWeb one can answer 22 of these 28 questions (78,6%). In 13 of these cases the semantic filter helped to reduce the result set. For a summary of all questions the reader may consult Table 3.6.

Question	160	161	162	163	164	165	166	167	168	169
Answered	√	√	√	√	√	√	√	√	√	√
Filter	√	√	√		√	√				
Question	170	171	172	173	174	175	176	177	178	179
Answered			√	√	√	√	√	√		
Filter					√		√	√		
Question	180	181	182	183	184	185	186	187	Count	
Answered	√	√	√	√		√	√		22	
Filter	√		√	√		√	√		13	

Table 3.6: Summary of TREC Genomics 2006 answering capabilities of GoWeb

For GoWeb the questions were transformed into keywords. The complete listing of questions and keywords is available in Table 3.7. A question was marked as successfully handled, if there was at least one snippet that contained a valid answer. The answers had to be available in the top 20 search results. The second aspect addressed with this benchmark was to evaluate the capabilities of the filtering feature. Filtering by background knowledge helps to reduce large results to a smaller set of relevant documents. It was marked as applied, if the answers were found by using the filtering feature. If the semantic filter feature was used, the new top 20 after filtering and re-ranking were checked. In most of the cases the re-ranking shifted a valid answer and evidence in the top 5 or better. Similar to the first benchmark, precision is not applicable. The gold-standard for the TREC Genomics Track 2006 contains only passages from the original corpus and competition. New answers, as well as answers from other sources are not comparable.

The answers for the first four questions (160–164) are shown in Table 3.8. They also demonstrate the of textual evidence GoWeb can provide as answers. For example, there is the question 160: *What is the role of PrnP in mad cow disease?* To answer it, the keyword “PrnP” is submitted to GoWeb. For the semantic filter, the MeSH concept “Encephalopathy, Bovine Spongiform” is selected; mad cow disease is a synonymous label for the concept. After the filtering, an answer to the question is available in the first part of the remaining relevant results. The given

	Question	Keywords
160	What is the role of PrnP in mad cow disease?	PrnP
161	What is the role of IDE in Alzheimer's disease?	IDE Alzheimer
162	What is the role of MMS2 in cancer?	MMS2
163	What is the role of APC (adenomatous polyposis coli) in colon cancer?	APC adenomatous polyposis coli
164	What is the role of Nurr-77 in Parkinson's disease?	Nurr-77
165	How do Cathepsin D (CTSD) and apolipoprotein E (ApoE) interactions contribute to Alzheimer's disease?	"Cathepsin D" "apolipoprotein E"
166	What is the role of Transforming growth factor-beta1 (TGF-beta1) in cerebral amyloid angiopathy (CAA)?	TGF-beta1 cerebral amyloid angiopathy
167	How does nucleoside diphosphate kinase (NM23) contribute to tumor progression?	NM23 tumor progression
168	How does BARD1 regulate BRCA1 activity?	BARD1 BRCA1
169	How does APC (adenomatous polyposis coli) protein affect actin assembly?	adenomatous polyposis coli actin assembly
170	How does COP2 contribute to CFTR export from the endoplasmic reticulum?	COP2 CFTR
171	How does Nurr-77 delete T cells before they migrate to the spleen or lymph nodes and how does this impact autoimmunity?	Nurr-77 T cell
172	How does p53 affect apoptosis?	p53 apoptosis
173	How do alpha7 nicotinic receptor subunits affect ethanol metabolism?	alpha7 nicotinic receptor ethanol
174	How does BRCA1 ubiquitinating activity contribute to cancer?	BRCA1 ubiquitinating
175	How does L2 interact with L1 to form HPV11 viral capsids?	L1 L2 HPV11
176	How does Sec61-mediated CFTR degradation contribute to cystic fibrosis?	Sec61 CFTR
177	How do Bop-Pes interactions affect cell growth?	Bop Pes cell growth
178	How do interactions between insulin-like GFs and the insulin receptor affect skin biology?	insulin-like GF insulin receptor
179	How do interactions between HNF4 and COUP-TF1 suppress liver function?	HNF4 COUP-TF1
180	How do Ret-GDNF interactions affect liver development?	Ret GDNF liver
181	How do mutations in the Huntingtin gene affect Huntington's disease?	Huntingtin gene
182	How do mutations in Sonic Hedgehog genes affect developmental disorders?	Sonic Hedgehog gene
183	How do mutations in the NM23 gene affect tracheal development?	NM23 tracheal development
184	How do mutations in the Pes gene affect cell growth?	Pes gene cell growth
185	How do mutations in the hypocretin receptor 2 gene affect narcolepsy?	hypocretin receptor 2 narcolepsy
186	How do mutations in the Presenilin-1 gene affect Alzheimer's disease?	Presenilin-1 Alzheimer
187	How do mutations in familial hemiplegic migraine type 1 (FHM1) gene affect calcium ion influx in hippocampal neurons?	FHM1 calcium neuron

Table 3.7: TREC Genomics 2006 questions and keywords

	Concept	original Pos	Evidence
160	Encephalopathy, Bovine Spongiform	**378:**	Transmissible Spongiform Encephalopathy Bovine spongiform encepalopathy (BSE) is a transmissible, … Mutations in the PRNP gene cause prion disease. … `www.answers.com/topic/spongiform-encephalopathy`
161		**1:**	Insulin-Degrading Enzyme as a Downstream Target of Insulin Receptor … effect relationship between insulin signaling and IDE upregulation. … P85) was correlated with reduced IDE in Alzheimer's disease (AD) brains and in … `alzheimer.neurology.ucla.edu/pubs/IDEzhao.pdf`
		2:	Insulin degrading enzyme - Wikipedia, the free encyclopedia 1 IDE and Alzheimer's Disease. 2 IDE Structure and Function. 3 References. 4 External links … between IDE, Aβ degradation, and Alzheimer's disease. … `en.wikipedia.org/wiki/Insulin_degrading_enzyme`
162	DNA Damage	**41:**	… concerted action of RAD5 with UBC13 and MMS2 in DNA damage repair is given by … Finally, it is shown that MMS2, like UBC13 and many other repair genes, is … `db.yeastgenome.org/cgi-bin/reference/reference.pl?dbid=S000061270`
163		**1:**	The official name of this gene is "adenomatous polyposis coli." APC is the gene's official symbol. … adenomatous polyposis - caused by mutations in the APC … `ghr.nlm.nih.gov/gene=apc`
164	Parkinson Disease	**40:**	The aetiology of idiopathic Parkinson's disease Nurr 1 was first recognised as a transcription factor that was primarily … Its close structural relation to Nur 77 led to its identification in stimulated … `www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1187126`
		132:	Concise Review: Therapeutic Strategies for Parkinson Disease Based on … nuclear related receptor 1 (Nurr-1), thereby withdrawing the cells of the cell … in the SVZ and the substantia nigra of the healthy adult rat brain [77, 98] … `stemcells.alphamedpress.org/cgi/content/full/25/2/263`
		221:	Parkinson's disease: piecing together a genetic jigsaw – Dekker et al … study decreased rapidly with later onset: 77% of patients with onset of disease … agenesis of mesencephalic dopaminergic neurons in Nurr-1 deficient mice. … `brain.oxfordjournals.org/cgi/content/full/126/8/1722`

Table 3.8: Answers for TREC Genomics 2006 questions 160 to 164

number 378 corresponds to the original position. This demonstrates that without the filter this answer would not have been found normally (Granka et al., 2004). For the question 161 the keywords were specific enough. This corresponds with the original rank of first and second position for the answers.

There are two main reasons for GoWeb not being able to find a answer for all questions in the benchmark. The first is that the question is too complex. This applies for all search engines. For GoWeb a question is too complex, if the answer is too long to be formulated in one sentence or snippet. For example the question 171 contains actually two questions. The second reason is that the question domain is not sufficiently modeled in the background knowledge. For question 178, for instance, *skin biology* has no corresponding concept and is too general to be mentioned directly in text.

3.4 Comparison of GoWeb to other Approaches

The three used benchmarks provide a basis for the evaluation of GoWeb. They demonstrate the power of the idea, but also its limitations. The starting point is the usage of snippets. This is already a limitation in terms of completeness. Snippets can be seen as an abstraction. They try to summarize facts related to the keywords. A snippet can be too short to contain very complex facts, resulting to information loss. But important information is more likely to be in the snippet, because important facts are often close to each other in the original text. Thus, it is more likely to be also contained in the snippet. The co-occurrence is used as approximation for relation extraction. One advantage of using this simpler approach is the reduced computational complexity. A proper NLP-based approach, e.g., GATE (Cunningham et al., 2002), would need more computational time. Furthermore, NLP is hindered by the unpredictable grammatical structure of snippets. With GoWeb the complete annotation for 1000 results can be done on-the-fly, with a response time of less than a second.

GoWeb might also profit from word sense disambiguation (see Section 2.5.1), but the snippets are too short to provide a reliable context for co-occurring concepts. Similar meta-data such as authors or journals is not available from the snippet. Additionally, machine-learning techniques would need to build new corpus from a previously unlabeled collection of text fragments.

The decision to use the Yahoo! BOSS API as search service was made on a technical level. The Yahoo! API allows for the most results per requests, parallel requests, unlimited queries per day and re-sorting of results. The latter is explicitly not allowed for the Google AJAX Search API in their current terms of service. The same holds also for the Microsoft Live Search API. Other APIs such as Amazon's Alexa Web Search or the Google SOAP Search API are deprecated and will be discontinued.

If more information than the snippet is required, it is necessary to fetch the web pages and analyze them. This could be done on runtime for the result set, or pre-calculated during a crawl of the internet. Both options have major drawbacks. Fetching and analyzing of web pages on-the-fly is not feasible with the requirement of a short response time. The crawling of the internet is possible, but requires a significant amount of resources in terms of hardware and bandwidth to keep the index up-to-date. This is demonstrated by the popular search engines (see also Section 2.4.2 and 2.4.3). All search engines use several data and computing centers. Although the search requests from the user are the main load, keeping the index up-to-date is an important aspect. One advantage of a separate crawl is the chance to build a semantically enhanced index. Such an enhanced index offers the option to include concepts directly into the search and not as post-processing step like GoWeb (See also Chapter 6).

Including all information from a web page will increase the recall. But it would also increase the false positives from matching errors or irrelevant parts. The false positives would also unnecessarily increase the size of an index. With the option to pre-process the information, e.g., with topic recognition or disambiguation algorithms, this can be compensated. For a specialized system with a limited number of documents and known document structure, a semantic index can be a better solution than GoWeb.

ontologies	(1) implicit through RDF, (2) GO, (3) MeSH
text mining	(4) NLP, (5) label extraction, (6) Ontology terminology, (7) biomedical entities, (8) Wikipedia terminology
type of documents	(9) RDF related, (10) web pages, (11) snippets, (12) abstracts, (13) fulltext
clustering of results	(14) RDF types, (15) extracted categories, (16) textual labels, (17) ontology, (18) answers, (19) query aspects
result type	(20) RDF resource, (21) extracted text, (22) answer, (23) snippet, (24) sentence, (25) fulltext, (26) cluster, (27) induced ontology, (28) abstract

Semantic Search Engines	structured/ unstructured	ontologies	text mining	number of documents	type of documents	clustering of results	result type	highlighting	scientifically evaluated
Swoogle	rdf	1		$\gg 10^6$	9		20		yes
SWSE	rdf	1		$\gg 10^6$	9	14	20		yes
Sindice	rdf	1		$\gg 10^6$	9		20		yes
Watson	rdf	1		$\gg 10^6$	9		20		yes
Falcons	rdf	1		$\gg 10^6$	9	14	20	yes	yes
CORESE	rdf	1		$\gg 10^6$	9		20		yes
WikiDB	rdf	1		$\gg 10^6$	9		20		
Hakia	txt		4	$\gg 10^9$	10	15	21	yes	
START	txt		4	$\gg 10^9$	10		22		yes
Ask.com	txt		4	$\gg 10^9$	10		23		
BrainBoost	txt		4	$\gg 10^9$	10		24	yes	
AnswerBus	txt		4	$\gg 10^9$	10		25	yes	
Cuil	txt		4,8	$\gg 10^9$	10	15	21	yes	
Clusty	txt		5	$\gg 10^9$	10	16	23,26	yes	
Carrot	txt		5	$\gg 10^9$	11	16	23,26	yes	yes
PowerSet	wiki		4,8	$\gg 10^6$	10	15	23,25	yes	
QuAliM	wiki/txt		4,8	$\gg 10^6$	11,10		22		yes
GoWeb	txt	2,3	6,7,8	$\gg 10^9$	11	17	23,27	yes	yes
askMedline	xml	3		$\gg 10^6$	12		28		yes
EAGLi	xml	2	4,6	$\gg 10^6$	12	18	22,28	yes	yes
GoPubMed	xml	2,3	6,7,8	$\gg 10^6$	12	17	23,27,28	yes	yes
ClusterMed	xml	3	5	$\gg 10^6$	12	16	26,28	yes	yes
IHop	xml	3	6,7	$\gg 10^6$	12	19	24,28	yes	yes
EBIMed	xml	2,3	6,7	$\gg 10^6$	12	17	24,27	yes	yes
XplorMed	xml	3	5,6	$\gg 10^6$	12	17	21,28	yes	yes
Textpresso	xml	2	6	$\gg 10^6$	13	17	28	yes	yes
Chilibot	xml		7	$\gg 10^6$	12		24	yes	yes

Table 3.9: Comparison of semantic search engines in addition with GoWeb

The application of text mining for concept identification is important for finding the relevant snippets in the search results. A simple keyword can not easily replace the additional information from the background knowledge. This includes synonymous labels and related concepts. For example for heart diseases in MeSH there are over 570 related labels. Although a user can try to emulate this behavior by using long Boolean queries, there is a prerequisite. The user has to know them before-hand. This expert knowledge is compressed and available by using ontology concepts.

The types of questions handled best by GoWeb have to be transformable into keywords and concepts. The answer provided by GoWeb will be either an inferred concept or a sentence/short text extract in the snippet. These options reduce the types of questions which can be answered by GoWeb. For example in a question classification by Tomuro and Lytinen (2001) GoWeb performs best with questions of type definition ('What does X do?'), reference ('What', 'Which') or entities ('Who'). But it can not answer question types like manner of action, degree or interval ('how many', 'how much' or 'how long', e.g. What percentage of children are vaccinated?) and procedure ('how to'). For a medical question taxonomy by Ely et al. (2000), GoWeb works with questions related to diagnosis branch, but it fails for questions from the treatment branch (What are the options for treatment of condition y in situation z?), management (What is the best way to discuss or approach discussion of difficult issue x?) and nonclinical (What are the legal considerations in situation y?). For more details on questions and more question types, the reader may wish to consult Section 2.2.1.

In comparison to existing systems which are mainly focusing on searching OWL and RDF content (e.g., Swoogle, SWSE) GoWeb covers a broader area. Current RDF search engines cover millions of RDF statements, whereas the internet search engines cover billions of websites. Unfortunately, most of the information in websites is unstructured text. GoWeb tries to bridge the semantic gap with the limited amount of available semantic annotations by employing text mining for extraction of ontology concepts from text. In a nutshell, GoWeb exploits that keywords and ontology terms co-occurring in snippets are often facts.

Traditional search engines like Google have great coverage but they miss the explicit usage of ontological background knowledge. They only present a long list of results. This works very well for simple retrieval of documents, but has limits for more complex task, e.g. answering questions. The semantic filtering with concepts as in GoWeb helps to reduce the result list to relevant answers. If a snippet does not contain the relevant terms, it is likely to be not relevant.

In comparison to other internet search based systems like Hakia or PowerSet the advantage of GoWeb is its additional background knowledge for the biomedical domain. Only GoWeb combines the usage of the GeneOntology (GO), the Medical Subject Heading (MeSH) and protein identification. A clustering of text labels like Clusty or Carrot can not replace the structural knowledge of an ontology. In comparison to PubMed-based systems GoWeb can index the additional resources of full text articles on the web. Table 3.9 summarizes a number of semantic search engines and their features compared to GoWeb.

The search interface of GoWeb provides with the *what-where-who-when* categories a simple way to browse the results. Next to the actual search results GoWeb offers additional information, like definitions of concepts or Wikipedia links. Together with the filtering mechanism to reduce the result set from 1000 possible results to a small number of relevant entries GoWeb offers a powerful tool for semantic search in the biomedical domain.

3.5 Summary

GoWeb is a new web search engine using semantic technologies for the biomedical domain. It is the only semantic search engines that combines keyword-based web search with ontologies-based text mining using GO and MeSH, including biomedical named entity recognition.

GoWeb presents a solution to the **Open Problem 1** by combining the results of a classical search engine with the annotation process to match concepts from the ontological background knowledge. This is done in a post-processing step and the textual content of snippets can still be a good approximation of facts.

In response to **Open Problem 2**, GoWeb provides a web search interface. The free web application is online available[2]. The annotation algorithms reflect the time constraints of sub-second runtime for interactive application behavior. The adaption of the radix tree allows to annotate the snippets very fast with concepts and entities from the biomedical background knowledge. Furthermore, the radix-tree-based annotation algorithm can easily handle the truncation and irregular grammatical structure of snippets.

The GoWeb features, including the semantic filtering in connection with the chosen ontological background knowledge of GO and MeSH, are geared toward the biomedical domain and life science. The provided concept definitions, links for identified proteins, or recommended Wikipedia articles help to serve the information need of a researcher. The evaluation using the three benchmarks shows that GoWeb can aid to answer questions. In total the system provides speed and scalability, and it can still achieve success rates of up to 79%.

As future work, an improvement for the semantic annotation of GoWeb could be the development of a word sense disambiguation suitable for this application. In contrast to existing approaches of word sense disambiguation, it has to be addressed, that any implemented approach has to work with irregular and short text fragments. Currently, there is no such corpus for training of machine learning approaches. The application of word sense disambiguation can be applied for the annotation with ontology concepts but also for entity recognition. There the disambiguation can be applied for name with common words or to distinguish between different entity types with equal names, e.g., person name or company.

A further improvement could be a user-based customization. This can include the option to use additional custom background knowledge for their specialized research domain. Also, the customization of the result page with a selected set of ontology categories is a possibility.

[2] http://gopubmed.org/goweb

CHAPTER 4

SEMANTIC BROWSING

Semantic search as presented with GoWeb in the previous chapter is only one aspect for exploring the knowledge space. Another facet is semantic browsing. The goal of semantic browsing is to connect the current normal browsing experience with additional resources using semantic technologies. Semantic browsing enabled tools are intended to connect existing textual content to eScience infrastructure. Such tools connect the eScience services such as Web/Grid Services and its XML-based standards and ontologies to textual resources, see also Chapter 2.4.3.2.

In the case of semantic web browsers (SWB) this means the browsers identify textual labels in the pages, available from the background knowledge, e.g. as provided by ontologies. Based on the identified concepts, semantic hyperlinks are offered. In general the service module helps to facilitate the matching of concepts and entities to available services. In particular for the biomedical domain, entities such as proteins or genes are often the starting point for further investigations (*Open Problem 2*).

This chapter describes semantic hyperlinks and semantic web services as basis for a semantic web browser. The implementation options for semantic hyperlinks are presented in two systems. The first is a direct integration as in GoPubMed-Extended and a post-publishing approach for existing content. For the later, the implementation using a light-weight web proxy and JavaScript is described and compared to existing implementation approaches. An evaluation with a usability study is carried out to show to advantages of semantic browsing.

4.1 Semantic Hyperlinks

The basis for navigating the web are hyperlinks. They are embedded in the web pages. The idea of semantic hyperlinks is to provide a semantic extension to a normal hyperlink. Such semantic links facilitate the connection of resources due to the provided semantic annotation or typing of links. This allows to extend or provide new semantic hyperlinks directly to databases or web services. The target can be either a web interface, or the results pages of services.

The starting points for semantic hyperlinks are entities with semantic annotations. In case of websites, this can be provided with text mining techniques as discussed in Section 2.5 and 3.1. Depending on the concept type, there are several options to present a semantic hyperlink. Such semantic hyperlinks can have different levels of complexity and presentation options:

- Offer background information related to the entity, e.g. definitions or related concepts

- Provide a web service call for retrieving the sequence or protein 3D-structure in case of a protein identifier or name.

The results of such services may be used as input for further exploring. Ideally the results are enriched with additional semantic hyperlinks.

The additional layer of semantic hyperlinks can provide knowledge not known to the original content provider. It is an approach of dynamic linking. It is driven by the definitions and structure of the background knowledge, such as ontologies (Bechhofer et al., 2008). As discussed by Sutherland et al. (2008) semantic hyperlinks provide a good entry point for bioinformatics-related web services. The requirements and uses cases of semantic browsing have also been documented by Tvarozek and Bielikova (2009). They focus on a streamlined user interface for integrating semantic web resources.

4.1.1 Semantic Web Services

Web services are the basis for providing additional content and functionality. See Section 2.4.3.2 for more details on web services implementations, descriptions and standards. Currently, most services provide a textual description. It is usually not intended to be machine readable, but they are rather offered for a human user. Furthermore, the details might be unclear, the type, the default values of parameters, or the return value.

To solve this problem, semantic web service are a possible approach. They offer additionally typing information for the input, performed operation and possible return types. This information allows to turn a web service from "text in and text out (garbage in, garbage out)" into typed input with known functionality and typed output.

But, as with providing semantically annotated content for the web, there is a bottle neck in the annotation of web services. There are many existing web services, but very few of them provide semantic markups. For providing the semantic markup of services there are two important questions:

1. How are the descriptions provided?

2. What kind of descriptions are used?

For the format of providing such annotations for web services there are several standards. There is for instance the OWL-S or the Semantic Annotations for WSDL and XML Schema (SAWSDL). For details see also background Section 2.4.3.3.

After deciding the format, the next level is the choice of an appropriate controlled vocabulary or ontology. In the ideal case all services are annotated with the same vocabulary. If this is not possible, a mapping to existing vocabularies is desirable. The re-usage is necessary to make use of existing semantic annotations. The mapping of ontologies, with different vocabulary and modeling ideas, e.g., level of structure is an open research topic (Lambrix and Tan, 2008; Wächter et al., 2006).

The choice of vocabulary also influences the expressiveness of the supported service descriptions. But also the modeling and the domain have requirements with regards to the capabilities of the language and formal semantics. For instance, the typical bioinformatics task of multiple sequence alignment requires a list of objects, or functions with variable number of parameters. If neither is available, a direct modeling is not possible. Similarly, for the result of BLAST sequence search a list type as result is required.

After these defining questions have been answered or a decision regarding to an existing standard or software solution has been taken, the services can be finally used. For practical reasons, it is useful to either provide a service repository, or publish the service to an existing registry. There is, for instance, the BioCatalogue (Belhajjame et al., 2008b), BioMoby (Wilkinson and Links, 2002) or the FUSION Semantic Registry (Kourtesis et al., 2007) available as web service repositories.

These issues have also been addressed as part of the sealife project (Schroeder et al., 2006). There the task of creating a controlled vocabulary for the semantic annotation of bioinformatics services was addressed by Afzal et al. (2008). An existing large effort, in particular for bioinformatics web services, is the myGrid Project[1]. This UK-based project provides also an ontology for service description and annotation. Wolstencroft et al. (2007) show how this ontology can be applied to discover relevant services. Another approach to annotate web services is proposed by Giantsiou et al. (2009). They use a three step process to discover and annotate web services. At the end they offer an RDF repository with search functionalities.

An additional perspective on semantic web services is introduced by agent technologies. Agents are also a technology to find and process information. A review about agents and multi-agent systems in bioinformatics is available by Merelli et al. (2007). This includes for instance rule-based agents, e.g., implemented using RuleML (Paschke et al., 2007). Agents are an option to implement complex search strategies (Weber et al., 2009; Blacoe et al., 2010). Agents are also helpful in basic tasks, such as service discovery and service status checks. Agents can also offer their results and services as web service. They complement the traditional information sources with web services, e.g., databases.

4.1.2 Compositions of Web Services

A single web service is often only a building block in a longer pipeline or workflow. The composition of multiple services into an information extraction pipeline is a technique often required in bioinformatics. The service calls allow the integration of multiple, sometimes heterogeneous data sources. Defined and automated workflows allow to simplify and standardize the re-occurring research tasks.

This need has been addressed by various user tools. There is the TAVERNA project with its workbench[2] (Hull et al., 2006), Bio-jETI (Lamprecht et al., 2009), Kepler (Altintas et al., 2004), Grid Workflow Execution Service (GWES) [3] (Neubauer et al., 2006). There are options to use a semi-automatic composition of web services by combining BioMoby and TAVERNA (DiBernardo et al., 2008). Workflows defined by users can also be shared, for instance using the web platform myExperiment[4] (De Roure et al., 2009). Furthermore, the information encoded in user created existing workflows can also be used to annotate web services (Belhajjame et al., 2008a).

If multiple resources are integrated, the probability of inconsistencies increases. This inconsistencies may results to input or conversion errors, which may also be introduced by differences in the weighting of scientific credibility. To automatically solve such conflicts, there is for instance the idea of automated argumentation. It has been used to tackle inconsistency and incompleteness in online distributed life science resources (McLeod and Burger, 2007). It can also be employed during the composition of web services as demonstrated by Sutherland et al. (2009).

The created workflows or pipelines are resources to be provided via semantic hyperlinks. They enable users to reuse existing knowledge, on how to address or solve scientific questions. Thus, they allow to share the implicit knowledge encoded in workflows.

4.1.3 Semantic Hyperlinks in GoPubMed Extended

To demonstrate the possibilities in the aforementioned directions, the GoPubMed system (Doms and Schroeder, 2005) was extended to include semantic hyperlinks. The selection of semantic links keeps in mind use cases with three application scenarios in evidence-based medicine, literature and patent mining, and molecular biology. They all relate to the study of infectious diseases. To address the three use cases GoPubMed Extended contains the following additional semantic hyperlinks links:

[1] `http://www.mygrid.org.uk`

[2] `http://www.taverna.org.uk/`

[3] `http://www.gridworkflow.org/kwfgrid/gwes/docs/`

[4] `http://www.myexperiment.org/`

Wikipedia	Links to relevant Wikipedia web pages
Proteins	Links to Uniprot entries of identified protein names
PubChem	Links to PubChem database for matching substances and compounds
HMMer	HMMerThread database of weakly conserved domains
RiDDLE	RNAi by DEQOR-designed lookup of esiRNAs (Kittler et al., 2007)
KEGG	Kyoto Encyclopedia of Genes and Genomes (Kanehisa et al., 2008)
OMIM	Online Mendelian Inheritance in Man, OMIM is a comprehensive, authoritative, and timely compendium of human genes and genetic phenotypes (McK).
GO2Human	Composed semantic web service for anatomical integration

The basis for providing these links are the text mined entities from the PubMed abstracts. In the case of the databases, e.g., Wikipedia, PubChem, or KEGG, the identified concepts are matched against the entries. Only matching resources are considered. This helps to reduce the number of provided links. There the goal is to reduce the complexity for the user. Matching documents to Wikipedia entries has also been proposed by Mihalcea and Csomai (2007).

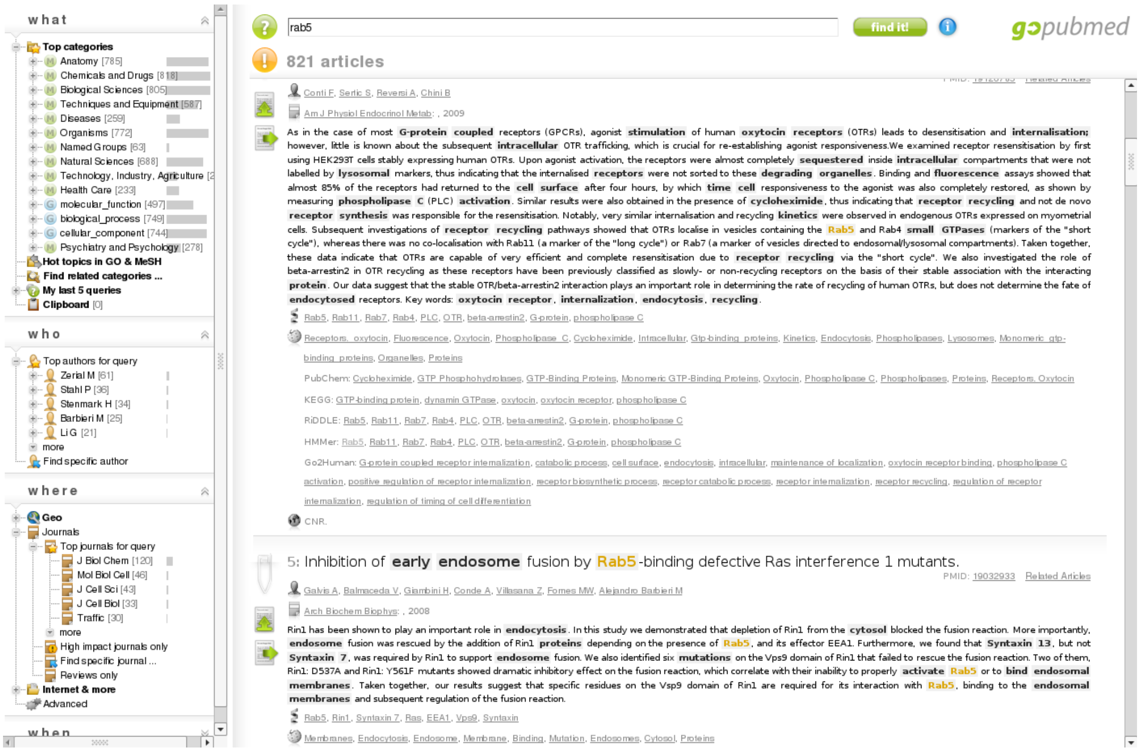
For web services, such as the HMMer and RIDDLE, the type of an entity is the matching criteria. For both web services the type Gene or Protein is required. Similarly, the composed web service is provided for corresponding GeneOntology concepts. A screen shot visualizing the GoPubMed Extended and the semantic hyperlinks for an abstract is available in Figure 4.1a and 4.1b.

4.1.4 Website Annotation with GoWeb

Semantic hyperlinks are an additional layer of hyperlinks. As they are usually not integrated in the original website, a system is implemented which enriches the original website. Such a system should allow the user to highlight relevant entities and follow further links to relevant resources. This is illustrated in Figure 4.2. There the system highlights for instance a protein name. This allows the user a direct lookup of the UniProtKB/SwissProt entry and protein sequence.

For GoWeb this has been implemented. The user sees the highlight of keywords and concepts from the background knowledge not only on the search results (see also Section 3), but also in the linked web page. For a screen shot of this feature see Figure 4.3. The induced ontology tree offers the option to change the highlight (turn on or off) for different branches of the background knowledge. To change the highlighted concepts, a simple selection by clicking of the concept is sufficient. The induced ontology tree also offers additional information, such as concept definitions, via the tools tips.

For the highlighting of concepts in websites and enriching it with additional semantic hyperlinks, there are also other tools available. There are the Conceptual Open Hypermedia Service (COHSE) (Bechhofer et al., 2006; Yesilada et al., 2006), Conceptual Resource Search Engine (CORESE) systems (Dieng-Kuntz and Corby, 2005) and the Semantic Web browser: SemWeB (Şah et al., 2010). COHSE uses a portal-based implementation allowing a more personalized ex-



Conti F, Sertic S, Reversi A, Chini B

Am J Physiol Endocrinol Metab: , 2009

As in the case of most **G-protein coupled** receptors (GPCRs), agonist **stimulation** of human **oxytocin receptors** (OTRs) leads to desensitisation and **internalisation;** however, little is known about the subsequent **intracellular** OTR trafficking, which is crucial for re-establishing agonist responsiveness.We examined receptor resensitisation by first using HEK293T cells stably expressing human OTRs. Upon agonist activation, the receptors were almost completely **sequestered** inside **intracellular** compartments that were not labelled by **lysosomal** markers, thus indicating that the internalised **receptors** were not sorted to these **degrading organelles**. Binding and **fluorescence** assays showed that almost 85% of the receptors had returned to the **cell surface** after four hours, by which **time cell** responsiveness to the agonist was also completely restored, as shown by measuring **phospholipase C** (PLC) **activation**. Similar results were also obtained in the presence of **cycloheximide**, thus indicating that **receptor recycling** and not de novo **receptor synthesis** was responsible for the resensitisation. Notably, very similar internalisation and recycling **kinetics** were observed in endogenous OTRs expressed on myometrial cells. Subsequent investigations of **receptor recycling** pathways showed that OTRs localise in vesicles containing the **Rab5** and Rab4 **small GTPases** (markers of the "short cycle"), whereas there was no co-localisation with Rab11 (a marker of the "long cycle") or Rab7 (a marker of vesicles directed to endosomal/lysosomal compartments). Taken together, these data indicate that OTRs are capable of very efficient and complete resensitisation due to **receptor recycling** via the "short cycle". We also investigated the role of beta-arrestin2 in OTR recycling as these receptors have been previously classified as slowly- or non-recycling receptors on the basis of their stable association with the interacting **protein**. Our data suggest that the stable OTR/beta-arrestin2 interaction plays an important role in determining the rate of recycling of human OTRs, but does not determine the fate of **endocytosed** receptors. Key words: **oxytocin receptor**, **internalization**, **endocytosis**, **recycling**.

Rab5, Rab11, Rab7, Rab4, PLC, OTR, beta-arrestin2, G-protein, phospholipase C

Receptors, oxytocin, Fluorescence, Oxytocin, Phospholipase C, Cycloheximide, Intracellular, Gtp-binding proteins, Kinetics, Endocytosis, Phospholipases, Lysosomes, Monomeric gtp-binding proteins, Organelles, Proteins

PubChem: Cycloheximide, GTP Phosphohydrolases, GTP-Binding Proteins, Monomeric GTP-Binding Proteins, Oxytocin, Phospholipase C, Phospholipases, Proteins, Receptors, Oxytocin

KEGG: GTP-binding protein, dynamin GTPase, oxytocin, oxytocin receptor, phospholipase C

RiDDLE: Rab5, Rab11, Rab7, Rab4, PLC, OTR, beta-arrestin2, G-protein, phospholipase C

HMMer: Rab5, Rab11, Rab7, Rab4, PLC, OTR, beta-arrestin2, G-protein, phospholipase C

Go2Human: G-protein coupled receptor internalization, catabolic process, cell surface, endocytosis, intracellular, maintenance of localization, oxytocin receptor binding, phospholipase C activation, positive regulation of receptor internalization, receptor biosynthetic process, receptor catabolic process, receptor internalization, receptor recycling, regulation of receptor internalization, regulation of timing of cell differentiation

CNR.

5: Inhibition of **early endosome** fusion by **Rab5**-binding defective Ras interference 1 mutants.

PMID: 19032933 Related Articles

Galvis A, Balmaceda V, Giambini H, Conde A, Villasana Z, Fornes MW, Alejandro Barbieri M

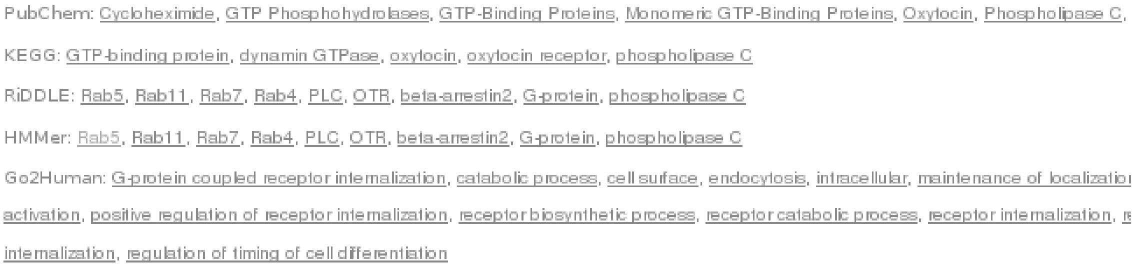
Arch Biochem Biophys: , 2008

Rin1 has been shown to play an important role in **endocytosis**. In this study we demonstrated that depletion of Rin1 from the **cytosol** blocked the fusion reaction. More importantly, **endosome** fusion was rescued by the addition of Rin1 **proteins** depending on the presence of **Rab5**, and its effector EEA1. Furthermore, we found that **Syntaxin 13**, but not **Syntaxin 7**, was required by Rin1 to support **endosome** fusion. We also identified six **mutations** on the Vps9 domain of Rin1 that failed to rescue the fusion reaction. Two of them, Rin1: D537A and Rin1: Y561F mutants showed dramatic inhibitory effect on the fusion reaction, which correlate with their inability to properly **activate Rab5** or to **bind endosomal membranes**. Taken together, our results suggest that specific residues on the Vsp9 domain of Rin1 are required for its interaction with **Rab5**, binding to the **endosomal membranes** and subsequent regulation of the fusion reaction.

Rab5, Rin1, Syntaxin 7, Ras, EEA1, Vps9, Syntaxin

Membranes, Endocytosis, Endosome, Membrane, Binding, Mutation, Endosomes, Cytosol, Proteins

(a) Full Screen Shot of GoPubMed Extended



PubChem: Cycloheximide, GTP Phosphohydrolases, GTP-Binding Proteins, Monomeric GTP-Binding Proteins, Oxytocin, Phospholipase C,

KEGG: GTP-binding protein, dynamin GTPase, oxytocin, oxytocin receptor, phospholipase C

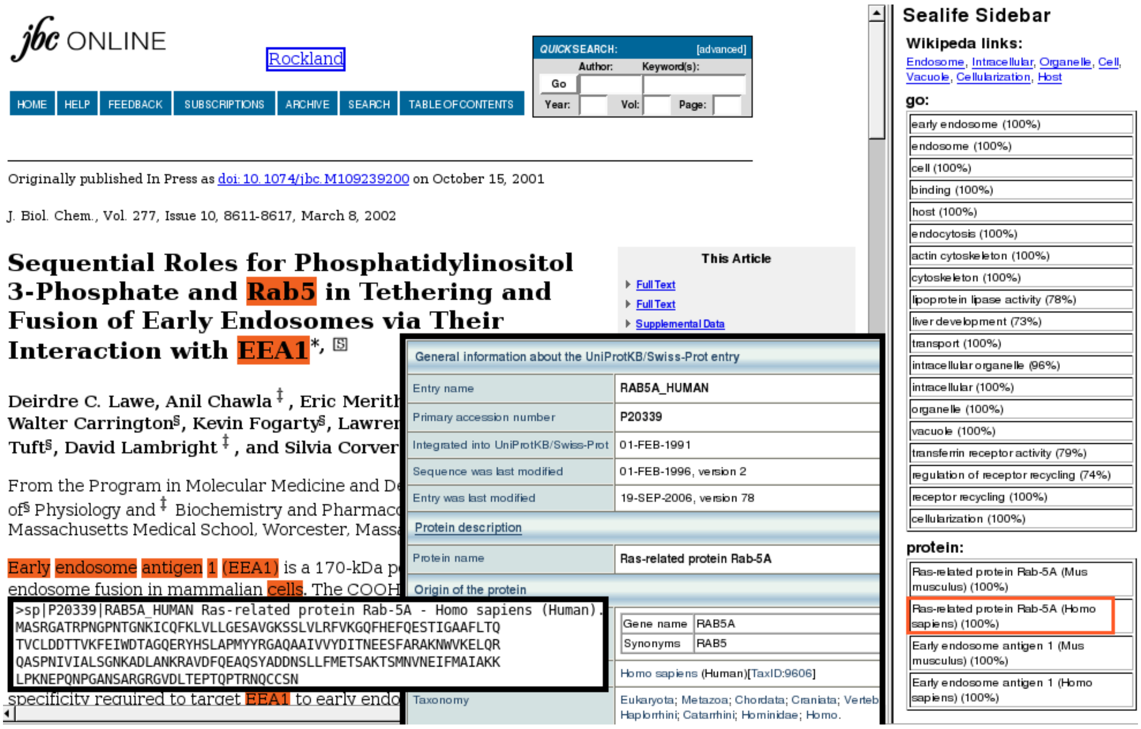
RiDDLE: Rab5, Rab11, Rab7, Rab4, PLC, OTR, beta-arrestin2, G-protein, phospholipase C

HMMer: Rab5, Rab11, Rab7, Rab4, PLC, OTR, beta-arrestin2, G-protein, phospholipase C

Go2Human: G-protein coupled receptor internalization, catabolic process, cell surface, endocytosis, intracellular, maintenance of localizatio
activation, positive regulation of receptor internalization, receptor biosynthetic process, receptor catabolic process, receptor internalization, r
internalization, regulation of timing of cell differentiation

(b) Semantic Hyper Links to PubChem, KEGG, RIDDLE, HMMer and composite web service Go2Human

Figure 4.1: GoPubMed Extended Screenshot



Originally published In Press as doi: 10.1074/jbc.M109239200 on October 15, 2001

J. Biol. Chem., Vol. 277, Issue 10, 8611-8617, March 8, 2002

Sequential Roles for Phosphatidylinositol 3-Phosphate and Rab5 in Tethering and Fusion of Early Endosomes via Their Interaction with EEA1*, S

Deirdre C. Lawe, Anil Chawla‡, Eric Merith
Walter Carrington§, Kevin Fogarty§, Lawren
Tuft§, David Lambright‡, and Silvia Corver

From the Program in Molecular Medicine and De
of§ Physiology and ‡ Biochemistry and Pharmaco
Massachusetts Medical School, Worcester, Massa

Early endosome antigen 1 (EEA1) is a 170-kDa p
endosome fusion in mammalian cells. The COOH

specificity required to target EEA1 to early endo

General information about the UniProtKB/Swiss-Prot entry	
Entry name	RAB5A_HUMAN
Primary accession number	P20339
Integrated into UniProtKB/Swiss-Prot	01-FEB-1991
Sequence was last modified	01-FEB-1996, version 2
Entry was last modified	19-SEP-2006, version 78
Protein description	
Protein name	Ras-related protein Rab-5A
Origin of the protein	
	Gene name RAB5A Synonyms RAB5
	Homo sapiens (Human)[TaxID:9606]
Taxonomy	Eukaryota; Metazoa; Chordata; Craniata; Verteb Haplorrhini; Catarrhini; Hominidae; Homo.

```
>sp|P20339|RAB5A_HUMAN Ras-related protein Rab-5A - Homo sapiens (Human).
MASRGATRPNGPNTGNKICQFKLVLLGESAVGKSSLVLRFVKGQFHEFQESTIGAAFLTQ
TVCLDDTTVKFEIWDTAGQERYHSLAPMYYRGAQAAIVVYDITNEESFARAKNWVKELQR
QASPNIVIALSGNKADLANKRAVDFQEAQSYADDNSLLFMETSAKTSMNVNEIFMAIAKK
LPKNEPQNPGANSARGRGVDLTEPTQPTRNQCCSN
```

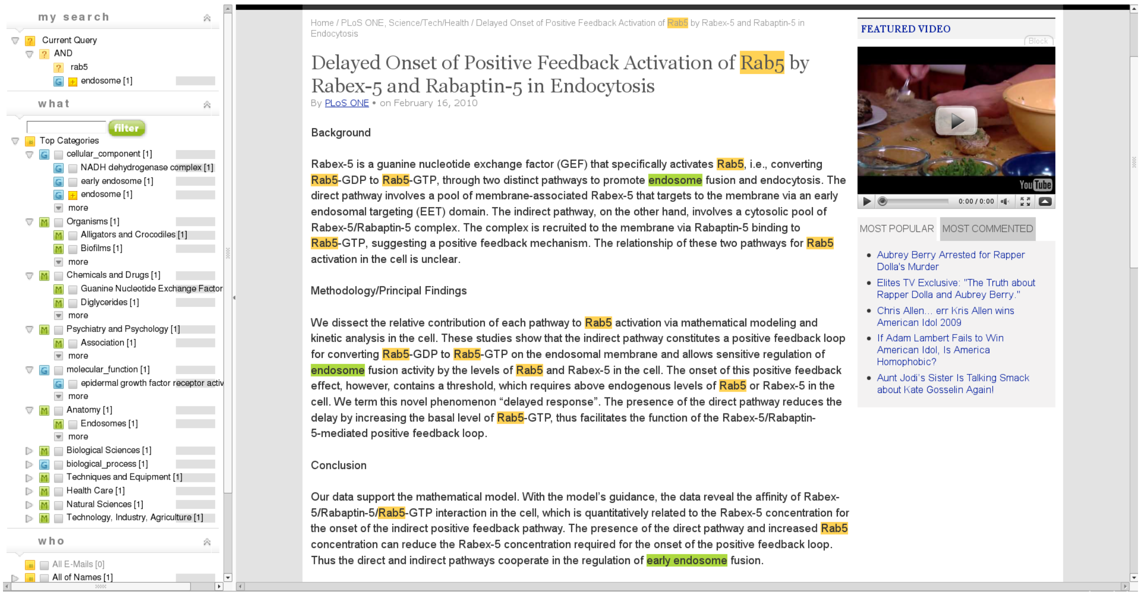
Sealife Sidebar

Wikipeda links:
Endosome, Intracellular, Organelle, Cell, Vacuole, Cellularization, Host

go:
- early endosome (100%)
- endosome (100%)
- cell (100%)
- binding (100%)
- host (100%)
- endocytosis (100%)
- actin cytoskeleton (100%)
- cytoskeleton (100%)
- lipoprotein lipase activity (78%)
- liver development (73%)
- transport (100%)
- intracellular organelle (96%)
- intracellular (100%)
- organelle (100%)
- vacuole (100%)
- transferrin receptor activity (79%)
- regulation of receptor recycling (74%)
- receptor recycling (100%)
- cellularization (100%)

protein:
- Ras-related protein Rab-5A (Mus musculus) (100%)
- Ras-related protein Rab-5A (Homo sapiens) (100%)
- Early endosome antigen 1 (Mus musculus) (100%)
- Early endosome antigen 1 (Homo sapiens) (100%)

Figure 4.2: Website Annotation as starting point for further tasks. Going from Identified proteins from full text article to UniProtKB/SwissProt entry to protein sequence.



Home / PLoS ONE, Science/Tech/Health / Delayed Onset of Positive Feedback Activation of Rab5 by Rabex-5 and Rabaptin-5 in Endocytosis

Delayed Onset of Positive Feedback Activation of Rab5 by Rabex-5 and Rabaptin-5 in Endocytosis

By PLoS ONE • on February 16, 2010

Background

Rabex-5 is a guanine nucleotide exchange factor (GEF) that specifically activates Rab5, i.e., converting Rab5-GDP to Rab5-GTP, through two distinct pathways to promote endosome fusion and endocytosis. The direct pathway involves a pool of membrane-associated Rabex-5 that targets to the membrane via an early endosomal targeting (EET) domain. The indirect pathway, on the other hand, involves a cytosolic pool of Rabex-5/Rabaptin-5 complex. The complex is recruited to the membrane via Rabaptin-5 binding to Rab5-GTP, suggesting a positive feedback mechanism. The relationship of these two pathways for Rab5 activation in the cell is unclear.

Methodology/Principal Findings

We dissect the relative contribution of each pathway to Rab5 activation via mathematical modeling and kinetic analysis in the cell. These studies show that the indirect pathway constitutes a positive feedback loop for converting Rab5-GDP to Rab5-GTP on the endosomal membrane and allows sensitive regulation of endosome fusion activity by the levels of Rab5 and Rabex-5 in the cell. The onset of this positive feedback effect, however, contains a threshold, which requires above endogenous levels of Rab5 or Rabex-5 in the cell. We term this novel phenomenon "delayed response". The presence of the direct pathway reduces the delay by increasing the basal level of Rab5-GTP, thus facilitates the function of the Rabex-5/Rabaptin-5-mediated positive feedback loop.

Conclusion

Our data support the mathematical model. With the model's guidance, the data reveal the affinity of Rabex-5/Rabaptin-5/Rab5-GTP interaction in the cell, which is quantitatively related to the Rabex-5 concentration for the onset of the indirect positive feedback pathway. The presence of the direct pathway and increased Rab5 concentration can reduce the Rabex-5 concentration required for the onset of the positive feedback loop. Thus the direct and indirect pathways cooperate in the regulation of early endosome fusion.

- Aubrey Berry Arrested for Rapper Dolla's Murder
- Elites TV Exclusive: "The Truth about Rapper Dolla and Aubrey Berry."
- Chris Allen... err Kris Allen wins American Idol 2009
- If Adam Lambert Fails to Win American Idol, Is America Homophobic?
- Aunt Jodi's Sister Is Talking Smack about Kate Gosselin Again!

Figure 4.3: Website annotation as available in GoWeb

perience. The CORESE system is a browser plug-in, which allows also a graphical exploration of the entities and semantic hyperlinks. Together with GoPubMed Extended and GoWeb, they are all semantic web browsers for different aspects. Each browser complements the rest with different presentation options and abilities.

4.2 Architecture

To support semantic browsing, there are several options to implement this feature. The first option to classify the different approaches is to separate between:

1. implementation directly integrated by the content providers in the original websites and
2. modifications of the content in a post processing step by third parties.

First, semantic extensions provided by the original content or website owner allow for a tight and virtually seamless integration of additional semantic features. This enables short response times, as with a normal website browsing experience. The drawback of the content provider-based approach is that the changes, like an additional database, needs to be done by the provider. The feature of customization for users has to be included in the design and implemented features by the provider. This might not be available or possible. An example for the semantic content from the provider is GoPubMed Extended.

Second, if enrichment is not possible on the provider-side, an additional processing step is introduced to enable post-publishing semantic hyperlinks.

Post-publishing Semantic Hyperlinks

To introduce semantic hyperlinks in published content a post processing is required, that involves several sub tasks:

1. fetch the original site
2. parse, possibly do some format checking
3. perform the semantic annotation
4. modify existing content, or add additional content
5. present to the user

Where these steps are executed depends on the implementation of the semantic browser. In general there are three options:

- server side, using a proxy
- client side, extending existing web browsers
- combination of both.

All three solutions have advantages and disadvantages. A server side implementation can control perfectly what is presented to the user. Every request is passed through the proxy providing the semantic browser. This is also the drawback. Every small request is passed through and requires some processing capacity. Also a proxy might have problems with complex operations (e.g., script-based requests and custom binary protocols) and secure connections, such as HTTPS (Rescorla and Schiffman, 1999) or SSL/TSL (Dierks and Rescorla, 2008).

The client side using a browser offers a tight control on the presentation to the user. The extension of existing browser is possible with an application specific extension or plug-in system. For instance with the plug-in system of the Open Source browser Firefox or the Browser Extensions of the Microsoft Internet Explorer. With these systems it is possible to implement more complex features, for instance interactive maps for exploring the results.

The drawback is that the APIs of the extension are not compatible or even comparable. The Firefox plug-ins use JavaScript, a special language XUL and XML-based configuration. The Internet Explorer can be extended with various Windows-APIs, e.g., active-script or Windows registry modifications and custom DLLs. With the emergence of other browsers such as Opera, Epiphany, Safari or Google-Chrome the number of APIs increased further. There are also attempts to provide a common interface for plug-ins. For instance, there is the Firefox GreaseMonkey plug-in. It provides a *reduced* API for DOM manipulations. The support for GreaseMonkey scripts is also provided in other browsers, such as Google-Chrome, Opera, Safari and even Internet Explorer via an Internet explorer extension.

The combination of both has to be done carefully to avoid the introduction of the drawbacks of both approaches. The standard way of rendering in the browser should be used as much as possible. The influences to add semantic content should be as little as possible. The same goes for the fetching of resources or other requests to the server. Also with the goal in mind to minimize the number of exceptions and specialized implementations for different browser, the most common subset should be selected. In current browsers, this is the plain HTML and JavaScript for dynamic modifications.

All three options have been used to implement semantic browsing. The COHSE system uses a portal/portlet approach with proxy functionality. The Corese system and SemWeb are Firefox plug-ins. This allows to provide more sophisticated features in the GUI. Corese provides a zoomable network view of the relation network for identified concepts.

Light-weight Proxy Implementation

The website annotation in GoWeb uses the combined approach. There is a proxy to modify and annotate the original website, but by using standard HTML tags, it is possible to send the request for additional resources (e.g. images) to the original server. The original content is modified as little as possible to preserve the original layout of the website. The additional information is presented in a sidebar (see Figure 4.3). In detail the website annotation in GoWeb does the following:

1. Extract the target URL from the request, e.g. via URL parameter
2. Check the content type with `HTTP head`, if it isn't a processable format, stop, sent a redirect via http status code `HTTP/1.1 302 Object moved`
3. Fetch the website content, check content type again
4. Parse the HTML, preserve the original characters as much as possible, Required to preserve the layout in preformatted text areas, markup within HTML tags `<PRE></PRE>`
5. Extract text, annotate text, calculate markup ranges adhering to the characters from the original document
6. Create the modified website using the following steps:
   - Find header in HTML, if it not exists create the header using the HTML tag `HEAD`.
   - Search for an existing base URL in the head with HTML tag `<BASE HREF="base url" />`, if it exists input header modifications before this tag.
   - Add a base URL for the server providing the website annotation
   - Add JavaScript for functionalities introduced to the website, e.g. highlighting of keywords and concepts.

  - Add additional base URL containing the extracted base URL from the request URL or if it exists, keep the original base URL.
  - Write website, add `SPAN` tags for markup of annotated content.

7. Send modified website to the user as response to the request.

With these steps, the modified HTML site is rendered by the browser with correct URL for the scripts added into the site. Also due to the preserved – or added – base URL the resources are requested from the originally intended address.

Limitations
Although JavaScript is available on most browsers, there are different levels, on what is available on each system. Consequently only a minimal set of JavaScript can be employed. Also it is problematic to modify web pages with dynamic content. There the proxy to annotate the web page sees only the scaffold for loading the content. The actual content is hidden, as it is requested after the web page is rendered in the browser, e.g., with JavaScript. This is often the case for many modern Web 2.0 web sites.

A further limitation of this approach is the security restriction policy of browsers. There is the restriction of cross-site scripting. This applies, if content is added from a different domain than the original one. The restriction was introduced to minimize harmful attacks, e.g. gaining elevated access to read sensitive information such as session cookies for authentication. To avoid this problem, most of the additional content is added as static content as part of the modification in the proxy. The dynamic parts, such as the induced ontology tree, are presented in a separated frame. This frame has the correct server domain for dynamic content.

The client-server-based semantic web browsers have in common that they annotate the website content on the server. Currently the text mining algorithms are too resource intensive to be executed on the client side.

4.3 Evaluation: Usability Study

The usability of semantic browsing as implemented in GoPubMed-Extended and GoWeb (Chapter 3) is evaluated by Oliver et al. (2009). During user experiments technical and functional feedback has been collected for three semantic browsers. The obtained technical parameters were the submitted query, number and type (e.g., selection in the induced ontology) of requests, the response time for each request, and the time in-between. The functional evaluation followed by enlarge six hypotheses proposed by Oliver et al..

The goal is to assess whether the semantic browsing helps to improve the user experience. This evaluation is part of the Sealife project. The three evaluated semantic browsers are the COHSE and Corese browser applied to the National Electronic Library of Infection (NeLI[5]) portal in the United Kingdom and GoPubMed(–Extended). Together the three semantic browsers are compared to non-semantic systems, e.g., GoPubMed-Extended vs. PubMed. For the evaluation the following hypotheses are examined:

- Mobility and travel within the system

  Hypothesis 1 The semantic web browser reduces the time taken for users to find information or perform tasks.

  Hypothesis 2 The semantic web browser shortens the pathway taken to find information or perform tasks.

[5] `http://www.neli.org.uk`

Hypothesis 3 Where semantic links are available, users will always follow them instead of non-semantic links.

- User attitude and satisfaction

Hypothesis 4 Users find the semantic web browser easier to use than the control platform.

Hypothesis 5 Where semantic links and ranking are available, users prefer them to non-semantic links and ranking.

Hypothesis 6 Use of the semantic web browser is intuitive:

a) Users think the semantic web browser helps them to find information or complete tasks.

b) Users intuitively understand how to use the semantic web browser to find such information or complete tasks.

The setting for the evaluation consists of an online part and organized workshops. Both variants use a pre- and post-questionnaire to gather the user feedback. Workshops allow for a more direct and detailed feedback in form of interviews. But the usage of workshops is limited due the time intensive nature of such events. The advantage of the online approach is to reach as many participants as possible. It introduces also the constraints of unknown users and heterogeneous computer systems. For instance the system requirements for Corese (Firefox-Browser) or COHSE (Internet Explorer 7) is a limiting factor for the number of participants applicable for these systems. To observe the user behavior during the evaluation web server logs are used. The logs contain the clicks of users in form of HTTP requests to the server. To identify individual users during a web session they are tracked by a session id. The id is provided by the evaluation framework with the provided link to the semantic browser. The framework also provides the interface for the questionnaire and tasks. Such tasks are typically answering questions such as “What are the recommended guidelines for hygienic cleaning of surfaces after flooding?” or “What is the main role of the gene MMS2? Name 3 genes related to it in literature”.

The logs collected during such tasks can also be used to validate the feedback provided by the participants. This may include misinterpretations of questions, or the comparison of user impression of time spend on a task versus the measured time intervals.

The overall summary of the evaluation regarding the formulated hypotheses is available in Table 4.1. The main conclusion is that semantic browsers are helpful in finding question answers. The GoPubMed system full-filled the expectations, but for COHSE and Corese this was not always the case. The users favored a polished, mature, and friendly interface, since to the user a system is the visible interface. The algorithms behind an interface are not of interest to the user. Thus a good interface is necessary for a positive user experience. This problem also includes the initial usage barrier. Installing a plug-in into the web browser may already present a barrier to high for a user.

4.4 Summary

Semantic Browsing is a promising direction for supporting the researchers in their tasks. It provides a scheme to address the *Open Problem 2*. A semantic browser helps to explore the search results or websites. It provides, via semantic hyperlinks, additional resources or web services to help answer the users’ questions. The semantic hyperlinks can be included by design in an application, i.e., with the GoPubMed-Extended. Also, semantic hyperlinks can be added in a dynamic way to existing web sites using a light-weight proxy solution (not all requests are routed via the proxy) with JavaScript-based modifications as implemented and shown in GoWeb. This implementation does not need additional modifications of existing web browsers.

	Hypothesis	COHSE	CORESE	GoPubMed
H1	The SWB reduces the time taken for users to find information or perform tasks.	No	Yes	No
H2	The SWB shortens the pathway taken to find information or perform tasks.	No (targets not found)	No (targets found by few users)	—
H3	Where semantic links are available, users will always follow them instead of nonsemantic links.	No	Yes	No
H4	Users find the SWB easier to use than the control platform.	Yes and No	No	Yes
H5	Where semantic links and ranking are available, users prefer them to non-semantic links and ranking.	Yes	Yes	Yes
H6	Use of the SWB is intuitive:			
H6 a)	Users think the SWB helps them to find information or complete tasks.	No	No	Yes
H6 b)	Users intuitively understand how to use the SWB to find such information or complete tasks.	No	No	Yes

Table 4.1: Confirmation or contradiction of original hypotheses for the Evaluation of the Semantic Browsers as stated by Oliver et al. (2009), Table 9.

The evaluation with user study using workshops and online-questionnaire revealed that the interface to the user is a critical point. The interface is part of the entry barrier for users to accept, find and use the features and options. Semantic browser can be seen as tool for connecting information sources, e.g., literature or informal sources as websites, to the existing workflow tools and grid infrastructure.

As future work, the integration, composition of existing web services and workflows has to be evaluated, as existing tools are often stand-alone programs. There is also the challenge to provide post-publishing semantic hyperlinks in dynamic web content, as prevalent in web 2.0 sites. There the usage of the initial HTML as sole input for annotation and highlighting of relevant concepts is insufficient. A further improvement is the re-usage of existing in-line annotations of web sites, such as embedded RDF (eRDF), Microformats[6] or RDFa (Adida and Birbeck, 2008).

[6] http://microformats.org/

CHAPTER 5

SPECIALIZED SEMANTIC SEARCH ENGINES

As previously show with GoWeb, semantic search improves the search and supports the user to answer questions. But the web can be complemented with other diverse information sources relevant for the biomedical domain. To evaluate these sources for semantic search, further semantic search system were implemented. Each prototype handles different aspects or problems for the adaption of a document source for semantic search. They all address the research field of *Open Problem 3*. Examples for the tasks are the advantages and drawbacks of full text annotation, the aspect of intranet crawling, or the handling of new/different background knowledge. A different knowledge base can be more specific or general than the existing one in GoWeb with the GeneOntology and the Medical Subject Headings thesaurus.

The following chapter introduces patents as document source with the semantic search engine GoPatents. The second system GoCell uses full text articles in XML format. MousePubMed provides literature search for the specialized domain of mouse anatomy. The final system GoECDC uses the intranet as data source and combines it with a customized ontology. An intranet is a heterogeneous data source in terms of available documents and provided format.

5.1 GoPatents

The idea of semantic patent search has been addressed under the hood of GoPatents. Patents are an important information source for research and industrial applications. They provide early and in-depth information for many areas of interest. Patents also have a high economical value. Searching for relevant or related patents is part of the research and monetization, but patents may also help or hinder the research in the biomedical field.

5.1.1 Patents

A patent is a right granted to an individual or group, including corporations. It grants the owner "the right to exclude others from making, using, offering for sale, or selling"[1]. Patents themselves cannot stop someone from making or using an invention. The exclusive right has to be enforced by the owner with legal means in court. The World Intellectual Property Organization (WIPO) describes a patent[2] as follows: "A patent is a document which describes an invention which can be manufactured, used, and sold with the authorization of the owner of the patent. An invention is a solution to a specific technical problem. A patent document normally contains at least one claim, the full text of the description of the invention, and bibliographic information such as the applicant's name. The protection given by a patent is limited in time (generally 15 to 20 years from filing or grant). It is also limited territorially to the country or countries concerned.

A patent is an agreement between an inventor and a country. The agreement permits the owner to exclude others from making, using or selling the claimed invention." Due to the national nature of patents, it would be necessary to register a patent in each patent office. For harmonization and cooperation in the procedures for patent application, there have been several treaties. One important treaty is the European Patent Convention (EPC), which was signed in Munich on 5^{th} October 1973 and entered into force on 7^{th} October 1977. This treaty lead to the setup of the European Patent Office (EPO). The EPO currently provides services for patent applications in 41 European countries, including non-EU states such as Iceland, Switzerland or Turkey. A notable missing country is the Russian Federation. The EPO covers about 540 million inhabitants[3].

The international cooperation on patent applications is regulated with the Patent Cooperation Treaty (PCT). Signed on 19th June 1970, entered force 24th January 1978, it currently enables to file patents for more than 130 states. A patent application using the PCT has two phases. The first phase is the international phase. During this the patent protection is pending under a single patent application, which was filed with the local patent office, e.g., EPO. The second phase is the national and regional phase. There the rights are continued by filing necessary documents to other patent offices, e.g., USPTO. After a period of 18 month after the submission (priority date) the patent is published by the International Bureau of the World Intellectual Property Organization (WIPO).

Although, there have been many efforts for harmonization and cooperation, the final decision, which inventions are patentable is governed by local law. The most well known examples for diverse opinions are software patents and patents on (human) genes.

Patent and patent applications search may be used in the following use cases (Atkinson, 2008):

- patentability,
- clearance to market a product,
- patent validity,
- opposition to a patent being sought by another,

[1] http://www.uspto.gov/go/pac/doc/general/

[2] http://www.wipo.int/pctdb/en/glossary.jsp

[3] http://www.epo.org/about-us/press/backgrounders/epo.html

- infringement watch,
- creating intellectual property landscapes for business development or R&D,
- infringement defense,
- litigation,
- prosecution support,
- creation of portfolios for assignments, investments, mergers and acquisitions,
- licenses with legal status and contingency clauses.

Furthermore, patents are an important source to consider during research. There are three main reasons for using patents: (1) avoid reinvention, (2) minimize cost in form of licenses or royalty-fees for intellectual property and (3) early access to research results. For instance, if a new experimental essay was patented, then the researcher can decide if they want to use this technology (1). An informed decision can be made, if additional money and time should be spend for developing an alternative approach or if the royalty-fees for licensing are to be paid (2). The early access (3) is due to the patentability requirements. One requirement is the novelty criteria. To satisfy this, prior publications with the results or methods are not allowed. Thus recent patent applications contain information not yet available through literature.

Patents and patent applications (pending patents) are also available on the web and in specialized databases, for example see Table 5.1. A comparison of Dialog, esp@cenet, Questel-Orbit and STN with regard to intellectual property information is available in (Stock and Stock, 2006).

Name	URL	Provider	Subscription Required
Dialog	`http://www.dialog.com/`	ProQuest (former Thomson)	yes
esp@cenet	`http://ep.espacenet.com/`	European Patent Office	no
FreePatentsOnline	`http://www.freepatentsonline.com/`		no
Google Patents	`https://google.com/patents/`		no
PATENTSCOPE	`http://www.wipo.int/pctdb/`	WIPO	no
patentstorm	`http://www.patentstorm.us/`		no
Questel-Orbit	`http://www.questel.com/`	Questel	yes
STN International	`http://www.stn-international.de/`	FIZ Karlsruhe	yes
USPTO	`http://patft.uspto.gov/`	USPTO	no

Table 5.1: Patent databases with web search interfaces

In terms of information value patents have to be treated with special care. To maximize the protection and market value of a patent, it has to be very broad and contain as many claims as possible. Yet it should reveal only the minimal and by law required amount of information to the competitor. This can for instance be done by using non-obvious synonyms. The general layout of patents with title, abstracts, descriptions, claims (main and sub claims), possible multiple identifiers and translation into different languages do not make this simpler. Due to the high commercial interests related to patents, statements regarding patents have to be treated with care.

The high financial impact of patents, especially in the pharmaceutical/biomedical area, is illustrated with the drug Atorvastatin. As a Pfizer product it has the brand name Lipitor. Lipitor is a drug that is used to lower the blood cholesterol level. It is one of the best selling drugs in the market. In the year 2008 Pfizer's made a sales volume of 12.4 billion US-Dollars with Lipitor. Thus a single drug contributed a quarter of the overall sales volume (41.8 US$) of Pfizer. According to the FDA orange list Lipitor is protected by five patents. The first patent with the US patent number 4681893 was submitted in 1986. It has the title "*Trans-6-[2-(3- or 4-carboxamido-substituted pyrrol-1-yl)alkyl]-4-hydroxypyran-2-one inhibitors of cholesterol synthesis*". The patent has 10 pages and comprises 9 claims. The first 8 describe the compounds and only in the 9th claims the application is stated: "A method of inhibiting cholesterol biosynthesis in a patient in need of such treatment by administering a pharmaceutical composition as defined by claim 8." Due to the maximal period validity period of 15 years, the protection granted by the patent will expire in the year 2011. It is expected that the sales will drop to only a fraction. This was demonstrated by other patent expired drugs, such as antidepressant Prozac. The sales for Prozac dropped from 2 billion US$ to 500 million US$ (Salz, 2009).

5.1.2 Patent Classification Schemes

As Patents cover a wide range of topics, patent offices use standardized schemas to formalize the topic classification of patents. The office assigns in most of the cases one major classification and optionally secondary topics. The most used schemes are:

- IPC - International Patent Classification (IPC) is a hierarchical patent classification. It was started in 1971 and is updated on a regular basis by a Committee of Experts[4].

- USPC - United States Patent Classification is an official patent classification system used and maintained by the United States Patent and Trademark Office (USPTO)[5].

- ECLA - The European Classification (ECLA) is a patent classification system maintained by the European Patent Office (EPO). The ECLA classification system contains 134 000 subdivisions and is an extension of the International Patent Classification system. The ECLA is accessible via the esp@cenet search of the EPO[6].

- F-term - Japanese patent classification based on technical features of the inventions described in them[7].

Patent classification schemes are designed to be used by experts. They usually are taxonomies and they structure the domain into different areas or technologies. They are used during the process of the patent application and later for patent retrieval. For instance, the classification schemes help to restrict searches to the field of interest. Due to the historical growth and expert nature these taxonomies are difficult to navigate. Finding or assigning an appropriate term for a patent is not a simple process. Depending on the classification schema, there are sometime several options to describe a patent. Also if a mapping of assigned classes between schemas may not always be possible. If no corresponding classes exist, the decision for the classification may depend on the expertise of patent officer and/or the granularity of the used classification schema.

For example, there is the patent family WO0241414 (A1) with the title "light emitting component comprising organic layers". The applications of this patent via the WIPO required a classification in IPC, for the USPTO in USPC and for the EPO in ECLA. The most specialized classification is the ECLA term. The assigned "H01L51/50" and subclasses are an extension of the IPC classification, specifically created to describe OLEDs. In contrast the USPC classification of the

[4] http://www.wipo.int/classifications/ipc/en/

[5] http://www.uspto.gov/web/patents/classification/index.htm

[6] http://v3.espacenet.com/eclasrch?classification=ecla

[7] http://www.ipdl.inpit.go.jp/homepg_e.ipdl

patent consists of more general concept classes such as “Fluorescent, phosphorescent, or luminescent layer” (428/690). There are four additional classes (313/504; 313/506; 428/332; 428/917) to classify the OLED patent.

The class 428/690 is an example of the complex hierarchical structure in the USPC. The original two layer structure uses links in the second layer to provide detailed tree like taxonomic structures. For the example the class 428/690 as leaf has he following tree with `subclass-of` relations:

- 428 Stock Material Or Miscellaneous Articles
- 428/411.1 Stock material comprising plural layers* or surfaces, adhered or cohered to each other, identified by the composition of the layers*, and not elsewhere provided for.
- 428/688 Of inorganic material
- 428/689 Metal-compound-containing layer
- 428/690 Fluorescent, phosphorescent, or luminescent layer

Only in conjunction of all names and descriptions of all classes and subclasses the intended meaning is complete. This also include negations, for example in 428/411.1 with the clause “and not elsewhere provided for”.

The complexity of the patent classification schemes is known. To reduce the entry barrier for navigating and understanding the IPC Pesenhofer et al. (2008) proposed a mapping to the scientific structure of Wikipedia.

5.1.3 Patent Mining and Retrieval

Mining patents for information has similarities with literature mining. Both deal with large amounts of textual data and authors. There is a continuous stream of new applications to be processed. There are abstracts and full-text versions.

An important difference is that patents are often deliberately written in a vague, broad or even cloaking manner. One of the reasons for this behavior lies in the goal of writing patents to maximize the scope, but minimize the exposure of know-how. Ontologies in combination with text mining can help to handle this problem. The usage of synonyms and spelling variants is essential to maximize the identification of concepts. Also synonyms and related concepts from the background knowledge offer the possibility to group patents together, even if they use different terminologies for the same topic. This can be used to formalize, store and finally replace the need to include many keywords for the same topic in a patent search query. The language and writing style used in patents is also called Patentese (Atkinson, 2008).

For text mining there are two tasks of interest to be addressed. There is the task of proposing schema classes for patents or patent applications. The other task is information extraction, for instance identifying additional meta-data for more specialized queries.

The task of assigning classes of the patent classification schemes is not solved. Even human experts might not agree (inter-annotator agreement). As discussed in Section 5.1.2, the classification schemes are expert taxonomies. Similar to the GeneOntology the names and descriptions of the classes are not intended to be used as labels for text mining.

As matching labels is not a successful way to propose schema classes, other methods have been implemented. Larkey (1999) uses k-nearest neighbor clustering. Support vector machines have been used for categorization of the International Patent Classification (Fall et al., 2003) and F-term (Li and Bontcheva, 2008).

A method to complement the machine-learning approaches is to use the citation network. It provides a context of topic, categories and classes. Li et al. (2007) use this for an experimental study in nanotechnology. Similar citation networks have been applied for patent search and retrieval (Fujii, 2007).

The second task of finding additional information can use the already discussed techniques in Section 2.5.2. There are also NLP-based approaches for information extraction. Agatonovic et al. (2008) use a GATE-based pipeline to find additional information. They also highlight the necessity of using a scalable approach, including parallelization. The main reason is the size of the test set. Being only a limited subset of patents, it still has a size of 100 GByte.

Semantic search may help to support the patent search experience. Patents can cover nearly any field of application, thus a categorization of patents is mandatory. Patent search with GoPatents has been implemented using two approaches. The first implementation, called GoFreePatentsOnline, uses existing patent search platforms. The second implementation GoPIZ uses a customized patent database.

5.1.4 GoFreePatentsOnline

Patent search is available for free on the web and is provided by multiple search engines, as listed in Table 5.1. They all provide the well established keyword-based search interface with search results as long lists of patents. Their user interfaces are analog to the current web search engines. Similarly, reading result lists of patents or, at least, the patent summaries is a time consuming task.

The GoFreePatentsOnline systems address this problem by introducing semantic filtering to the patent search. It applies the ideas and algorithms developed for GoWeb (see Section 3) to the search results of the free search engine freepatentsonline.com[8]. The system uses the textual snippets from the search results as input for the text mining. The text mining algorithms match the textual labels from the background knowledge. These annotations provide the concepts for the semantic filtering of the search results.

In the following example, the query was "aspirin". The FreePatentsOnline search returns the first 200 of $31,549$ patents. The patent titles and short descriptions are used for the text mining. The concept "Inflammation", as top disease, is selected, which filters the list of patents to 11 relevant patents. The patent, which was originally on position 103, is now ranked to the top. To retrieve the patent full text, the user may click on the displayed patent number.

The re-ranking helps to find a relevant patent. But the text snippet provided by the free search engine is too short and unspecific. Especially during the application and validation phase of a patent it is important to use a patent search with a high recall. Missing patents are considered to be a problem for the use case of patent search. However, for patent retrieval GoFreePatentsOnline can help to reduce the time spend for scanning and reading the search results to identify relevant patents.

5.1.5 GoPIZ

To further explore the semantic search with patents we developed a prototype together with the Patent Information Centre (PIZ) Dresden. The GoPIZ system uses ontological background knowledge to classify full text patents. The usage of full text removes the problem of too short text snippets for the text mining as with the GoFreePatentsOnline system.

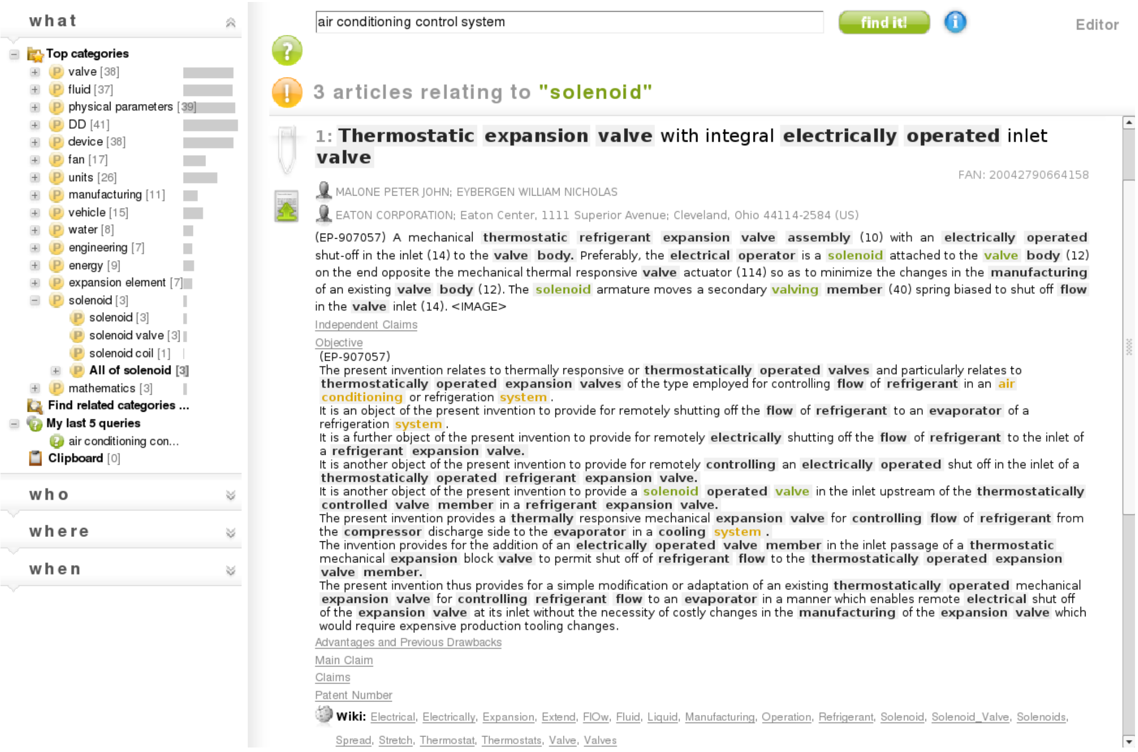
Similar to all current search engines, GoPIZ provides a simple keyword search interface but with the extension of semantic filtering with identified concepts from text. For the prototype GoPIZ, an expert for patents selected a set of more than 7700 patents with relevant text fields such as title, abstract, objective, advantages, independent claims, claims in English, French or German and main claim. These are mostly all available text fields of patents from the European Patent Office (EPO). The selection of the patents is based on an actual case in patent research. It is part of a patent application in the area of valve regulation in mechanical engineering.

The background knowledge was specifically created for this task. The knowledge network was semi-automatically generated and structured using a built-in ontology editor of the system.

[8] http://www.freepatentsonline.com/

With this system a non-expert created the GoPIZ taxonomy within a week. It covers more than 200 concepts, including synonyms, variations, and abbreviations.

The shown example for GoPIZ uses the query “air conditioning control system”. This results in 41 articles. To find all articles relating to “solenoid”, select the concept in induced ontology tree on the left side. This results in 3 articles. The full text is hidden by default and can easily be viewed by just clicking on the required part. To view the objective of the patent, the user may click on the “Objective” heading. As a result of this action, the text of the claim is annotated, the identified concepts and keywords are highlighted, and in the interface this claim is added (Figure 5.1). Additionally, GoPIZ offers links to Wikipedia entries, for instance, to learn more about solenoids. This additional information source includes the fact, that electro-mechanical solenoids can be used as special type of relay in pneumatic or hydraulic valves.



3 articles relating to "solenoid"

1: Thermostatic expansion valve with integral electrically operated inlet valve

FAN: 20042790664158

MALONE PETER JOHN; EYBERGEN WILLIAM NICHOLAS

EATON CORPORATION; Eaton Center, 1111 Superior Avenue; Cleveland, Ohio 44114-2584 (US)

(EP-907057) A mechanical thermostatic refrigerant expansion valve assembly (10) with an electrically operated shut-off in the inlet (14) to the valve body. Preferably, the electrical operator is a solenoid attached to the valve body (12) on the end opposite the mechanical thermal responsive valve actuator (114) so as to minimize the changes in the manufacturing of an existing valve body (12). The solenoid armature moves a secondary valving member (40) spring biased to shut off flow in the valve inlet (14). <IMAGE>

Independent Claims

Objective

(EP-907057)
The present invention relates to thermally responsive or thermostatically operated valves and particularly relates to thermostatically operated expansion valves of the type employed for controlling flow of refrigerant in an air conditioning or refrigeration system .
It is an object of the present invention to provide for remotely shutting off the flow of refrigerant to an evaporator of a refrigeration system .
It is a further object of the present invention to provide for remotely electrically shutting off the flow of refrigerant to the inlet of a refrigerant expansion valve.
It is another object of the present invention to provide for remotely controlling an electrically operated shut off in the inlet of a thermostatically operated refrigerant expansion valve.
It is another object of the present invention to provide a solenoid operated valve in the inlet upstream of the thermostatically controlled valve member in a refrigerant expansion valve.
The present invention provides a thermally responsive mechanical expansion valve for controlling flow of refrigerant from the compressor discharge side to the evaporator in a cooling system .
The invention provides for the addition of an electrically operated valve member in the inlet passage of a thermostatic mechanical expansion block valve to permit shut off of refrigerant flow to the thermostatically operated expansion valve member.
The present invention thus provides for a simple modification or adaptation of an existing thermostatically operated mechanical expansion valve for controlling refrigerant flow to an evaporator in a manner which enables remote electrical shut off of the expansion valve at its inlet without the necessity of costly changes in the manufacturing of the expansion valve which would require expensive production tooling changes.

Advantages and Previous Drawbacks

Main Claim

Claims

Patent Number

Wiki: Electrical, Electrically, Expansion, Extend, FlOw, Fluid, Liquid, Manufacturing, Operation, Refrigerant, Solenoid, Solenoid_Valve, Solenoids, Spread, Stretch, Thermostat, Thermostats, Valve, Valves

Figure 5.1: GoPIZ with query “air conditioning control system” and selected concept “solenoid”, including the highlighted and annotated patent objective. Wikipedia links are available below the patent.

The biggest technical and algorithmic challenges of GoPIZ is the handling of full text patents. Patents as used in GoPIZ are structured entries, but the fields containing the claims and descriptions are long free text passages. These text fields alone may possibly be longer than a normal scientific full text article. This needs to be reflected in system architecture. For this the rendering of patents search results, especially the more detailed claims, are done asynchronously and on-demand. To achieve this AJAX with JSON messages is employed. In the initial search result presented to the user, the claims are not rendered. Instead a place holder with the option to open the claims is shown to the user, thus reducing the rendering time.

For the induction of the tree for the filtering feature, all the patents are pre-annotated with the background knowledge and cached. Therefore no additional logic is required. The reduced presentation of patents enables also a more compact overview of the search results and scanning for relevant patents by the user.

As a result of this pilot study and resulting prototype, feedback was collected from the PIZ. The main comment, was that the interface was more comfortable, than current systems such as the patent search of the European Patent Office (EPO)[9]. Furthermore, the filtering system of GoPIZ that includes the sub-concepts of a concept helped to explore the patents.

There are two main requests to improve the prototype form the PIZ. First, remove the step of gathering potentially relevant patents. This means that the system should have access to all patents. Second, integrate the existing patent classification systems. Unfortunately, in this pilot study both requests are out of the scope for the prototype. Other requirements for a patent search system have been proposed by Tseng and Wu (2008). Their list consists of the following items:

Self-defined search result item display function To asses the relevance the most used parts of a patent are the summary snippets, pictures, titles, and claims. However, most government patent search websites do not display these data on the search result page.
GoPIZ includes these parts already in the search results.

Suggested vocabulary feedback function Provide a mechanism to suggest and extend search vocabulary, if the initial search results are unsatisfactory.
GoPIZ provides the ontology to identify related concepts.

Word and command error correction function Correct errors in keywords and search commands, especially for more complex Boolean queries. *GoPIZ allows to construct complex queries with the help of the ontology.*

Correspondence system for frequently used vocabulary between languages Provide built-in support for cross-language queries and search for common concept in multiple languages, such as translations between traditional Chinese, simplified Chinese, and English. *Can be added in GoPIZ by adding non-English synonyms to the ontology.*

Look-up system between and mapping between patent classification schemes Provide a helper to map and classification numbers of different patent classification schemes, i.e. country specific to improve multi-lingual patent search. *Ontology and schema mapping is open research topic.*

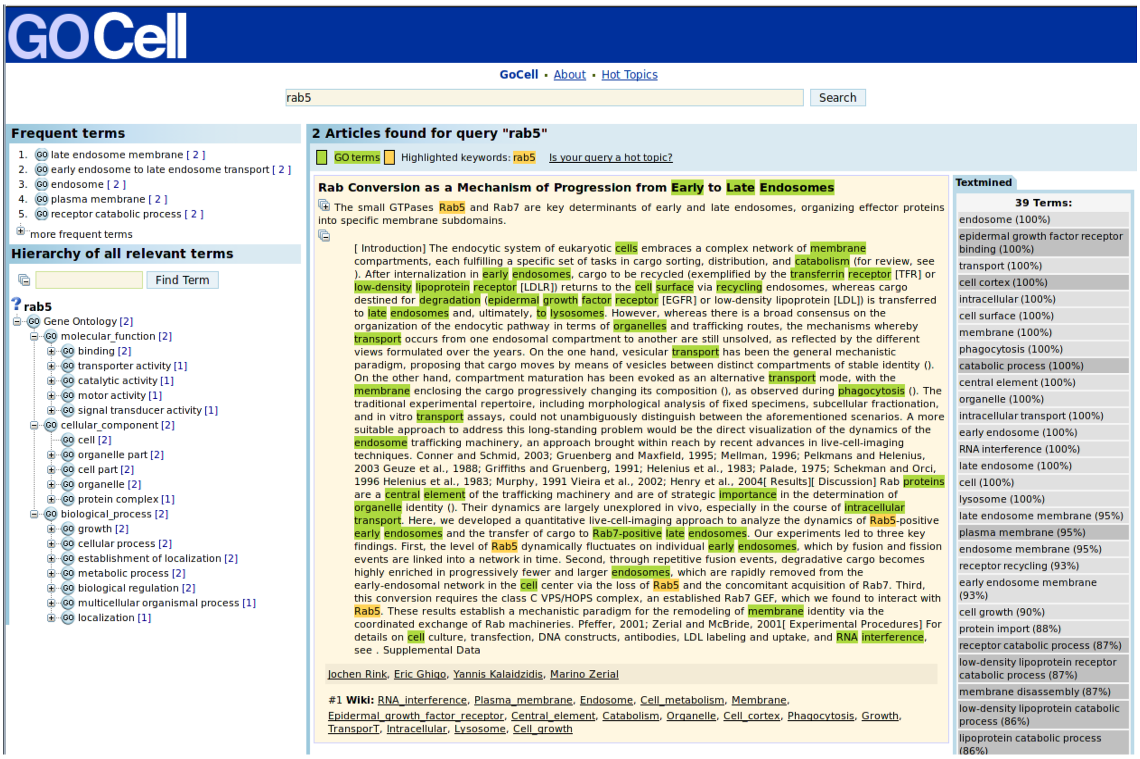
These items show that the current patent search system needs a customization feature and many convenience-based hints and links. GoPIZ already incorporates the first three of these feature requests.

Semantic patent search is a promising application for semantic technologies. They help to filter the large quantities of data that are connected to patent search. But one current limit is also the data size, as any serious effort is straightaway confronted with scaling issues.

[9] `http://ep.espacenet.com/`

5.2 GoCell

The GoCell prototype system was developed to explore the applicability of ontology-based semantic search for full text from the scientific literature and publications. The GeneOntology is used to annotate ten years of full text articles of the Cell journal[10]. The XML data files were provided as part of a cooperation with the publisher Elsevier[11]. The goal was to improve the system to handle full text and provide a demonstrator of semantic search for the publisher. For a screen shot of the system see Figure 5.2.



Frequent terms

1. GO late endosome membrane [2]
2. GO early endosome to late endosome transport [2]
3. GO endosome [2]
4. GO plasma membrane [2]
5. GO receptor catabolic process [2]

⊞ more frequent terms

Hierarchy of all relevant terms

rab5

- GO Gene Ontology [2]
  - GO molecular_function [2]
    - GO binding [2]
    - GO transporter activity [1]
    - GO catalytic activity [1]
    - GO motor activity [1]
    - GO signal transducer activity [1]
  - GO cellular_component [2]
    - GO cell [2]
    - GO organelle part [2]
    - GO cell part [2]
    - GO organelle [2]
    - GO protein complex [1]
  - GO biological_process [2]
    - GO growth [2]
    - GO cellular process [2]
    - GO establishment of localization [2]
    - GO metabolic process [2]
    - GO biological regulation [2]
    - GO multicellular organismal process [1]
    - GO localization [1]

2 Articles found for query "rab5"

🟩 GO terms 🟨 Highlighted keywords: rab5 Is your query a hot topic?

Rab Conversion as a Mechanism of Progression from Early to Late Endosomes

The small GTPases Rab5 and Rab7 are key determinants of early and late endosomes, organizing effector proteins into specific membrane subdomains.

[Introduction] The endocytic system of eukaryotic cells embraces a complex network of membrane compartments, each fulfilling a specific set of tasks in cargo sorting, distribution, and catabolism (for review, see). After internalization in early endosomes, cargo to be recycled (exemplified by the transferrin receptor [TFR] or low-density lipoprotein receptor [LDLR]) returns to the cell surface via recycling endosomes, whereas cargo destined for degradation (epidermal growth factor receptor [EGFR] or low-density lipoprotein [LDL]) is transferred to late endosomes and, ultimately, to lysosomes. However, whereas there is a broad consensus on the organization of the endocytic pathway in terms of organelles and trafficking routes, the mechanisms whereby transport occurs from one endosomal compartment to another are still unsolved, as reflected by the different views formulated over the years. On the one hand, vesicular transport has been the general mechanistic paradigm, proposing that cargo moves by means of vesicles between distinct compartments of stable identity (). On the other hand, compartment maturation has been evoked as an alternative transport mode, with the membrane enclosing the cargo progressively changing its composition (), as observed during phagocytosis (). The traditional experimental repertoire, including morphological analysis of fixed specimens, subcellular fractionation, and in vitro transport assays, could not unambiguously distinguish between the aforementioned scenarios. A more suitable approach to address this long-standing problem would be the direct visualization of the dynamics of the endosome trafficking machinery, an approach brought within reach by recent advances in live-cell-imaging techniques. Conner and Schmid, 2003; Gruenberg and Maxfield, 1995; Mellman, 1996; Pelkmans and Helenius, 2003 Geuze et al., 1988; Griffiths and Gruenberg, 1991; Helenius et al., 1983; Palade, 1975; Schekman and Orci, 1996 Helenius et al., 1983; Murphy, 1991 Vieira et al., 2002; Henry et al., 2004[Results][Discussion] Rab proteins are a central element of the trafficking machinery and are of strategic importance in the determination of organelle identity (). Their dynamics are largely unexplored in vivo, especially in the course of intracellular transport. Here, we developed a quantitative live-cell-imaging approach to analyze the dynamics of Rab5-positive early endosomes and the transfer of cargo to Rab7-positive late endosomes. Our experiments led to three key findings. First, the level of Rab5 dynamically fluctuates on individual early endosomes, which by fusion and fission events are linked into a network in time. Second, through repetitive fusion events, degradative cargo becomes highly enriched in progressively fewer and larger endosomes, which are rapidly removed from the early-endosomal network in the cell center via the loss of Rab5 and the concomitant acquisition of Rab7. Third, this conversion requires the class C VPS/HOPS complex, an established Rab7 GEF, which we found to interact with Rab5. These results establish a mechanistic paradigm for the remodeling of membrane identity via the coordinated exchange of Rab machineries. Pfeffer, 2001; Zerial and McBride, 2001[Experimental Procedures] For details on cell culture, transfection, DNA constructs, antibodies, LDL labeling and uptake, and RNA interference, see . Supplemental Data

Jochen Rink, Eric Ghigo, Yannis Kalaidzidis, Marino Zerial

#1 **Wiki:** RNA_interference, Plasma_membrane, Endosome, Cell_metabolism, Membrane, Epidermal_growth_factor_receptor, Central_element, Catabolism, Organelle, Cell_cortex, Phagocytosis, Growth, TransporT, Intracellular, Lysosome, Cell_growth

Textmined

39 Terms:

endosome (100%)
epidermal growth factor receptor binding (100%)
transport (100%)
cell cortex (100%)
intracellular (100%)
cell surface (100%)
membrane (100%)
phagocytosis (100%)
catabolic process (100%)
central element (100%)
organelle (100%)
intracellular transport (100%)
early endosome (100%)
RNA interference (100%)
late endosome (100%)
cell (100%)
lysosome (100%)
late endosome membrane (95%)
plasma membrane (95%)
endosome membrane (95%)
receptor recycling (93%)
early endosome membrane (93%)
cell growth (90%)
protein import (88%)
receptor catabolic process (87%)
low-density lipoprotein receptor catabolic process (87%)
membrane disassembly (87%)
low-density lipoprotein catabolic process (86%)
lipoprotein catabolic process (86%)

Figure 5.2: Screen shot of GoCell prototype system.

Technical aspects introduced in the project are the capability of handling large and complex XML documents. This includes a more complex XML schema and DTDs for validation. For example the elsevier DTD 5 had 449 pages of documentation (Bernickus et al., 2005). Additionally the information is distributed over multiple files for single articles. Thus information aggregation in multiple parsing steps is required for the information preprocessing.

For the GoCell system the presentation of full text articles in combination with annotations of full text articles is an open problem. There are several options how to design and implement the annotation process and presentation to the user. The direct solution is to annotate the complete full text via text mining and to present the article in full, including the identified and highlighted concepts, as search results.

This option may cause several problems. As for the text mining, some sections of an article are less relevant and produce lots of irrelevant hits. This may include article sections such as background, methods, and literature references. The information gain from using the full text sections versus a well structured abstract is an open problem (Lin, 2009). Similarly, for the presentation to the user rendering the complete article may be perceived as too long.

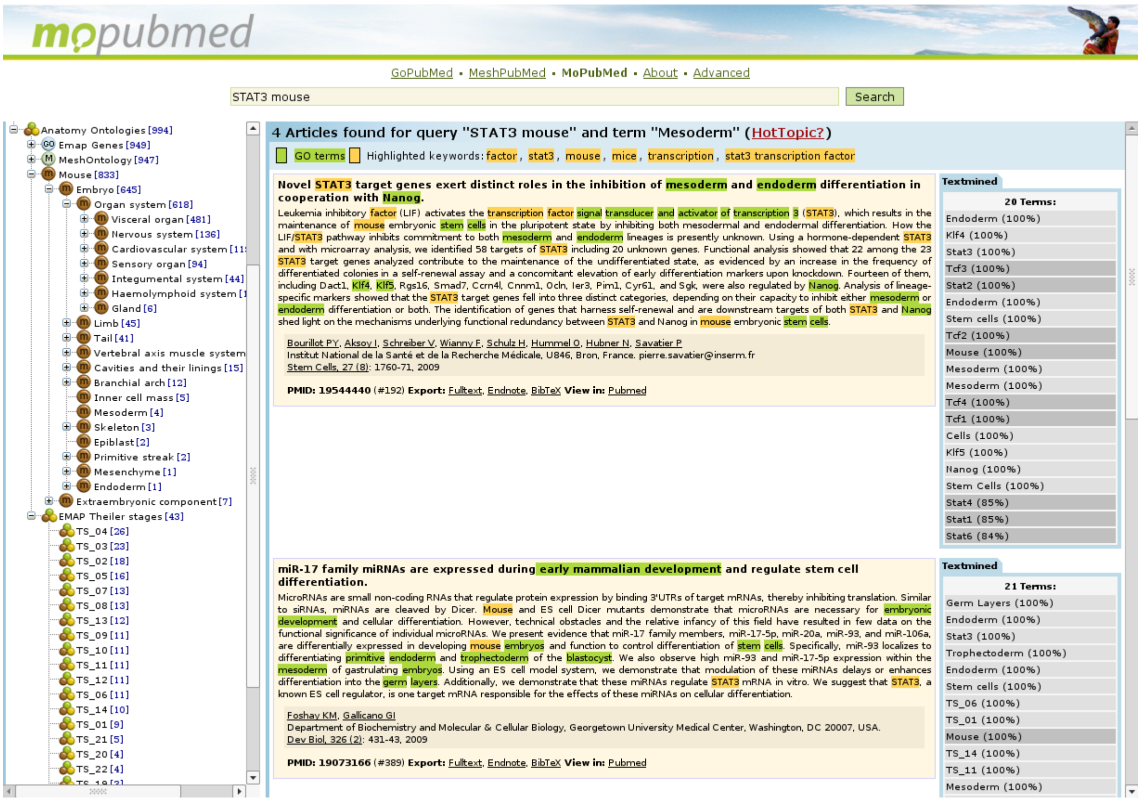
In cooperation with Elsevier the GoCell system has been developed further into a second prototype. This system is called GoElsevier. It contains additional full texts of various Elsevier

[10] http://www.cell.com/
[11] http://www.elsevier.com

journals and the GeneOntology and Medical Subject Headings as background knowledge. Later, this updated system platform was used to create and test an ontology for carbon-dioxide sequestrations (Gutknecht, 2010). For this study, the system had to process and annotate textual data using a non-biological ontology. This indicates the capability of such a semantic search system with regards to portability.

5.3 MousePubMed

To use ontology-based literature search for developmental biology, we built MousePubMed using vocabularies for mouse anatomy (EMAP, Baldock et al. 2003), human anatomy (EHDA, Hunter et al. 2003), mouse genes (from EMAP), and mouse developmental stages (Theiler, 1989) as resources. See Figure 5.3 for an example screen shot of the application. To demonstrate MousePubMed's utility, we evaluate it against tissue and developmental stage annotations in the Edinburgh Mouse Atlas. Before we discuss this evaluation, we introduce the matching algorithm developed.



4 Articles found for query "STAT3 mouse" and term "Mesoderm" (HotTopic?)

🟩 GO terms 🟨 Highlighted keywords: factor, stat3, mouse, mice, transcription, stat3 transcription factor

Novel STAT3 target genes exert distinct roles in the inhibition of mesoderm and endoderm differentiation in cooperation with Nanog.

Leukemia inhibitory factor (LIF) activates the transcription factor signal transducer and activator of transcription 3 (STAT3), which results in the maintenance of mouse embryonic stem cells in the pluripotent state by inhibiting both mesodermal and endodermal differentiation. How the LIF/STAT3 pathway inhibits commitment to both mesoderm and endoderm lineages is presently unknown. Using a hormone-dependent STAT3 and with microarray analysis, we identified 58 targets of STAT3 including 20 unknown genes. Functional analysis showed that 22 among the 23 STAT3 target genes analyzed contribute to the maintenance of the undifferentiated state, as evidenced by an increase in the frequency of differentiated colonies in a self-renewal assay and a concomitant elevation of early differentiation markers upon knockdown. Fourteen of them, including Dact1, Klf4, Klf5, Rgs16, Smad7, Ccrn4l, Cnnm1, Ocln, Ier3, Pim1, Cyr61, and Sgk, were also regulated by Nanog. Analysis of lineage-specific markers showed that the STAT3 target genes fell into three distinct categories, depending on their capacity to inhibit either mesoderm or endoderm differentiation or both. The identification of genes that harness self-renewal and are downstream targets of both STAT3 and Nanog shed light on the mechanisms underlying functional redundancy between STAT3 and Nanog in mouse embryonic stem cells.

Bourillot PY, Aksoy I, Schreiber V, Wianny F, Schulz H, Hummel O, Hubner N, Savatier P
Institut National de la Santé et de la Recherche Médicale, U846, Bron, France. pierre.savatier@inserm.fr
Stem Cells, 27 (8): 1760-71, 2009

PMID: 19544440 (#192) **Export:** Fulltext, Endnote, BibTeX **View in:** Pubmed

Textmined

20 Terms:

Endoderm (100%)
Klf4 (100%)
Stat3 (100%)
Tcf3 (100%)
Stat2 (100%)
Endoderm (100%)
Stem cells (100%)
Tcf2 (100%)
Mouse (100%)
Mesoderm (100%)
Mesoderm (100%)
Tcf4 (100%)
Tcf1 (100%)
Cells (100%)
Klf5 (100%)
Nanog (100%)
Stem Cells (100%)
Stat4 (85%)
Stat1 (85%)
Stat6 (84%)

miR-17 family miRNAs are expressed during early mammalian development and regulate stem cell differentiation.

MicroRNAs are small non-coding RNAs that regulate protein expression by binding 3'UTRs of target mRNAs, thereby inhibiting translation. Similar to siRNAs, miRNAs are cleaved by Dicer. Mouse and ES cell Dicer mutants demonstrate that microRNAs are necessary for embryonic development and cellular differentiation. However, technical obstacles and the relative infancy of this field have resulted in few data on the functional significance of individual microRNAs. We present evidence that miR-17 family members, miR-17-5p, miR-20a, miR-93, and miR-106a, are differentially expressed in developing mouse embryos and function to control differentiation of stem cells. Specifically, miR-93 localizes to differentiating primitive endoderm and trophectoderm of the blastocyst. We also observe high miR-93 and miR-17-5p expression within the mesoderm of gastrulating embryos. Using an ES cell model system, we demonstrate that modulation of these miRNAs delays or enhances differentiation into the germ layers. Additionally, we demonstrate that these miRNAs regulate STAT3 mRNA in vitro. We suggest that STAT3, a known ES cell regulator, is one target mRNA responsible for the effects of these miRNAs on cellular differentiation.

Foshay KM, Gallicano GI
Department of Biochemistry and Molecular & Cellular Biology, Georgetown University Medical Center, Washington, DC 20007, USA.
Dev Biol, 326 (2): 431-43, 2009

PMID: 19073166 (#389) **Export:** Fulltext, Endnote, BibTeX **View in:** Pubmed

Textmined

21 Terms:

Germ Layers (100%)
Endoderm (100%)
Stat3 (100%)
Trophectoderm (100%)
Endoderm (100%)
Stem cells (100%)
TS_06 (100%)
TS_01 (100%)
Mouse (100%)
TS_14 (100%)
TS_11 (100%)
Mesoderm (100%)

Figure 5.3: MousePubMed example screen shot

5.3.1 Extracting Gene Names, Anatomy Concepts and Developmental Stages

Ontology-based text mining is not restricted to finding words or word groups in text. The structure of the ontology can be used to state the relation between a concept and a document by finding the children of the concept. This task is reasonably well solvable for the Gene Ontology, where its concept labels are self descriptive. Many concepts in GO are contained in their children concepts (Ogren et al., 2004). As an example, the concept "envelope" is refined into "organelle envelope" and further to "organelle envelope lumen". The ontology for the Abstract Mouse contains anatomical concepts in the mouse embryo at different embryonic developmental stages. The vocabulary is used to annotate images of mouse embryos. It unifies the vocabulary needed to describe the different parts throughout 26 Theiler stages. Concepts like organs or body parts are further refined into tissue types, unspecific loci such as “cavities”, “left”, “upper”, as well as general concepts such as “node” or “skin”. Considering only textual labels, one cannot distinguish between the different ontological concepts. For example, “chorion” has the children “mesoderm”, “ectoderm” and “mesenchyme”. “Amnion” and “yolk sac” have children sharing the same labels. Searching for documents related to “chorion” will retrieve very similar document sets to searching for “amnion”, only because the documents mention “mesoderm”, in this case with meaning “mesoderm specific to amnion”. Different anatomical concepts share the same concept label. For instance, there exist 171 individuals with label “epithelium”. These all refer to different body parts at a specific stage in development.

Ontology-based text mining relies on the assumption that unique or similar types of directed non-cyclic relationships exist, which can be unified in the hierarchical relationships creating a taxonomy. This assumption does not hold for the Abstract Mouse ontology. There does not always exist a path to the common root supported by only one type of hierarchical relationships. Therefore a document is annotated with a concept from the Abstract Mouse ontology, using only the concept label and its synonymous labels. In the Abstract Mouse Ontology the concept labels follow various creation patterns. Sometimes a child concept contains information of the parent concept (for example, “cavities” has the child “amniotic cavity”). In other cases a concept like “umbilical vein” has the children “left” and “right”, rather than “left umbilical vein” and “right umbilical vein”, respectively. These short and common sense labels make the text annotations arbitrary.

For our experiments we slightly adapted the ontology. For the concepts “left”, “right”, “upper”, “lower”, “common”, “anterior”, and “posterior” we expanded the concept labels with its parent’s labels. For example, with the parent concept “Eyelids” we replaced “upper” with “upper eyelids” and “lower” with “lower eyelids”. To distinguish between common concepts such as “skin” — for instance, occurring for different organs — the matching algorithm took text annotations for ancestor concepts into account. Terms with the same label were grouped according to the number of text annotations for their ancestors in the same document. Only annotations of the top ranked group were confirmed.

Figure 5.4 shows an example for the concept “skin”. There were multiple possibilities to resolve this concept to a specific tissue. Only when a parental concept (shoulder, upper arm, etc.) was found, the text was annotated with the specific skin.

Finding gene names in documents is done using exact matching against gene names contained in EMAP. We enriched this set using additional names and synonyms for each gene taken from the MGI database[12]. We tested all 1437 genes mentioned in EMAP for their annotations with tissues and Theiler stages in PubMed.

We analyzed $123,074$ abstracts retrieved from PubMed with the query “mouse AND development”. This amounted for approximately 0.7% of all documents listed in PubMed. Based on the document annotations with ontology concepts we issued in total $36,358$ statements on relations between genes, tissue and developmental stages, which we extracted from EMAP. Cases with multiple Theiler stages from EMAP were split into separate statements. We evaluated the tissues mentioned using EMAP’s Abstract Mouse ontology and the anatomy part of MeSH. For path descriptions like “embryo.ectoderm” in EMAP we required the matching document to be annotated with the concepts “embryo” and “ectoderm”. For MeSH we also included descending concepts. A document was annotated with the concept “embryo” if annotations for its descendants, for example, “germ layers” or its children “ectoderm”, “endoderm” or “mesoderm” were found.

To find mentions of Theiler stages in text, it was not enough to search for them directly, as they seldom occur as such in abstracts (“Theiler stage 12”, “TS12”, etc.). We therefore compiled

[12] See `http://www.informatics.jax.org`

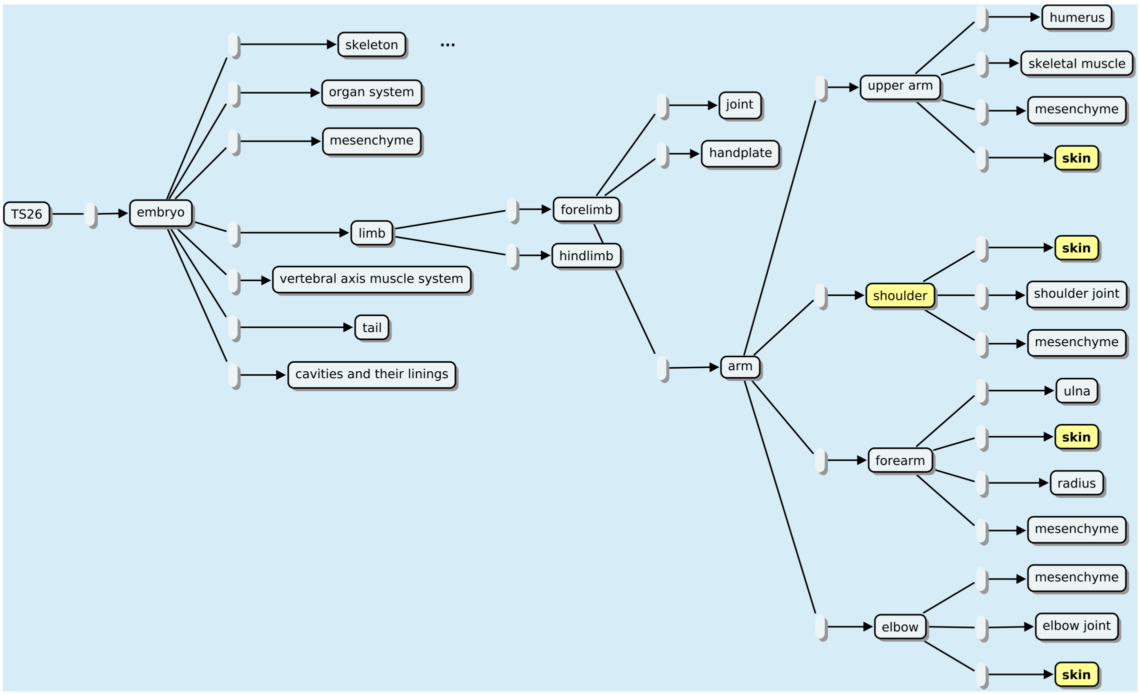


Figure 5.4: Excerpt from the anatomy ontology, for different types of skin. Occurrences of the concept “skin” (yellow concept nodes) in a text were resolved using the hierarchical dependencies. Only when a parental node was also found, for instance, “shoulder”, we annotated the text with “skin”.

a set of regular expressions based on two main notions, the mentioning of embryonic days (E) and of days post coitum (dpc). These expressions had to capture occurrences like

- “embryonic day 10.5”,
- “day 9 mouse embryos”,
- “between E3.5 (E = embryonic day) and E8.5”,
- “12.5 days post coitum”, and also
- “7.5-13.5 days post-conception”.

As mentionings of Theiler stages do not often occur, but rather general time spans are given (“early embryonic development”), we decided to assign Theiler stages 1 – 14 to “early development”, and stages 20 – 27 to “late development”, respectively. Thus every mention of an “early developmental stage” was treated as a match for stages 1 – 14. Both assignment were based on statements found in PubMed relating days to general time spans.

5.3.2 Experiment Designs

To assess the potential of ontology-based literature searches, we designed two experimental scenarios. For the first, we manually collected two sets of queries and detailed answers. For the second scenario, we evaluated the complete EMAP data. Using the methodology described in the previous section, we tried to find textual evidences for all sets in PubMed. This means that we searched PubMed for abstracts that shared annotations for each collected triple consisting of a gene, tissue, and Theiler stage.

5.3.2.1 Manually Curated Test Set

We first manually selected a set of questions to study results in detail. The idea was to send simple keyword queries to MousePubMed, asking for mouse abstracts that discuss a certain tissue and embryonic day. MousePubMed should then identify all genes mentioned in the top-ranking abstracts. Questions and retrieved answers were as follows:

- Which genes play a role in the development of the nervous system in Theiler stage 14? A query for "mouse development nervous system 9 dpc" finds the genes Adamts9, Hoxb4, Otx3, and EphA4 within the first eight abstracts[13]. In addition, the genes EphA2, A3, A7, B1, B2, and B4 are found, which are not yet annotated in the EMAP database.
- Which genes play a role in sex differentiation during murine embryo development? A corresponding query for "mouse sex 10 dpc" results in a set of eight genes within the first fifteen abstracts: Fgf9, Asx11, Sry, Sox9, Usp9x, Maestro/Mro, Wt1, Amh1 and Fra1[14]. Only half of the genes can be found in EMAP so far.
- Which genes play a role in the development of the murine embryonic liver? A query for "mouse 'liver development' " results in a set of several genes, most of which can be found in EMAP as well: Shc, Pxn, Grb2, PEST/Pcnp, GATA6, HNF4a, Foxa1/2, Zhx2, HNF6, Mtf1, SEK1, Nfkb1, c-Jun, Itih-4, and Hex. However, to answer this question exactly, too few abstracts mention particular Theiler stages or days post congestion. They rather refer to "early stages of development", and the exact time span might be presented in the full text article only.

All the results, in particular where genes and exact Theiler stages are concerned, are highly dependent on the ordering of abstracts as provided by PubMed. Whenever a new publication appears containing the same search keywords, it will displace abstracts potentially more informative regarding the original question. Abstracts answering the original question might not appear among the top search results. However, text mining methods will still extract all the data, even from older publications, thus allow filtering to the right set of articles. The abstracts resulting from a keyword search occur in the same ordering as provided by PubMed. That is, in general, the most recent articles occur first. However, querying for species, tissues, and stages still returns the abstracts that discuss the relevant genes. Although corresponding expression patterns might first have been described in older publications, even in recent publications the relevant genes re-appear quite often.

5.3.2.2 Reconstructing Outcomes of Large-Scale Screening

Thut et al. (2001) provided a list of 62 genes found expressed during eye development in mice, together with developmental stage and substructure. Of the 62 genes, 26 were not previously reported (as of 2001); to 16 genes, novel valuable information could be added; 20 genes were fully reported before. Expression patterns were summarized for E12.5, E13.5, E14.5, E16.5, E18.5 and P2. Using MousePubMed, we tried to reconstruct the result of this large-scale screen of 1000 genes.

As Table 5.2 shows, nine PubMed abstracts containing the full information as stated by Thut et al. (2001), mentioning gene, tissue, and specific stages (days). However in most cases, not all data were contained in one single abstract. In three cases, we were not able to automatically spot the gene name (left column), in all cases this was due to synonyms lacking in EMAP and MGI. Note that the assessment of recognizing genes was based only on genes mentioned in EMAP. The tissue could be found in almost all of the cases; from most abstracts, even the specific part of the eye could be extracted.

[13] Important for answering this query are returned PubMedIDs 12736215, 12055180, 11403717.

[14] Important are PubMedIDs 16540514, 16412590, 14978045, 14684990, 14516667, 12889070, 9879712, 9115712.

Gene	Tissue	Stage	PubMedID
Sparc	retina, RPE, eye	E4.5, E5, E10, E14, E17	9367648
Sparc	lens	embryonic day (E)14	16303962
Stat3	retina, RPE, eye	-no specific stage-	12634107
Stat3	lens	E10.5	14978477
Pedf	RPE	-no specific stage-	7623128
Pedf	retina	E14.5, 18.5	12447163
Runx1	inner retina	embryonic day 13.5	16026391
Col15a1	conjunctiva, cornea	E10.5-18.5	14752666
Otx2	outer retina	-no specific stage-	15978261
Edn1	retina	-no stage-	11413193
IGF-II	eye, cornea, retina, scleral cells	E14	2560708
Wnt7b	anterior eye, cornea, optic cup, iris	-no specific stage-	16258938
CDH2	—	-no stage-	9210582
not found, “N2A/3T3 cells” not solved			
—	lens	-no stage-	9211469
Col9a1	eye, lens vesicle, neural retina, ciliary epithelial cells, cornea	13.5, 16.5-18.5 d.p.c.	8305707
Tgfb2	cornea, lens, stroma	-no specific stage-	11784073
Thra	retina	-no specific stage-	9412494
BMP4	retina	E5	17050724
Bmp4	optic vesicle, lens	-no specific stage-	15558471
BMP4	lens, optic vesicle	-no specific stage-	9851982
—	eyes	N/A	15902435
Sox1/2	lens	-no stage-	15902435
—	retina, eye axis	E2, E3, E5	15113840
Notch1	eye	-no specific stage-	11731257
Notch2	eye	-no specific stage-	11171333

Table 5.2: Expression patterns identified by MousePubMed in articles derived from Thut et al. (2001). Often, an abstract does not mention a (specific) developmental stage; The — means MousePubMed did not find this particular fact, otherwise the facts are listed as identified by MousePubMed. Given are only tissues related to the murine eye.

5.3.2.3 Complete EMAP Test Set

To evaluate capabilities of automated searches against the complete EMAP data, the experimental setting was as follows. Genes in EMAP have annotated tissues, in which they were detected at various stages of embryo development. Thus, we queried MousePubMed with each gene and checked which tissues were mentioned in the resulting PubMed abstracts. This was based on co-occurrence of the gene considering, a tissue, and a Theiler stage (day) in the same abstract. Currently, there are 1437 genes in the EMAP database annotated with (sometimes multiple) tissues and stages. All in all, we identified 18,179 such triples — gene, tissue, and stage — in EMAP. Many of the annotations consist of general annotations for tissue, like “mouse”, “embryo”, “left”, “female”, “node”. We removed such trivial instances, because they would very frequently found. 12,782 triples referred to specific tissues, and we tried to find these triples using the aforementioned concept extraction (also see Table 5.3).

As Table 5.4 shows, we were able to reconstruct 31% of the gene-tissue associations in EMAP using PubMed abstracts. Only 13% of the full information (gene, tissue, exact stage) was contained in abstracts. All in all, the data recovered from PubMed included information on about

Type of information	Amount of data
Genes with tissues, stages	1437
Genes with at least one non-trivial tissue, stages	1346
Triples of gene, tissue, stage	18, 179
Triples of gene, non-trivial tissue, stage	12, 782
Tuples of gene, non-trivial tissue	8653

Table 5.3: Types of information and quantity contained in EMAP.

Type of information	Amount of data	
Triples of gene, non-trivial tissue, stage	1637	(12.8%)
Tuples of gene, non-trivial tissue	2667	(30.8%)
Genes with at least one tissue and stage	537	(37.4%)

Table 5.4: Number of tuples/triples consisting of gene and tissue or gene, tissue and stage found in PubMed abstracts retrieved by the query "mouse AND development".

37% of the EMAP genes. We noticed that in many cases abstracts do not mention specific time points during development. Sometimes, "early" and "late development" are mentioned, which we resolved as described previously in this section. On the other hand, mentions like "in early liver development" could not be resolved to specific overall-stages without background information. Cross-checks revealed that indeed much of the necessary information was only mentioned in the full text of references annotated by EMAP for a specific association.

5.3.3 Conclusion

We discussed the specific extraction algorithms needed for MousePubMed and evaluated them small scale on examples relating to eye development and large scale on gene-tissue-stage triple from the Edinburgh Mouse Atlas. We were able to reconstruct 37% of genes, 31% of gene-tissue associations and 13% of gene-tissue-stage associations from PubMed abstracts. These figures are encouraging as only abstracts are used.

The special features of the anatomy ontology, meaning very short names and temporal descriptions, emphasize the need for at least a partial customization of semantic search systems and annotation algorithms. Here a rule-based approach is used to address the special requirements of the mouse anatomy domain.

5.4 GoECDC

The GoECDC system is the result of a cooperation with the European Centre for disease control (ECDC). GoECDC is a semantic search engine with specialized ontological background knowledge. The system identifies concepts using the ECDC-specific ontology. To goal of the project is to highlight the added value of semantically enriched intranet search (Mangold, 2007). Semantic intranet search can help to handle the growing intranets and local data repositories. The new task introduced by such a system is the capability to handle heterogeneous data and text extraction of various document and file types (see Section 2.4.1).

A full system capable of handling the requirements of the ECDC needs to integrate several data sources. This includes documents repositories and internal wikis within the intranet, but also popular commercial systems for sharing documents, such as a Microsoft Exchange. For an overview of the different sources and other components of the architecture see Figure 5.5.

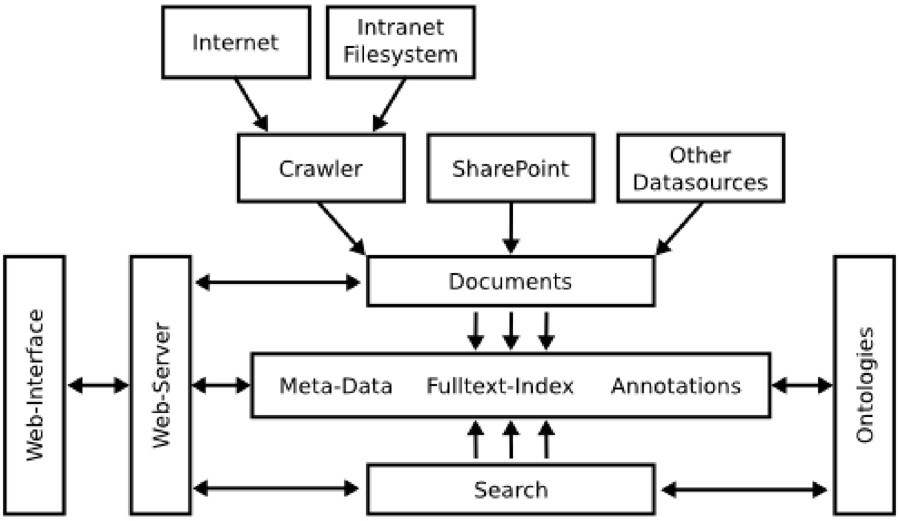


Figure 5.5: GoECDC Architecture Overview

For the technology preview, the implementation focuses on the intranet crawling. The tool used to implement the crawling is the Open Source project Nutch[15] (see Section 2.6 for more details on algorithms used in Nutch). It offers options for using http and file-based protocols to access the content of an intranet. For this technology preview the indexed documents are a sample of an bigger ECDC-provided internal document set.

The decision to implement the system using a crawler is based on the intention to provide an easy upgrade path form the prototype to a large scale intranet search system. In general, a simple file-based monolithic system is easier to implement, but for large application scenarios, a distributed system is required. Only such a system can provide the replication and partitioning required for a reliable and high-performance system, even for large workloads. Similarly, a distributed system can run on a standard hardware cluster and thus reduce the investment needed for installing a system. Furthermore, such a large scale system requires algorithms that determine the importance of a document. For this task existing approaches such as PageRank, HITS or OPIC can be used (see Section 2.6.2). For instance, Nutch uses OPIC to provide an estimate of the global relevance.

One additional advantage is the re-use of existing code for the text extraction from the documents. As described in Section 2.4.1 the content can be provided in many formats. There, a mature code-base can help to reduce the individual pitfalls of each format. Still the reliable extraction of

[15] `http://nutch.apache.org/`

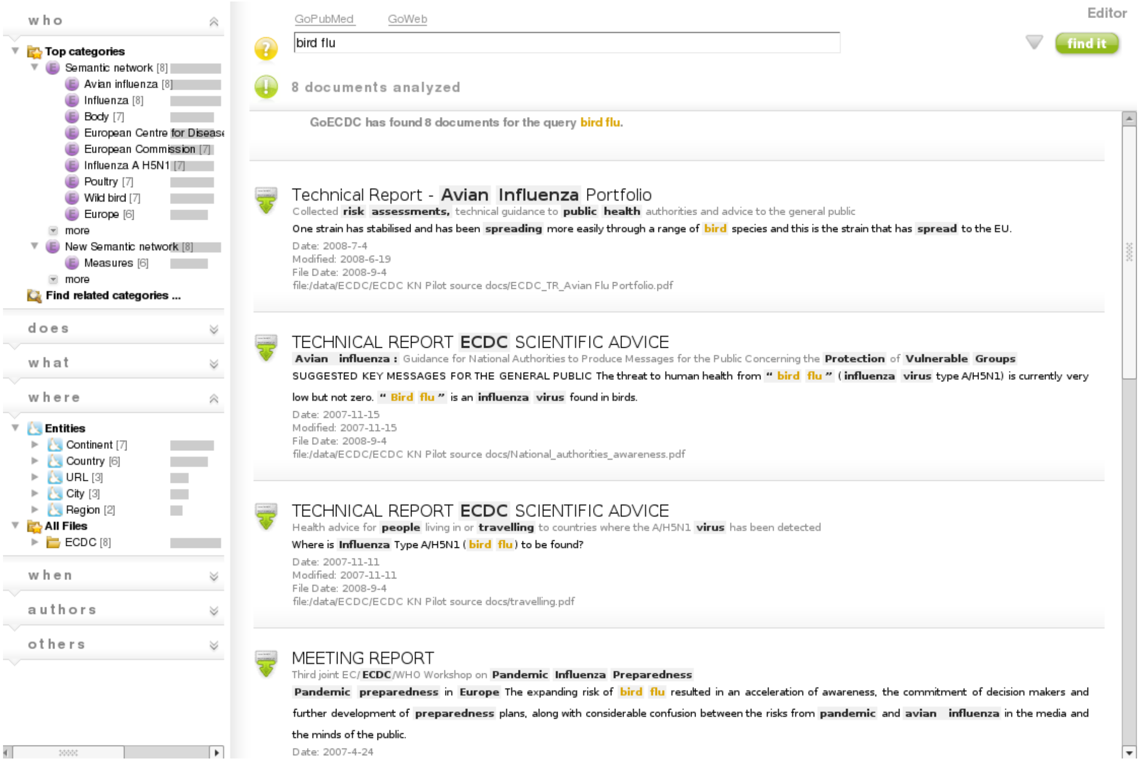
meta data is an open issue. The extraction of authors and dates from specific fields of a document is not sufficient. For instance, if a document was created through PDF export, the meta data in the PDF might declare the export tool as author.

GoECDC retrieves ECDC publications for the search query and sorts relevant information to the 3 top level categories: "Who" — "Does" — "What". The top level categories represent the following roles:

- "Who" — actors
- "Does" — activities
- "What" — topics

They are implemented using ontological background knowledge. The "Who" and "What" categories use the semantic network provided by the ECDC. The background knowledge for the "Does" categories was specifically created for the technology preview. It and describe some actions and activities relevant for the disease and prevention context. This background knowledge was created using the ontology generation tools (Wächter, 2009).

Furthermore, the GoECDC offers three more categories. These categories, "Where" — "When" — "Authors", organize the retrieved results according to places, dates and persons related to the query. These entities are extracted from the textual information and complement the incomplete meta data of the indexed documents. For a screen shot of such a result page see Figure 5.6.



bird flu

8 documents analyzed

GoECDC has found 8 documents for the query bird flu.

Technical Report - Avian Influenza Portfolio
Collected risk assessments, technical guidance to public health authorities and advice to the general public
One strain has stabilised and has been spreading more easily through a range of bird species and this is the strain that has spread to the EU.
Date: 2008-7-4
Modified: 2008-6-19
File Date: 2008-9-4
file:/data/ECDC/ECDC KN Pilot source docs/ECDC_TR_Avian Flu Portfolio.pdf

TECHNICAL REPORT ECDC SCIENTIFIC ADVICE
Avian influenza : Guidance for National Authorities to Produce Messages for the Public Concerning the Protection of Vulnerable Groups
SUGGESTED KEY MESSAGES FOR THE GENERAL PUBLIC The threat to human health from " bird flu " (influenza virus type A/H5N1) is currently very low but not zero. " Bird flu " is an influenza virus found in birds.
Date: 2007-11-15
Modified: 2007-11-15
File Date: 2008-9-4
file:/data/ECDC/ECDC KN Pilot source docs/National_authorities_awareness.pdf

TECHNICAL REPORT ECDC SCIENTIFIC ADVICE
Health advice for people living in or travelling to countries where the A/H5N1 virus has been detected
Where is Influenza Type A/H5N1 (bird flu) to be found?
Date: 2007-11-11
Modified: 2007-11-11
File Date: 2008-9-4
file:/data/ECDC/ECDC KN Pilot source docs/travelling.pdf

MEETING REPORT
Third joint EC/ECDC/WHO Workshop on Pandemic Influenza Preparedness
Pandemic preparedness in Europe The expanding risk of bird flu resulted in an acceleration of awareness, the commitment of decision makers and further development of preparedness plans, along with considerable confusion between the risks from pandemic and avian influenza in the media and the minds of the public.
Date: 2007-4-24

Figure 5.6: GoECDC system with the example query *birdflu*.

5.4.1 Examples Knowledge-based Searching

To demonstrate the capabilities of the semantic search the ECDC provided some use cases. These allow to run and to document a series of test searches in the provided test library of documents.

5.4.1.1 What is done regarding HIV/AIDS in ECDC?

“Who”	—	actors	=	ecdc
“Does”	—	activities	=	?
“What”	—	topics	=	HIV or AIDS

This question can be answered with two different answering strategies.

Option 1 – Query *ecdc*

The first option is to query the GoECDC system with a simple keyword *ecdc*. The GoECDC search finds and analyzes 50 documents for the query.

For this question the system provides the semantic filter to retrieve the answers related to HIV infection. The user can find the filter for the concept “HIV infection” in the following hierarchy: “What” ⇒ Disorder ⇒ Infection ⇒ Infection classified by organism ⇒ Viral infection ⇒ HIV infection. By applying the filter, the document set is boiled down to six articles relating to “HIV infection”. The answers to the question are presented in form of the following extracted titles and snippets:

- **ECDC Programme of Work for 2007**
  “Final version adopted by the Management Board
  Best current examples are the assessment tools used for the influenza preparedness and the AMR visits, but the ***ECDC* intends to make similar tools for other diseases (e.g., *HIV* and tuberculosis**)”

- **meeting Report NET WORKING FOR PUBLIC HEALTH**
  “ECDC Scientific Consultation Group Workshop
  In addition to general and financial information, **participants had an opportunity to learn about and comment upon several *ECDC* ’case studies’ such as *HIV testing*, immunisation schedules**, outbreak alert systems, pandemic flu preparation, ship-borne and other outbreaks.”

- **ECDC Annual Work Programme 2008**
  “Document MB11/5 Approved by the *ECDC* Management Board at its 11th meeting in Stockholm, 13-14 December 2007, including suggestions made at that meeting
  HIV/AIDS, STI, Hepatitis B&C Improved surveillance methods re HIV/AIDS, chlamydia, STI, hepatitis B & C, relevant behaviours; **reports on *HIV/AIDS* epidemiology and on HIV in migrants**; guidance on chlamydia control; **assessment of *HIV testing policies***, practices and outcomes in EU countries; **evaluation of partner referral for STI and *HIV*; review and assessment of *HIV* prevention and control programmes to identify and share best practices; informative website on HIV, STI and viral hepatitis.**”

Option 2 – Query *ecdc HIV*

The second option to answer the question is to query the GoECDC search engine with the query *ecdc HIV*. The system identifies 14 documents relevant for the query. Answers pertaining to the question are available in the prevention branch of the ECDC ontology. This branch can be found in the “Does” category. The prevention branch filter contains 13 documents. The user finds, for example, the following titles and the snippets:

- **Framework Action Plan To Fight Tuberculosis In The European Union**
  “Some factors lowering immune response such as *human immunodeficiency virus (HIV) infection* increase the chances of getting the disease following infection, while *preventive medication* reduces this risk.”

- ***HIV prevention* in Europe: Action, needs and challenges Stockholm, 23 October 2006**
  "***HIV prevention*** ... **diagnosed cases of HIV infection reported in 2005** ... The *HIV transmission* probability in the population has decreased as a result of the lower amount of HIV circulating Source: HIV Monitoring Foundation, Amsterdam"

5.4.1.2 Did anybody recently participate in meetings dealing with measles?

"Who"	—	actors	=	?
"Does"	—	activities	=	write a document or participate on meetings
"What"	—	topics	=	measles

In this case the actors need to be identified in the context of measles. GoECDC finds 4 documents for the query "measles". The answers related to persons can be retrieved by using the category "Authors" and select "person". By filtering with this category, there are two documents are related to person names. Overall, there are approximately 20 person names, that are extracted from these two resulting documents related to measles.

For example the document with the title "Pandemic influenza preparedness planning Report on the second joint WHO/European Commission" contains the sentences with highlighted keywords and person name entities:

- **Report on the second joint WHO/European Commission**
  "*Mila Vucic-Jankovic*, Serbia and Montenegro, representing the Stability Pact countries, gave the (near) eradication of poliomyelitis and *measles* as examples of where collaboration has worked. Pandemic influenza represents a new challenge and governments must put resources into outbreak investigations, antivirals, vaccines and other health measures."

5.4.1.3 What is in the multiannual working plan regarding developing surveillance activities?

The goal is to find a list of activities. The answers are to be extracted from the "Does" category. To retrieve the answers query GoECDC with *Multiannual plan*. This results in a search result set of three documents for the given query. The following answers related to "Does" are found:

- **ECDC Strategic Multiannual Programme 2007–2013**
  "Document approved by the ECDC Management Board at its tenth meeting in Vienna, 14-15 June 2007, including amendments made at that meeting
  This is followed by an analysis of what role ECDC should play in helping the EU and its MS to better *prevent* and *control* those diseases."

- **ECDC Annual Work Programme 2008**
  "Document MB11/5 Approved by the ECDC Management Board at its 11th meeting in Stockholm, 13-14 December 2007, including suggestions made at that meeting
  Disease-specific work Influenza A Seasonal Influenza Portfolio to Council (December 2008); improved *surveillance strategies* for seasonal and pandemic influenza; a research plan for influenza *transmission* and *control*; a burden of disease and foresight approach to influenza; revised pandemic preparedness indicators Tuberculosis Following the new TB Action Plan a joint WHO EURO/ECDC surveillance for TB is in place; new Network of TB reference laboratories; technical report on TB Action Plan; guidance documents on migrants."

The detailed instances and textual evidences for activities are available in the "Does" branch of the ECDC ontology. For the surveillance activities from the question, there are answers regarding to following seven instances:

“Does” ⇒ “Risk” ECDC Strategic Multiannual Programme 2007–2013
“The focus of its work is very complex and involves many *risk factors* that evolve over time.”

“Does” ⇒ “Play” “This is followed by an analysis of what role *ECDC should play* in helping the EU and its MS to better prevent and control those diseases.”

“Does” ⇒ “Need” “Choosing strategies involves identifying possible gaps in knowledge and action; considering ECDC’s mandate to deal with such gaps; and analyzing possible ECDC *actions* and *resources needed to do the work*”

“Does” ⇒ “Immunization”

- *Vaccine-preventable* diseases
  “The European *Vaccination* Expert Committee will be set up and functioning, discussing all issues related to childhood *immunisation* schedules; significant progress will be made towards the Measles and Rubella *immunisation*, including surveillance and outbreak monitoring systems; EU-wide surveillance of Invasive Bacterial Diseases.”
- ECDC Strategic Multiannual Programme 2007–2013
  “Operations will give highest priority to influenza, HIV/AIDS, TB, *vaccine* preventable diseases (notably Measles – to support WHO’s European Regional elimination target) and healthcare associated infections.”

“Does” ⇒ “Protection” “. . . Likewise, in the years to come globalization and the steady increase of business people and tourists travelling daily between Europe and the other regions of the world, will make it essential for ECDC to possess knowledge of potentially dangerous D developments (and countermeasures taken against them) all over the globe. This will be vital *in order to protect* the people of the EU. . . .”

“Does” ⇒ “Prevention” “Sharing of knowledge and experience, as well as scientific cooperation, will in the years ahead require ECDC to build very close and interactive partnerships with institutions and organisations that possess expertise in CD *prevention and control* at global and regional levels.”

“Does” ⇒ “Travel” “Likewise, in the years to come globalization and the steady increase of business people and *tourists travelling* daily between Europe and the other regions of the world, will make it essential for ECDC to possess knowledge of potentially dangerous CD developments (and countermeasures taken against them) all over the globe.”

5.4.1.4 Which diseases are treated in the document set?

To retrieve the answer to this question a global view is required. As GoECDC has indexed and analyzed all documents, the answer can be retrieved directly without a search. The answers are available as part of the disease sub branch. The following diseases are found:

- Influenza
- Pandemic Influenza
- HIV / AIDS
- Hepatitis
- Measles

- Rubella
- Campylobacteriosis
- Salmonellosis
- water-borne infections
- Tuberculosis
- healthcare associated infections

This type of answers can't be found using the traditional keyword search. Only because of the ontological background knowledge it is possible to relate the correct concepts. With simple keyword search it is nearly impossible to find all diseases and corresponding documents.

5.4.1.5 Recommendations to risk groups on influenza

"Who"	—	actors	=	risk groups
"Does"	—	activities	=	recommendations or reports
"What"	—	topics	=	influenza

In this example, the answers to the question can be retrieved using a keyword search and semantic filtering. GoECDC finds 30 documents for the query keyword *influenza*. The semantic filter with the concept "Risk groups" reduces the result set to 12 documents. The remaining search results contain reports such as:

- TECHNICAL REPORT ECDC SCIENTIFIC ADVICE
  "Avian *influenza*: Guidance for National Authorities to Produce Messages for the Public Concerning the Protection of *Vulnerable Groups*"
- "ECDC guidelines to minimise the *risk of humans* acquiring highly pathogenic avian *influenza* from exposure to infected birds or animals
  ECDC guidelines – human exposure to HPAI"
- Technical advice to EU public health authorities
  "6. Interim guidance for national authorities to produce messages for the public concerning the protection of *vulnerable groups* (March 2006)"
- TECHNICAL REPORT ECDC SCIENTIFIC ADVICE
  "Avian Influenza A/H5N1 in Bathing and Potable (Drinking) Water and *Risks to Human Health*"

5.4.1.6 Which groups should be immunized against influenza?

"Who"	—	actors	=	?
"Does"	—	activities	=	immunized
"What"	—	topics	=	influenza

Similar as in the previous example, the answer to the questions are retrieved by a keyword search (query *influenza* – 30 documents) and applying a semantic filter. Filtering with the concept "Immunization" results in 20 documents. The results relating to persons are available in the category "Who" ⇒ "Persons". There are 9 documents are related to person names with answers such as:

- "Regarding non-pharmaceutical interventions, issues of appropriate home care, hygiene and social distancing measures need to be addressed. For both pharmaceutical and non–pharmaceutical interventions, special needs may have to be addressed for certain population groups such as *children*."

- "Infant and *children* seasonal immunisation against influenza on a routine basis during inter-pandemic period.
  Are there indirect benefits to the community (herd immunity, reducing community transmission, etc) from vaccinating *children* against influenza? . . . Routine influenza immunisation of healthy children has been recommended in some countries, to reduce morbidity among children with the potential additional benefit of reducing the spread of disease and thus indirectly protect adults at high risk of severe influenza."

- "For targeting group d) *children*, the trigger should come from early studies on the pandemic virus characteristics, showing that children could be acting as pandemic amplifiers . . . There is a strong argument that children are the most potent spreaders of influenza in the community and that vaccinating them may influence the size and duration of the epidemic overall."

GoECDC is a show case for semantic search in the life sciences. It combines the specialized background knowledge for diseases with general concepts such as persons and locations. With the option to index local repositories, it allows to query a local knowledge base in combination with semantic filtering. Together they provide in GoECDC a first view on what semantic search can provide to a user with special interests (e.g., immunization as a prevention strategy for infectious diseases) and ontology such as the European Centre for Disease Control.

5.5 Summary

The five presented systems provide the technologies to handle diverse scenarios for semantic search (*Open Problem 3*). The GoCell system provides full text handling from complex XML-based data sources. MousePubMed is a special interest system (mouse anatomy) and handles its specific information need with customized rule-based annotators. The evaluation of MousePubMed shows that this is a promising but not perfect approach. In contrast, in the scenario of patent search with the two presented systems of GoFreePatentsOnline and GoPIZ, the target and source is content wise more general and heterogeneous. First results indicate that semantic technologies can help to handle this challenge. Finally, the GoECDC is a combination of special interests (disease prevention) with heterogeneous data. There the advantage is directly presented with several use-cases of intranet search and customized background knowledge.

The five systems also highlight the current limits of such systems. All systems with large text quantities have the problem of presenting it to the user in an easy and understandable way. Adding further semantic information can lead to an overflow of information in the user interface. This can be addressed by reducing the text amount in the first result set with configurable and or customized fields. Further limitations are the technical aspects introduced due to the special nature of a domain and related ontological background knowledge. For MousePubMed, the specialized domain of mouse anatomy and the related ontology required a customization of the annotation. Patents in contrast are so diverse that complete background knowledge for all patents is improbable. Furthermore, the textual part of patents is highly complex in terms of length, formulation, and intended meaning.

In general semantic search can be applied to many areas and help to improve the knowledge search and exploration, but custom solutions are still required. The adaption of background knowledge, the loading of an additional ontology or the usage of data schema mapping is currently not sufficient. An adaption and evaluation of a system is required.

CHAPTER 6

TOWARDS ANSWERING QUESTIONS WITH SEMANTIC INDEXING

Semantic Indexing is an alternative approach to combine keyword-based and semantic search (*Open Problem 1*). In contrast to the solution presented with GoWeb in Chapter 3, the semantic annotation of the content is a preprocessing step. We use question answering to evaluate the semantic index, as we did for GoWeb, GoECDC in Section 5.4.1, MousePubMed in Section 5.3.2.1, and the concept of semantic browsing as described in Section 4.1.3. All these systems help to answer questions in a semi-automatic way. We evaluate the semantic index in the biomedical domain. The user may wish to consult Section 2.2 for more background information on question answering.

6.1 Semantic Index using a Sentence-based Approach

The answers presented to a user are text fragments or sentences. In the general workflow for question answering the answers are extracted in a secondary step after identifying relevant documents (see Section 2.2.2 and Figure 2.6). In contrast the proposed and tested semantic index system starts with the extraction of such text passages. The search system pre-annotates all documents, blocks, spans, and sentences. Additionally, the system uses the ontological background knowledge and structural text features. The goal is to provide a system that allows to search for facts of a certain type, for instance, diseases associated with a certain protein.

To prepare the corpus for the semantic index, the HTML documents have been preprocessed. The first step is separating the text from the markup, including the stripping of HTML-tags and replacing of in-line images with their `alt`-values. A drawback of commonly used HTML and ASCII character set is the unavailable option for Greek letters, such as α or β. They are often inserted as small in-line images. This process leads to information-loss, for instance, italics may denote gene names instead of protein names, super- and sub-scripts are used in formula and for footnotes. The text extraction must also keep an association to the corresponding positions in the document, to be able to provide the original content and for a proper evaluation.

The next step in the preparation is the labeling of the text extracts with additional structural information. For instance, references at the end of the document are usually not valid answer statements, whereas extracts from the title or abstract might be more relevant for document retrieval. These distinctions can be gained from the HTML markup tags. For instance, if there is a heading "References" than it's likely that the following paragraphs or lists contain the citations.

Meta information, such as author names, is retrieved via the PubMed records. Also existing MeSH annotations are associated with the document. The next step is annotating the article with concepts from the background knowledge according to the extracted passages. The background knowledge includes the GeneOntology (GO), Medical Subject Headings (MeSH) and protein and gene names. For the annotation of GO and MeSH the algorithms provided by GoPubMed are employed (Doms and Schroeder, 2005). The information about sentence splitting is also retained from this step. The approach by Hakenberg et al. (2008) is used for the protein and gene name annotation.

The information gained in the preparation is used to extract facts from text. In our model a sentence is the scope of an individual fact. Keywords, annotations and entities co-occurring in one sentence are all potential facts. This assumption is reflected in the design of the semantic index that is based on the Lucene Java library[1]. During the index creation sentence are included as an individual document with the following fields:

- DocumentId – here PubMedID
- TextRange – for retrieving the original content
- Content – extracted text from the HTML
- Span – corresponding legal span position, if available
- Level – indicator whether the entry is a sentence, block, or document
- DocumentStructure – Title, Abstract, Other, Table, Image, Reference;
- Separate fields for MetaData: Date, Authors, Affiliation, Journal, Country
- Annotations (concepts from GO and MeSH)
- Proteins

To favor headings as a potentially more relevant part of a document, the relevance ranking adds to the document a boost-factor of two to increase the original match score.

The annotations, meaning the identifier of the concepts or headings, need to be stored in such a way, that it is possible to search for a top level concept and all of its sub concepts for a given relation type. In case of MeSH this would be, for instance, the top level term “Disease” and all headings contained in the sub-tree. Similarly, for example with “biological_process” in GO, all child concepts having a `is_a` relation to the concept, including transitivity, should be matched. To implement this, a simple OR-query for all related concepts from the sub-tree is too inefficient. Also the number of clauses in a query is often limited for efficiency purposes. Expanding the index is an alternative to reduce the query size. For instance, it is possible to store all parent identifiers for a concept. This approach can be extended to include the full transitive closure. Using the tree-like structure to enable efficient querying of sub trees in ontologies is a compromise. For MeSH there is already by default such a system, the tree numbers. For terms occurring in multiple sub-trees, there exist several tree numbers. A simple prefix search in tree numbers allows to identify all sub concepts of a branch. For directed acyclic graphs (DAG), an efficient way has been proposed by Trißl and Leser (2005, 2006) using a pre/post traversal strategy for assigning numbers to nodes. Using this scheme, a range can represent a sub-graph and replace the OR clauses in a search.

Facts are not always stated in a sentence; instead, they may be spread over larger blocks in a document. Also the PubMed-MeSH headings are not associated with single text passages, they relate to the complete article. Thus, the index contains the block level and the whole article as indexed individual.

The redundant indexing leads to multiple hits per PMID and/or sentence and requires a custom scoring schema in Lucene and a hit post-processing. The customized scoring boosts hits on lower

[1] `http://lucene.apache.org/java/docs/`

levels, giving sentences a higher boost than blocks and blocks a higher boost than document level hits. The post-processing deals with the duplication introduced during the index creation. It clusters multiple hits for one PMID into appropriate groups. As there might be answers with regard to multiple aspects, the grouping of extracted facts is done according to identified concepts or, if not assigned, to a keyword.

The answers provided by the system in case of sentence or block level hits have direct textual evidence. For document level hits the PMID is returned. Additionally, the identified answering concept is provided. Multiple hits are available as sub evidences and sorted by match value in descending order. The provided sub evidences may help reinforce the answer with additional textual context.

6.2 TREC Genomics 2006 Question Answering Benchmark

As part of the the Text REtrieval Conference (TREC)[2], the TREC Genomics track concentrates on question answering for the biomedical domain. For the 2006 installment, a corpus of full text documents was assembled. The documents are extracted from journals focusing on the biological domain. The documents were collected by a crawling step and stored in HTML-format. The document selection contains articles from the electronic publisher Highwire Press[3]. There are $162,259$ documents from 49 journals in total, occupying about 3 GByte while compressed and 12.3 otherwise. For the whole corpus a mapping of document URL to PubMed identifier (PMID) is provided. This mapping is correct for 99% of the cases. An important tool for handling the corpus are the so called "legal spans". The legal spans describe valid blocks of text in the original HTML document. Such spans are defined as any non-empty text in between HTML paragraph tags `<p` or `</p`. If no ending tag exists the next `<p` ends the current span. The corpus contains $12,641,127$ legal spans. The spans are represented as pairs of character-based offset in the source file and length of the span. These legal spans are used, for instance, in the definition and evaluation of the benchmark. They also help to skip irrelevant parts of HTML-documents, such as the HTML-header or JavaScript. For more details on the corpus, including the license agreement to download and use of corpus data, see `http://ir.ohsu.edu/genomics/2006data.html`.

The benchmark consists of 28 official questions, which are called topics; two evaluation standards are available (Hersh et al., 2006) for each topic. The "gold-standard" contains altogether $3,451$ passages which have been marked as valid answers. The "raw relevance judgments" contain $28,799$ annotated passages with $24,934$ marked as "NOT" relevant.

This benchmark has three levels for evaluating and measuring the performance: passage retrieval, aspect retrieval, and document retrieval. The passage-level retrieval is character-based and helps to assess the quality of the extraction and the distillation of text-fragments as answers. The aspect-level allows to assess the quality with respect to the answer space. It helps to test whether all available topics for answers are present. The document-level deals with the document retrieval, finding all documents containing a valid answer for the given topic.

Starting with the original document size of 12.3 GByte, the additional information of annotations, proteins, and meta data, combined with the redundant storage, leads to an semantic sentence-based index with a size of 46 GByte.

6.3 Evaluation

The evaluation of the sentence-based semantic index with the TREC Genomics 2006 benchmark starts with mapping the questions to typed query parameters. See Appendix B.1 for a listing of the used mapping. A listing of answers provided by the system for the 28 topics is available in

[2] `http://trec.nist.gov/`

[3] `http://highwire.stanford.edu/`

Topic ID	proposed PMIDs	True Positives	False Positives	Unknown PMIDs	All Positive Examples	All True Negatives
160	180	151	20	9	198	125
161	120	26	40	54	28	486
162	34	0	1	33	10	571
163	1160	149	148	863	166	269
164	0	0	0	0	7	447
165	15	9	3	3	10	442
166	3	2	1	0	8	612
167	23	9	8	6	56	360
168	62	42	18	2	43	280
169	142	41	39	62	64	422
170	12	4	5	3	4	581
171	6	0	2	4	34	552
172	3531	232	276	3023	234	356
173	0	0	0	0	0	557
174	11	1	6	4	19	366
175	27	10	15	2	19	455
176	36	9	23	4	11	558
177	0	0	0	0	2	624
178	163	2	51	110	3	660
179	37	7	26	4	12	434
180	5	0	4	1	0	429
181	410	175	34	201	220	122
182	1	1	0	0	75	411
183	0	0	0	0	16	424
184	0	0	0	0	2	642
185	13	4	9	0	13	490
186	170	107	22	41	192	230
187	0	0	0	0	3	579
Sum	6161	981	751	4429	1449	12484

Table 6.1: Document-level Retrieval Performance of the Sentence-based Index — Absolute values.

TopicID	TP-Rate	FP-Rate	Unknown-Rate	Precision	Recall	F-Score
160	83.9%	11.1%	5.0%	88.3%	76.3%	81.8%
161	21.7%	33.3%	45.0%	39.4%	92.9%	55.3%
162	0.00%	2.9%	97.1%	0.0%	0.0%	0.0%
163	12.8%	12.8%	74.4%	50.2%	89.8%	64.4%
164	—	0.0%	—	—	0.0%	0.0%
165	60.0%	20.0%	20.0%	75.0%	90.0%	81.8%
166	66.7%	33.3%	0.0%	66.7%	25.0%	36.4%
167	39.1%	34.8%	26.1%	52.9%	16.1%	24.7%
168	67.7%	29.0%	3.2%	70.0%	97.7%	81.6%
169	28.9%	27.5%	43.7%	51.3%	64.1%	56.9%
170	33.3%	41.7%	25.0%	44.4%	100.0%	61.5%
171	0.00%	33.3%	66.7%	0.0%	0.0%	0.0%
172	6.6%	7.8%	85.6%	45.7%	99.2%	62.5%
173	—	0.0%	—	—	—	—
174	9.1%	54.6%	36.4%	14.3%	5.3%	7.7%
175	37.0%	55.6%	7.4%	40.0%	52.6%	45.5%
176	25.0%	63.9%	11.1%	28.1%	81.8%	41.9%
177	—	0.0%	—	—	0.0%	0.0%
178	1.2%	31.3%	67.5%	3.8%	66.7%	7.1%
179	18.9%	70.3%	10.8%	21.2%	58.3%	31.1%
180	0.0%	80.0%	20.0%	0.0%	0.0%	0.0%
181	42.7%	8.3%	49.0%	83.7%	79.6%	81.6%
182	100.0%	0.0%	0.0%	100.0%	1.3%	2.6%
183	—	0.0%	—	—	0.0%	0.0%
184	—	0.0%	—	—	0.0%	0.0%
185	30.8%	69.2%	0.0%	30.8%	30.8%	30.8%
186	62.9%	12.9%	24.1%	83.0%	55.8%	66.7%
187	—	0.0%	—	—	0.0%	0.0%
Avg:	34.0%	26.2%	32.6%	44.9%	43.8%	34.1%

Table 6.2: Document-level Retrieval Performance of the Sentence-based Index — Rates and Measures.

Appendix B.2. The results for document level retrieval on the TREC Genomics 2006 benchmarks are available in Table 6.1 and 6.2.

The approach of semantic indexing can provide answers but is limited to the available entities. For instance, for question #177 regarding the "Bop-Pes" interaction the system does not provide an answer because the system did not identify the articles containing both proteins. The same problem occurs for topic #164 and "Nurr-77". Furthermore, insufficient background knowledge may lead to missed hits. For the topic #183 the search for "tracheal development" is currently not very well modeled in GO and MeSH. The hits marked as relevant for topic #183 are sometimes very general. For instance, the gene *NM23* is only referenced in an indirect way via the function of the gene. The corresponding passages from the gold-standard look as follows:

- **PMID:** 11532922 **Span:** 124161-140
  "Development of the Drosophila tracheal system occurs by a series of morphologically distinct but genetically coupled branching events"
- **PMID:** 12466193 **Span:** 90654-82,
  PMID: 12930776 **Span:** 93936-82,
  PMID: 14681183 **Span:** 78237-84,
  PMID: 15269170 **Span:** 75762-90
  "Genetic control of branching morphogenesis during Drosophila tracheal development"
- **PMID:** 14597571 **Span:** 90816-81
  "Genetic control of epithelial tube size in the Drosophila tracheal system"

This feature of the benchmark combined with described text mining system explains that there were no answers found for the topics #164, #177, #183, #184, and #187. On the positive side the system did not propose an answer if the benchmark does not contain an answers, as for question #173. An exception is topic #180, where the system proposed known false positives.

For questions with many answers (#160,#163,#169,#172,#181,#186) there are sufficient correct answers, and the number of known false positives is in average 13.4%, see Table 6.2 for individual values. There, the synonyms and included sub-concepts expand the result space which explains the large rate of unknown answers (32.64%, maximum of 97.06% for topic #162). For the TREC Genomics 2006 benchmark, not all text-passages of all documents have been examined by the reviewers (Hersh et al., 2006). Only the passages which were submitted to the original contest have been judged, thus, new passages to the benchmark are an unknown quantity.

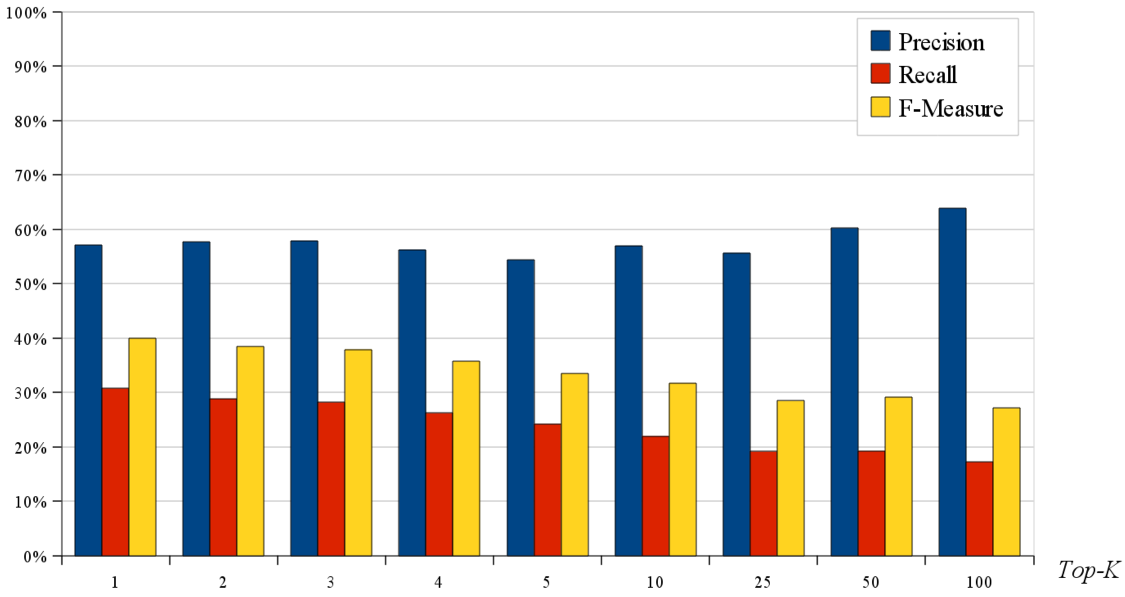
The system may return more than one answer for a question. The ideal ranking should return the valid answers as the top hits. The ranking can be compared with top-k lists. Such a list is defined as follows: the k is a non-negative integer and it represents the number of the first k results from the result list. For $k = 1$, for instance, only the first and highest ranking result is regarded. For $k = 5$ the list with first five results are analyzed.

In the context with TREC Genomics 2006 benchmark, the top-k list is tested, if it contains correct, false, or unknown answers. The block level-based comparison results for question answering with the index are shown in Figure 6.1.

The top-k ranking evaluation illustrates that the used ranking of the system provides answers with an equal distribution of correct and known false positives. The current ranking of the system has no option to employ additional information to improve the behavior and top-k ranking. This could be, for instance, feedback by users or other training data in form of known false positives. The problem of ranking results was not directly addressed in this semantic indexing system.

The limits of the semantic index is also reflected in the Mean Average Precision (MAP) score of 0.17 for question answering on span level. The MAP score is the standard way of comparing the results of different rankings in the TREC Genomics competition. The achieved MAP score is in range of the baseline systems submitted to the competition (Hersh et al., 2006). The top result of Zhou et al. (2006) with MAP $= 0.54$ indicates that the current semantic index is not a complete

a)



b)

Top-k	Positive	Negative	Unknown	Precision	Recall	F-Measure
1	8	6	8	57.1%	30.8%	40.0%
2	15	11	17	57.7%	28.9%	38.5%
3	22	16	26	57.9%	28.2%	37.9%
4	27	21	36	56.3%	26.2%	35.8%
5	31	26	47	54.4%	24.2%	33.5%
10	53	40	102	57.0%	22.0%	31.8%
25	98	78	238	55.7%	19.1%	28.5%
50	159	105	409	60.2%	19.3%	29.2%
100	223	126	736	63.9%	17.2%	27.2%

Figure 6.1: Span Level Question Answering Performance for different top-ks
a) Historgram illustrating precision, recall, and F-measure for the different top-ks
b) Table for absolute counts of positive, negative, and unknown answers and for calculated precision, recall and f-measure for different top-ks

and fully comparable question answering system. The weak point of the system is not the index itself but the additional components required for a question answering system (cf. Section 2.2). The more successful systems have a more complex query parsing and expansion system and the employed ranking schemas are far more complex and include, as mentioned above, a feedback system.

6.4 Conclusion

Semantic indexing can be used to answer questions (*Open Problem 1*). The results show that the semantic-index-based question answering system proposes valid answers as top results, but also known false positives. The system uses a simple co-occurrence matching of text mined concepts. The evaluation of the applicability to question answering with the TREC Genomics 2006 competition as benchmark shows that the index provides valid answers. The results also indicate that the performance is not yet up to the current state-of-the-art question answering systems. With a MAP score of 0.17, the semantic index offers a performance comparable to base line question answering systems. This can be attributed to the fact, that the system consists primarily of a semantic index plus some minor parts required for question answering.

The usability of the system is limited by the number of false positive answers. An important technical limit is the additional storage space overhead required for the redundant search levels. This might be addressed by changing the current indexing schema. An alternative schema could treat the terms as special tokens in a normal text stream. In this case the index could be build without additional fields for annotations. Similarly, instead of storing each sentence, block and document as separate entries sentence delimiters could be introduced in the index. To maintain the feature of phrase search, the sentence delimiters need a logical length of zero, meaning the token count for the adjacency checks is not increased by the delimiters tokens. This modification allows to search for sentences or longer passages without the need a redundant storage. An inverted file index provides the required flexibility for such an approach (Section 2.6.1.1).

Nevertheless, the concept of the semantic index is shown to be an alternative implementation for semantic search. In contrast to GoWeb (see Section 3), the additional semantic annotation post-processing of search results is not required. The complete annotation process is done in a pre-processing step during the index creation. Using such an index the annotation does not impact the runtime behavior for live question answering. This advantage is countered by the required resources to create and maintain such an index. The techniques and algorithms to handle the crawling exist, and index even PetaByte data collections. As described in Section 2.4.3 and 2.6.1, there are the Map-Reduce pattern, distributed file systems, and inverted-file indices. Still, the amount of hardware and human resources is substantial. Even legal concerns might play a role as content providers can restrict their sites that only certain search engines are allowed to crawl their content (see also Section 2.4.3.1).

Overall, the approach of reusing results of existing search engines for GoWeb is a practical trade-off.

CHAPTER 7

CONCLUSION AND FUTURE WORK

Semantic search and browsing is a powerful tool for retrieving and analyzing information. Searching and analyzing are common tasks for the research and even practical applications in the biomedical domain. Current search systems are either traditional keyword-based systems with limited semantic awareness or as with semantic-enabled systems have a limited database, as they mostly rely on existing semantic content.

This thesis introduces with GoWeb a bridging approach to combine the advantages of both ideas (**Open Problem 1**). GoWeb combines the simple keyword-based search and semantic filtering, see Chapter 3. GoWeb uses a traditional search engine and in a post-processing step semantically annotates the search results. The radix tree algorithms for annotating the content reflect the need for speed and handling of unstructured and possibly malformed text snippets in this interactive system. The radix tree scales to the needs of large word dictionaries with more than one million entries. However, the radix tree offers only limited capabilities in terms of rule-based label extension, as compared to regular expressions. The radix tree is used to annotate the snippets with the biomedical domain specific background knowledge, the GeneOntology and the Medical Subject Headings. The radix tree is also used to identify relevant entities like person and company names.

The GoWeb system is evaluated using three benchmarks in context of the biomedical domain (**Open Problem 2**) with success rates of up to 79%. To improve the search experience semantic hyperlinks are introduced. They offer additional information based on the identified concepts and entities and thus help to explore the knowledge space. Semantic hyperlinks are the basis for semantic browsing, see Chapter 4. With semantic browsing it is possible to complement question answering and to meet the research tasks in the biomedical domain. For instance, semantic browsing connects the relevant web services and workflows to an identified entity, e.g., protein or gene names. To identify relevant web services, a semantic annotation of web services is required. Such semantic web service may also help to address the problem of automatic workflow creation.

Semantic hyperlinks can be provided directly in an application like in the GoPubMed-Extended or the links to Wikipedia in GoWeb. A limit is that semantic links have to be implemented by the content provider. Alternatively, the semantic links can be added in a post processing step. This dynamic link layer can be added to existing resources using standard technologies, i.e. web proxy or web browser extension. GoWeb uses a hybrid approach by combing a simplified proxy and highlighting in the browser using JavaScript. The implemented solution minimizes the proxy server load and keeps the rendering task in the browser. In GoWeb, this proxy provides the annotation and highlighting of websites. The evaluation of semantic browsing shows that it is helpful and improves the search experience.

Relevant facts can be found in various data sources other than the web. As interesting data source, and not only for the biomedical domain, there are patents, XML documents, literature databases, or an intranet repository. The application of semantic search to each of these data source is described in Chapter 5. Similarly, scientific questions may need more specialized ontologies and background knowledge to reflect the research sub-domain.

The integration of scientific full text documents in XML format introduced the need for processing of complex XML schemas for text and meta data extraction. The presentation of full text documents in GoCell with semantic annotations is an open issue. The combination of text and all available annotations leads to an information overflow in presentation. For patents in the GoPatents search engine, which can have even longer texts, this is a similar problem. An additional task in patent search is the user need for completeness in terms of recall. A single missed patent may invalidate the research. The semantic approach addresses this issue by using synonyms and related concepts from the background knowledge.

Other special research interests, as with mouse anatomy, require a customized system for literature search. There an anatomy ontology is applied to the literature data base PubMed to create the MousePubMed semantic search engine. An combination of specialized research interest and custom data sources is the intranet search engine GoECDC for the European Centre for Disease Control. There the extraction of texts and meta data is a critical step due to the different document formats. Additionally, accessing the documents may require a crawler or data extraction out of commercial document sharing systems.

The developed prototypes address the portability issue for the data source adaptors and annotator (**Open Problem 3**). All systems need an adaption step for scaling and data extraction. Special information needs lead to the development of application specific annotators and rules. This customization can use hand-coded rules, as with MousePubMed, or other machine-learning techniques. The prototypes also show that a system handling all resources and taking advantage of all provided background knowledge from the document source is not yet available.

Semantic indexing (see Chapter 6) addresses the **Open Problem 1**. The evaluation using question answering from the biomedical domain benchmark TREC Genomics 2006 demonstrates the general capabilities of this approach. It shows that a semantic index on its own is not a full state-of-the-art question answering system. The implemented semantic index performs similar to a base line question answering system. The main limitation is currently the number of false positive hits. Furthermore, to apply it for large data sets, the discussed index update with a more compact index structure is required. Still, the decision to maintain a semantic index versus a more lightweight approach, as by GoWeb, depends strongly on the data source and available resources. The advantage of GoWeb is, that it reuses existing search infrastructure, whereas a semantic index requires an additional one.

There is a drawback in the presented semantic systems. From the user perspective, all systems add an additional layer of complexity. A user has to understand how to use the additional options and semantic filtering system. This holds true for the search engines systems such as GoWeb, GoPatents, MousePubMed, GoCell, GoECDC, or GoPubMed-Extended. Also semantic browsing is complex with its options to link the diverse eScience infrastructure including composed web service workflows. In general, a well polished user interface is an integral part for a successful user experience.

An approach to minimize the entry gap and a promotion for semantic search is the Find42.com project. It is a system directly based on GoWeb, but in contrast to, it focuses on entities without the usage of an additional ontology. With this simplification, Find42.com tries to introduce the user to the idea of filtering large results sets using additional features. Furthermore, this system is used as show case for the commercialization of technologies developed in course of this thesis.

The main contribution of this thesis is the development of the semantic search engine GoWeb for the biomedical domain, which is available online: gopubmed.org/goweb. This includes a newly developed annotation algorithm using a radix tree. Semantic search with GoWeb improves biomedical question answering compared to keyword-based search with Google to a success rate of 79%. To complement this, semantic browsing and semantic indexing are implemented and demonstrated. Furthermore, semantic search is applied to other data sources such as patents, full text articles, and intranet repositories.

7.1 Future Work

A future improvement for the semantic annotation of GoWeb could be the development of a word sense disambiguation suitable for this application. In contrast to existing approaches of word sense disambiguation, it has to be addressed, that any implemented approach has to work with irregular and short text fragments. Currently, there is no such corpus for training. The application of word sense disambiguation can be used for the annotation with ontology concepts but also for entity recognition. There the disambiguation can be applied to names with common words or to distinguish between different entity types with equal names, e.g., person name or company.

A second field of future work is the improvement for patent search. There the usage of a larger corpus with continuous updates is an important milestone. A long term goal is an up-to-date search for a complete data base, such as the European Patent Office patent data base. Another milestone is the integration of the existing patent classification schemes, like the USPC or ECLA, as navigational resource. A further technical task is the support for multiple and cross language annotation and search. For the usability of the system the grouping of patents belonging to a patent family might help to reduce the result set size.

For patent search and question answering the approach of the semantic indexing has to be reworked. There an inverted file index with in-lined terms and sentence delimiters will reduce the index size.

Such an index update can be the basis for creating a competitive question answering system. There interesting questions are to use also more complex annotation algorithms with ontology-based word sense disambiguation. Also to improve the relation extraction a natural language processing might help to reduce invalid associations. Also the extracted type of relation might be offered as search criteria or it may be used to rank the relations according to confidence.

ACKNOWLEDGMENTS

I would like to acknowledge my supervisor Prof. Michael Schroeder. I am most grateful, that he provided me the opportunity to work in the interesting and challenging field of bioinformatics. I would like to thank him for the fruitful research environment with the successful combination of strong support, guidance and academic freedom.

Special thanks to Thomas Wächter and Andreas Doms as friends and colleagues. They provided advice, knowledge, and professionalism right from the beginning. I am also thankful to Dimitra Alexopoulou, Conrad Plake, and Loïc Royer for the joint work, discussions and publications.

A huge acknowledgement goes to the Transinsight Team, the collaboration with team provided the opportunity to implement the ideas in real life projects. In particularly, I am grateful to Matthias Zschunke and Jan Mönnich, which provided valuable discussions with regards to implementation and application design. To Liliana Barrio-Alvers and Michael Alvers, I am thankful for testing the application and providing feedback and improvement ideas from an enterprise point of view.

For spending a lot of their time to proof read my thesis, I am especially grateful to Sunna Torge, George Tsatsaronis, Rainer Winnenburg, Matthias Reimann, and Thomas Wächter.

Altogether, I would like to thank the whole bioinformatics group for providing a creative and friendly environment. The discussions always provided valuable feedback and inspirations. A special mention also goes to Mandy Gläßer, who always expertly solved the administrative details. Furthermore, special thanks also go to the computer department, which provides and maintains the necessary computer infrastructure.

Finally, I would like to thank my family, which has supported me during the whole time of my studies. Special gratitude goes to my mother and father for all the offered encouragements.

BIBLIOGRAPHY

Online Mendelian Inheritance in Man, OMIM (TM). McKusick-Nathans Institute of Genetic Medicine, Johns Hopkins University (Baltimore, MD) and National Center for Biotechnology Information, National Library of Medicine (Bethesda, MD).
URL `http://www.ncbi.nlm.nih.gov/omim/`.

Abiteboul, Serge; Preda, Mihai; and Cobena, Gregory. **Adaptive on-line page importance computation**. In *WWW '03: Proceedings of the 12th international conference on World Wide Web*, pages 280–290, New York, NY, USA, 2003. ACM. ISBN 1-58113-680-3. doi: 10.1145/775152.775192.

Adida, Ben and Birbeck, Mark. **RDFa Primer – Bridging the Human and Data Webs**. W3C Working Draft, June 2008.
URL `http://www.w3.org/TR/2008/WD-xhtml-rdfa-primer-20080620/`.

Afzal, Hammad; Stevens, Robert; and Nenadic, Goran. **Towards Semantic Annotation of Bioinformatics Services: Building a Controlled Vocabulary**. In *3rd International Symposium on Semantic Mining in Biomedicine*, 2008.

Agatonovic, Milan; Aswani, Niraj; Bontcheva, Kalina; Cunningham, Hamish; Heitz, Thomas; Li, Yaoyong; Roberts, Ian; and Tablan, Valentin. **Large-scale, parallel automatic patent annotation**. In *PaIR '08: Proceeding of the 1st ACM workshop on Patent information retrieval*, pages 1–8, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-256-6. doi: 10.1145/1458572.1458574.

Agirre, Eneko and Edmonds, Philip, editors. *Word Sense Disambiguation: Algorithms And Applications*. Springer Verlag, 2006.

Aho, Alfred V. and Corasick, Margaret J. **Efficient string matching: an aid to bibliographic search**. *Communications of the ACM*, 18(6):333–340, 1975. ISSN 0001-0782. doi: 10.1145/360825.360855.

Akkiraju, Rama; Farrell, Joel; Miller, John; Nagarajan, Meenakshi; Schmidt, Marc-Thomas; Sheth, Amit; and Verma, Kunal. **Web Service Semantics - WSDL-S**. W3C Member Submission, November 2005.
URL `http://www.w3.org/Submission/WSDL-S/`.

Alexopoulou, Dimitra; Andreopoulos, Bill; Dietze, Heiko; Doms, Andreas; Gandon, Fabien; Hakenberg, Jörg; Khelif, Khaled; Schroeder, Michael; and Wächter, Thomas. **Biomedical word sense disambiguation with ontologies and metadata: automation meets accuracy**. *BMC Bioinformatics*, 10:28, 2009. doi: 10.1186/1471-2105-10-28.

Alonso, Gustavo; Casati, Fabio; Kuno, Harumi; and Machiraju, Vijay. *Web services : concepts, architectures and applications*. Springer, Berlin; Heidelberg, 2004.

Altintas, Ilkay; Berkley, Chad; Jaeger, Efrat; Jones, Matthew B.; Ludäscher, Bertram; and Mock, Steve. **Kepler: An Extensible System for Design and Execution of Scientific Workflow**. In *Proceedings of the 16th International Conference on Scientific Database Management (SSDBM 2004)*, pages 423–424, Santorini Island, Greece, June 2004. IEEE Computer Society. doi: 10.1109/SSDM.2004.1311241.

Altschul, Stephen F.; Gish, Warren; Miller, Webb; Myers, Eugene W; and Lipman, David J. **Basic local alignment search tool.** *Journal of Molecular Biology*, 215(3):403–410, Oct 1990. doi: 10.1006/jmbi.1990.9999.

Amir, Arnon; Srinivasan, Savitha; and Efrat, Alon. **Search the audio, browse the video: a generic paradigm for video collections**. *EURASIP Journal on Advances in Signal Processing*, 2003(1):209–222, 2003. ISSN 1110-8657.

Ananiadou, Sophia. **A methodology for automatic term recognition**. In *Proceedings of the 15th conference on Computational linguistics*, pages 1034–1038, Morristown, NJ, USA, 1994. Association for Computational Linguistics. doi: 10.3115/991250.991317.

Andrenucci, Andrea and Sneiders, Eriks. **Automated Question Answering: Review of the Main Approaches**. In *ICITA '05: Proceedings of the Third International Conference on Information Technology and Applications (ICITA'05) Volume 2*, pages 514–519, Washington, DC, USA, 2005. IEEE Computer Society. ISBN 0-7695-2316-1. doi: 10.1109/ICITA.2005.78.

Androutsopoulos, Ion; Ritchie, Graeme D.; and Thanisch, Peter. **Natural Language Interfaces to Databases - An Introduction**. *Natural Language Engineering*, 1:29–81, 1995.

d'Aquin, Mathieu; Baldassarre, Claudio; Gridinoc, Laurian; Angeletou, Sofia; Sabou, Marta; and Motta, Enrico. **Characterizing Knowledge on the Semantic Web with Watson**. In *Workshop on Evaluation of Ontologies and Ontology-based tools, 5th International EON Workshop, collocated with the International Semantic Web Conference (ISWC'07), Busan, Korea*, 2007.

Ashburner, Michael; Ball, Catherine A.; Blake, Judith A.; Botstein, David; Butler, Heather; Cherry, J. Michael; Davis, Allan P.; Dolinski, Kara; Dwight, Selina S.; Eppig, Janan T.; Harris, Midori A.; Hill, David P.; Issel-Tarver, Laurie; Kasarskis, Andrew; Lewis, Suzanna; Matese, John C.; Richardson, Joel E.; Ringwald, Martin; Rubin, Gerald M.; and Sherlock, Gavin. **Gene Ontology: tool for the unification of biology. The Gene Ontology Consortium**. *Nature Genetics*, 25(1):25–9, May 2000. doi: 10.1038/75556.

Atkinson, Kristine H. **Toward a more rational patent search paradigm**. In *PaIR '08: Proceeding of the 1st ACM workshop on Patent information retrieval*, pages 37–40, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-256-6. doi: 10.1145/1458572.1458582.

Azuaje, Francisco; Wang, Haiying; and Bodenreider, Olivier. **Ontology-driven similarity approaches to supporting gene functional assessment**. In *Proceedings of the ISMB'2005 SIG meeting on Bio-ontologies*, pages 9–10, Detroit, June 21 2005.

Baeza-Yates, Ricardo and Castillo, Carlos. **Crawling the infinite Web: five levels are enough**. In *Proceedings of the third Workshop on Web Graphs (WAW)*, Rome, Italy, October 2004. Springer LNCS. URL `citeseer.ist.psu.edu/baeza-yates04crawling.html`.

Baeza-Yates, Ricardo and Ribeiro-Neto, Berthier, editors. *Modern Information Retrieval*. Addison Wesley Longman Publishing Co. Inc., 1999.

Baker, Christopher J. and Witte, René. **Mutation Mining–A Prospector's Tale**. *Information Systems Frontiers*, 8(1):47–57, 2006. ISSN 1387-3326. doi: 10.1007/s10796-006-6103-2.

Baker, Christopher J. O. and Cheung, Kei-Hoi, editors. *Semantic Web: Revolutionizing Knowledge Discovery in the Life Sciences*. Springer, December 2006. ISBN 0387484361. doi: 10.1007/978-0-387-48438-9_17.

Baldock, Richard A.; Bard, Jonathan B. L.; Burger, Albert; Burton, Nicolas; Christiansen, Jeff; Feng, Guanjie; Hill, Bill; Houghton, Derek; Kaufman, Matthew; Rao, Jianguo; Sharpe, James; Ross, Allyson; Stevenson, Peter; Venkataraman, Shanmugasundaram; Waterhouse, Andrew; Yang, Yiya; and Davidson, Duncan R. **EMAP and EMAGE: a framework for understanding spatially organized data.** *Neuroinformatics*, 1(4):309–325, December 2003. doi: 10.1385/NI:1:4:309.

Balog, Krisztian; de Vries, Arjen P.; Serdyukov, Pavel; Thomas, Paul; and Westerveld, Thijs. **Overview of the TREC 2009 Entity Track**. In *TREC 2009 Working Notes*. NIST, November 2009. URL `http://trec.nist.gov/pubs/trec18/papers/ENT09.OVERVIEW.pdf`.

Banko, Michele and Brill, Eric. **Scaling to Very Very Large Corpora for Natural Language Disambiguation**. In *ACL '01: Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, pages 26–33, Morristown, NJ, USA, 2001. Association for Computational Linguistics. doi: 10.3115/1073012.1073017.

Bechhofer, Sean; Stevens, Robert D.; and Lord, Phillip W. **GOHSE: Ontology driven linking of biology resources.** *Web Semantics: Science, Services and Agents on the World Wide Web*, 4(3):155–163, 2006. doi: 10.1016/j.websem.2005.09.003.

Bechhofer, Sean; Yesilada, Yeliz; Stevens, Robert; Jupp, Simon; and Horan, Bernard. **Using Ontologies and Vocabularies for Dynamic Linking**. *IEEE Internet Computing*, 12(3):32–39, 2008. ISSN 1089-7801. doi: 10.1109/MIC.2008.68.

Belhajjame, Khalid; Embury, Suzanne M.; Paton, Norman W.; Stevens, Robert; and Goble, Carole A. **Automatic annotation of Web services based on workflow definitions**. *ACM Transactions on the Web (TWEB)*, 2(2):1–34, 2008a. ISSN 1559-1131. doi: 10.1145/1346237.1346239.

Belhajjame, Khalid; Goble, Carole; Tanoh, Franck; Bhagat, Jiten; Wolstencroft, Katherine; Stevens, Robert; Nzuobontane, Eric; McWilliam, Hamish; Laurent, Thomas; and Lopez, Rodrigo. **BioCatalogue: A Curated Web Service Registry for the Life Science Community**. In *Microsoft eScience conference*, 2008b.

Berberich, Klaus; Vazirgiannis, Michalis; and Weikum, Gerhard. *Algorithms and Models for the Web-Graph*, volume 3243 of *Lecture Notes in Computer Science*, chapter T-Rank: Time-Aware Authority Ranking, pages 131–142. Springer, Berlin / Heidelberg, 2004. doi: 10.1007/b101552.

Bergmark, Donna; Lagoze, Carl; and Sbityakov, Alex. **Focused Crawls, Tunneling, and Digital Libraries**. In *ECDL '02: Proceedings of the 6th European Conference on Research and Advanced Technology for Digital Libraries*, pages 91–106, London, UK, 2002. Springer-Verlag. ISBN 3-540-44178-6. URL `http://portal.acm.org/citation.cfm?id=646635.700069`.

Berners-Lee, Tim; Hendler, James; and Lassila, Ora. **The Semantic Web**. *Scientific American*, 284(5): 34–43, May 2001.

Bernickus, Bill; Migchielsen, Jos; Pepping, Simon; and Schrauwen, Rob. *Tag by Tag – The Elsevier DTD 5 Family of XML DTDs*. Elsevier, March 2005.

Bieganski, Paul; Riedl, John; Carlis, John V.; and Retzel, Ernest F. **Generalized Suffix Trees for Biological Sequence Data: Applications and Implementation**. In *Biotechnology Computing, Proceedings of the Twenty-Seventh Hawaii International Conference on System Sciences*, volume V, pages 35–44, 1994. doi: 10.1109/HICSS.1994.323593.

Bizer, Christian; Lehmann, Jens; Kobilarov, Georgi; Auer, Sören; Becker, Christian; Cyganiak, Richard; and Hellmann, Sebastian. **DBpedia – A Crystallization Point for the Web of Data**. *Journal of Web Semantics: Science, Services and Agents on the World Wide Web*, 7:154–165, 2009. doi: 10.1016/j.websem.2009.07.002.

Blacoe, Ian W.; Tamma, Valentina; and Wooldridge, Michael J. **Evaluation of scalable multi-agent system architectures for searching the Semantic Web**. *International Journal of Metadata, Semantics and Ontologies*, 5(2):99–119, 2010. ISSN 1744-2621. doi: 10.1504/IJMSO.2010.033281.

Blaschke, Christian; Leon, Eduardo Andres; Krallinger, Martin; and Valencia, Alfonso. **Evaluation of BioCreAtIvE assessment of task 2.** *BMC Bioinformatics*, 6 Suppl 1:S16, 2005. doi: 10.1186/1471-2105-6-S1-S16.

Bodenreider, Olivier. **The Unified Medical Language System (UMLS): integrating biomedical terminology**. *Nucleic Acids Research*, 32(Database-Issue):267–270, 2004. doi: 10.1093/nar/gkh061.

Bollacker, Kurt; Evans, Colin; Paritosh, Praveen; Sturge, Tim; and Taylor, Jamie. **Freebase: a collaboratively created graph database for structuring human knowledge**. In *SIGMOD '08: Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-102-6. doi: 10.1145/1376616.1376746.

Boyer, Robert S. and Moore, J. Strother. **A fast string searching algorithm**. *Communications of the ACM*, 20(10):762–772, 1977. ISSN 0001-0782. doi: 10.1145/359842.359859.

Brickley, Dan; Guha, R.V.; and McBride, Brian. **RDF Vocabulary Description Language 1.0: RDF Schema**. W3C Recommendation, February 2004.
URL `http://www.w3.org/TR/rdf-schema/`.

Brill, Eric; Lin, Jimmy; Banko, Michele; Dumais, Susan; and Ng, Andrew. **Data-Intensive Question Answering**. In *In Proceedings of the Tenth Text REtrieval Conference (TREC*, pages 393–400, 2001.

Brin, Sergey and Page, Lawrence. **The anatomy of a large-scale hypertextual Web search engine**. *Computer Networks and ISDN Systems*, 30(1-7):107–117, 1998. ISSN 0169-7552. doi: 10.1016/S0169-7552(98)00110-X.

de Bruijn, Jos; Bussler, Christoph; Domingue, John; Fensel, Dieter; Hepp, Martin; Keller, Uwe; Kifer, Michael; König-Ries, Birgitta; Kopecky, Jacek; Lara, Rubén; Lausen, Holger; Oren, Eyal; Polleres, Axel; Roman, Dumitru; Scicluna, James; and Stollberg, Michael. **Web Service Modeling Ontology (WSMO)**. W3C Member Submission, June 2005a.
URL `http://www.w3.org/Submission/WSMO/`.

de Bruijn, Jos; Fensel, Dieter; Keller, Uwe; Kifer, Michael; Lausen, Holger; Krummenacher, Reto; Polleres, Axel; and Predoiu, Livia. **Web Service Modeling Language (WSML)**. W3C Member Submission, June 2005b.
URL `http://www.w3.org/Submission/WSML/`.

Caporaso, J. Gregory; Baumgartner, William A.; Randolph, David A.; Cohen, K. Bretonnel; and Hunter, Lawrence. **Rapid pattern development for concept recognition systems: application to point mutations.** *Journal of Bioinformatics and Computational Biology (JBCB)*, 5(6):1233–59, Dec 2007.

Castillo, Carlos. **Effective Web Crawling - Chapter 2**, 2004.

Castillo, Carlos. **Practical Issues of Crawling Large Web Collections**, 2005.

Ceusters, Werner; Smith, Barry; and van Mol, Maarten. **Using ontology in query answering systems: scenarios, requirements and challenges**. In Bernardi, Raffaella and Moortgat, Michael, editors, *Proceedings 2nd CoLogNET ElsNET Symposium – Questions and Answers: Theoretical and Applied Perspectives*, 2003.

Chakrabarti, Soumen; van den Berg, Martin; and Dom, Byron. **Focused crawling: a new approach to topic-specific Web resource discovery**. *Computer Networks*, 31(11–16):1623–1640, 1999. Amsterdam, Netherlands.

Chen, Hao and Sharp, Burt M. **Content-rich biological network constructed by mining PubMed abstracts.** *BMC Bioinformatics*, 5:147, Oct 2004. doi: 10.1186/1471-2105-5-147.

Cheng, Gong; Ge, Weiyi; and Qu, Yuzhong. **Falcons: Searching and Browsing Entities on the Semantic Web**. In *World Wide Web Conference*, Beijing, China, April 2008.

Chi, Lucas and Hui, Kwong. *Combinatorial Pattern Matching*, chapter Color Set Size problem with applications to string matching, pages 230–243. Springer Berlin / Heidelberg, 1992. doi: 10.1007/3-540-56024-6.

Chinnici, Roberto; Moreau, Jean-Jacques; Ryman, Arthur; and Weerawarana, Sanjiva. **Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language**. W3C Recommendation, June 2007.
URL `http://www.w3.org/TR/wsdl20/`.

Cho, Junghoo and Adams, Robert E. **Page Quality: In Search of an Unbiased Web Ranking**. Technical report, UCLA Computer Science Department, November 2003.

Clarke, Charles L.A.; Craswell, Nick; and Soboroff, Ian. **Overview of the TREC 2009 Web Track**. In *TREC 2009 Working Notes*. NIST, 2009.
URL `http://trec.nist.gov/pubs/trec18/papers/WEB09.OVERVIEW.pdf`.

Clauson, Kevin A.; Polen, Hyla H.; Boulos, Maged N. Kamel; and Dzenowagis, Joan H. **Scope, completeness, and accuracy of drug information in Wikipedia.** *The Annals of Pharmacotherapy*, 42(12): 1814–21, Dec 2008. doi: 10.1345/aph.1L474.

ClearForrest. **Calais: Connect. Everything.** WebService, provided by ClearForest, a Thomson Reuters Company, 2008.
URL `http://opencalais.com`.

Clements, Mark. **WikiDB**.
URL `http://www.kennel17.co.uk/testwiki/WikiDB`.

Commentz-Walter, Beate. **A String Matching Algorithm Fast on the Average**. In *Proceedings of the 6th Colloquium, on Automata, Languages and Programming*, pages 118–132, London, UK, 1979. Springer-Verlag. ISBN 3-540-09510-1.

Corbet, Jonathan. **Trees I: Radix trees**. Technical report, LWN.net, 2006.
URL `http://lwn.net/Articles/175432/`.

Crockford, D. **The application/json Media Type for JavaScript Object Notation (JSON)**. RFC 4627, The Internet Engineering Task Force (IETF) – Network Working Group, July 2006.
URL `http://www.ietf.org/rfc/rfc4627.txt`.

Croft, W. Bruce and Lafferty, John, editors. *Language Modeling for Information Retrieval*, volume 13 of *The Information Retrieval Series*. Kluwer Academic Publishers, 2003.

Cunningham, Hamish; Maynard, Dianna; Bontcheva, Kalina; and Tablan, Valentin. **GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications**. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02)*, Philadelphia, July 2002.

Datta, Ritendra; Ge, Weina; Li, Jia; and Wang, James Z. **Toward bridging the annotation-retrieval gap in image search by a generative modeling approach**. In *MULTIMEDIA '06: Proceedings of the 14th annual ACM international conference on Multimedia*, pages 977–986, New York, NY, USA, 2006. ACM. ISBN 1-59593-447-2. doi: 10.1145/1180639.1180856.

Datta, Ritendra; Joshi, Dhiraj; Li, Jia; and Wang, James Z. **Image retrieval: Ideas, influences, and trends of the new age**. *ACM Computing Surveys (CSUR)*, 40(2):1–60, 2008. ISSN 0360-0300. doi: 10.1145/1348246.1348248.

De Roure, David; Goble, Carole; and Stevens, Robert. **The design and realisation of the myExperiment Virtual Research Environment for social sharing of workflows**. *Future Generation Computer Systems*, 25(5):561–567, May 2009. ISSN 0167739X. doi: 10.1016/j.future.2008.06.010.
URL `http://eprints.ecs.soton.ac.uk/15709/`.

Dean, Jeffrey and Ghemawat, Sanjay. **MapReduce: simplified data processing on large clusters**. In *OSDI'04: Proceedings of the 6th conference on Symposium on Opearting Systems Design & Implementation*, pages 10–10, Berkeley, CA, USA, 2004. USENIX Association.

Dean, Jeffrey and Ghemawat, Sanjay. **MapReduce: simplified data processing on large clusters**. *Communications of the ACM*, 51(1):107–113, 2008. ISSN 0001-0782. doi: 10.1145/1327452.1327492.

del Pozo, Angela; Pazos, Florencio; and Valencia, Alfonso. **Defining functional distances over Gene Ontology**. *BMC Bioinformatics*, 9(1):50+, 2008. ISSN 1471-2105. doi: 10.1186/1471-2105-9-50.

DiBernardo, Michael; Pottinger, Rachel; and Wilkinson, Mark. **Semi-automatic web service composition for the life sciences using the BioMoby semantic web framework.** *Journal of Biomedical Informatics*, 41(5):837–47, October 2008. doi: 10.1016/j.jbi.2008.02.005.

Dieng-Kuntz, Rose and Corby, Olivier. **Conceptual graphs for Semantic Web applications**. In *International Conference on Conceptual Structures (ICCS)*, volume 3596 of *LNCS*. Springer, 2005. doi: 10.1007/11524564_2.

Dierks, Tim and Rescorla, Eric. **The Transport Layer Security (TLS) Protocol – Version 1.2**. RFC 5246, The Internet Engineering Task Force (IETF) – Network Working Group, 2008.
URL `http://tools.ietf.org/html/rfc5246`.

Dietze, Heiko and Schroeder, Michael. **GoWeb: A semantic search engine for the life science web**. In Albert Burger, Adrian Paschke, Paolo Romano and Splendiani, Andrea, editors, *In Proceedings of the Intl. Workshop on Semantic Web Applications and Tools for the Life Sciences SWAT4LS*, November 2008.

Dietze, Heiko and Schroeder, Michael. **GoWeb: a semantic search engine for the life science web**. *BMC Bioinformatics*, 10(Suppl 10):S7, 2009. ISSN 1471-2105. doi: 10.1186/1471-2105-10-S10-S7.

Dietze, Heiko; Alexopoulou, Dimitra; Alvers, Michael R.; Barrio-Alvers, Bill; Doms, Andreas; Hakenberg, Jörg; Mönnich, Jan; Plake, Conrad; Reischuk, Andreas; Royer, Loic; Wächter, Thomas; Zschunke, Matthias; and Schroeder, Michael. *Bioinformatics for Systems Biology*, chapter GoPubMed: Exploring Pubmed with Ontological Background Knowledge. Humana Press, 2008a.

Dietze, Heiko; Alexopoulou, Dimitra; Alvers, Michael R.; Barrio-Alvers, Bill; Doms, Andreas; Hakenberg, Jörg; Mönnich, Jan; Plake, Conrad; Reischuk, Andreas; Royer, Loic; Wächter, Thomas; Zschunke, Matthias; and Schroeder, Michael. **GoPubMed: Exploring Pubmed with Ontological Background Knowledge**. In Ashburner, Michael; Leser, Ulf; and Rebholz-Schuhmann, Dietrich, editors, *Ontologies and Text Mining for Life Sciences: Current Status and Future Perspectives*, number 08131 in Dagstuhl Seminar Proceedings, Dagstuhl, Germany, 2008b. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany.
URL `http://drops.dagstuhl.de/opus/volltexte/2008/1520`.

Ding, Chris; He, Xiaofeng; Husbands, Parry; Zha, Hongyuan; and Simon, Horst. **PageRank, HITS and a Unified Framework for Link Analysis**. Technical Report 49372, LBNL, 2002.

Ding, Li; Finin, Tim; Joshi, Anupam; Pan, Rong; Cost, R. Scott; Peng, Yun; Reddivari, Pavan; Doshi, Vishal; and Sachs, Joel. **Swoogle: a search and metadata engine for the semantic web**. In *CIKM '04: Proceedings of the thirteenth ACM international conference on Information and knowledge management*, pages 652–659, New York, NY, USA, 2004. ACM. ISBN 1-58113-874-1. doi: 10.1145/1031171.1031289.

Doms, Andreas. **Using sequence alignment algorithms to extract gene ontology terms in biomedical literature abstracts.** Diploma thesis, TU Dresden, 2004.

Doms, Andreas and Schroeder, Michael. **GoPubMed: exploring PubMed with the Gene Ontology.** *Nucleic Acids Research*, 33(Web Server issue):W783–W786, Jul 2005. doi: 10.1093/nar/gki470.

Dorow, Beate and Widdows, Dominic. **Discovering corpus-specific word senses**. In *EACL '03: Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics*, pages 79–82, Morristown, NJ, USA, 2003. Association for Computational Linguistics. ISBN 1-111-56789-0. doi: 10.3115/1067737.1067753.

Eaton, Alfred D. **HubMed: a web-based biomedical literature search interface.** *Nucleic Acids Research*, 34(Web Server issue):W745–W747, Jul 2006. doi: 10.1093/nar/gkl037.

Ehrig, Marc and Maedche, Alexander. **Ontology-focused crawling of Web documents**. In *SAC '03: Proceedings of the 2003 ACM symposium on Applied computing*, pages 1174–1178, New York, NY, USA, 2003. ACM Press. ISBN 1-58113-624-2. doi: 10.1145/952532.952761.

Ehrler, Frédéric; Geissbühler, Antoine; Jimeno, Antonio; and Ruch, Patrick. **Data-poor categorization and passage retrieval for Gene Ontology annotation in Swiss-Prot.** *BMC Bioinformatics*, 6 Suppl 1 (Suppl 1):S23, May 2005. ISSN 1471-2105. doi: 10.1186/1471-2105-6-S1-S23.

Ely, John W.; Osheroff, Jerome A.; Gorman, Paul N.; Ebell, Mark H.; Chambliss, M. Lee; Pifer, Eric A.; and Stavri, P. Zoe. **A taxonomy of generic clinical questions: classification study.** *British Medical Journal*, 321(7258):429–32, Aug 2000. doi: 10.1136/bmj.321.7258.429.

Fall, Caspar J.; Törcsvári, Attila; Benzineb, Karim; and Karetka, Gabor. **Automated categorization in the international patent classification**. *ACM SIGIR Forum*, 37(1):10–25, 2003. ISSN 0163-5840. doi: 10.1145/945546.945547.

Farkas, Richárd. **The strength of co-authorship in gene name disambiguation**. *BMC Bioinformatics*, 9: 69, 2008. doi: 10.1186/1471-2105-9-69.

Farrell, Joel and Lausen, Holger. **Semantic Annotations for WSDL and XML Schema**. W3C Recommendation, August 2007.
URL `http://www.w3.org/TR/sawsdl/`.

Fellbaum, Christiane, editor. *WordNet: An Electronic Lexical Database (Language, Speech, and Communication)*. The MIT Press, USA, May 1998. ISBN 026206197X.

Fujii, Atsushi. **Enhancing patent retrieval by citation analysis**. In *SIGIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 793–794, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-597-7. doi: 10.1145/1277741.1277912.

Fukuda, Ken-Ichiro; Tsunoda, Tatsuhiko; Tamura, Ayuchi; and Takagi, Toshihisa. **Toward information extraction: identifying protein names from biological papers.** In *Pacific Symposium on Biocomputing*, pages 707–718, Singapore, 1998.

Gale, William A.; Church, Kenneth W.; and Yarowsky, David. **One sense per discourse**. In *HLT '91: Proceedings of the workshop on Speech and Natural Language*, pages 233–237, Morristown, NJ, USA, 1992. Association for Computational Linguistics. ISBN 1-55860-272-0. doi: 10.3115/1075527.1075579.

Gaudan, Sylvain; Kirsch, Harald; and Rebholz-Schuhmann, Dietrich. **Resolving abbreviations to their senses in Medline.** *Bioinformatics*, 21(18):3658–64, Sep 2005. doi: 10.1093/bioinformatics/bti586.

Giantsiou, Lemonia; Loutas, Nikolaos; Peristeras, Vassilios; and Tarabanis, Konstantinos. **Semantic Service Search Engine (S3E): An Approach for Finding Services on the Web**. In *WSKS '09: Proceedings of the 2nd World Summit on the Knowledge Society*, pages 316–325, Berlin, Heidelberg, 2009. Springer-Verlag. ISBN 978-3-642-04753-4. doi: 10.1007/978-3-642-04754-1_33.

Giegerich, Robert and Kurtz, Stefan. **From Ukkonen to McCreight and Weiner: A Unifying View of Linear-Time Suffix Tree Construction**. *Algorithmica*, 19(3):331–353, November 1997. doi: 10.1007/PL00009177.

Ginter, Filip; Boberg, Jorma; Järvinen, Jouni; and Salakoski, Tapio. **New Techniques for Disambiguation in Natural Language and Their Application to Biological Text**. *Journal of Machine Learning Research*, 5:605–621, 2004. ISSN 1532-4435.

Gobeill, Julien; Ehrler, Frédéric; Tbahriti, Imad; and Ruch, Patrick. **Vocabulary-driven Passage Retrieval for Question-Answering in Genomics**. In *The Fifteenth Text REtrieval Conference (TREC 2007) Notebook*, 2007.

Gonzlez, Ivan; Marcus, Adam; Meredith, Daniel N.; and Nguyen, Linda A. **Effective web-scale crawling through website analysis**. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 1041–1042, New York, NY, USA, 2006. ACM Press. ISBN 1-59593-323-9. doi: 10.1145/1135777.1136005.

Good, Benjamin M; Kawas, Edward A; Kuo, Byron Yu-Lin; and Wilkinson, Mark D. **iHOPerator: user-scripting a personalized bioinformatics Web, starting with the iHOP website.** *BMC Bioinformatics*, 7:534, 2006. doi: 10.1186/1471-2105-7-534.

Gottschalk, Karl D; Graham, Stephen; Kreger, Heather; and Snell, James. **Introduction to Web services architecture**. *IBM Systems Journal*, 2:170–177, 2002. doi: 10.1147/sj.412.0170.

Graesser, Arthur C.; McMahen, C. L.; and Johnson, B. K. *Handbook of Psycholinguistics*, chapter Question asking and answering, pages 517–538. Academic Press, San Diego, CA, 1st edition, May 1994.

Granka, Laura A.; Joachims, Thorsten; and Gay, Geri. **Eye-tracking analysis of user behavior in WWW search**. In *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 478–479, New York, NY, USA, 2004. ACM. ISBN 1-58113-881-4. doi: 10.1145/1008992.1009079.

Green, Bert F.; Wolf, Alice K.; Chomsky, Carol; and Laughery, Kenneth. *Computers and Thought*, chapter BASEBALL: An Automatic Question Answerer, pages 207–216. McGraw-Hill, 1963.

Grigoris, Antoniou and van Harmelen, Frank. *A Semantic Web Primer (Cooperative Information Systems)*. The MIT Press, April 2004. ISBN 0262012103.

Gruber, Thomas R. **A Translation Approach to Portable Ontology Specifications**. *Knowledge Acquisition*, 5(2):199–220, April 1993.

Guha, Ramanathan V.; McCool, Rob; and Miller, Eric. **Semantic search**. In *WWW '03: Proceedings of the 12th international conference on World Wide Web*, pages 700–709, New York, NY, USA, 2003. ACM. ISBN 1-58113-680-3. doi: 10.1145/775152.775250.

Gutknecht, Benjamin D. **CCS ONTOLOGY: Domain Knowledge for Semantic Search**. Diploma thesis, Universität Kiel - Institut für Geowissenschaften - Geophysik und Geoinformation, 2010.

Gwehenberger, Gernot. **Anwendung einer binären Verweiskettenmethode beim Aufbau von Listen**. *Elektronische Rechenanlagen*, 10(5):223–226, Oktober 1968.

Haase, Peter; Herzig, Daniel; Musen, Mark; and Tran, Thanh. **Semantic Wiki Search**. In *ESWC 2009 Heraklion: Proceedings of the 6th European Semantic Web Conference on The Semantic Web*, pages 445–460, Berlin, Heidelberg, 2009. Springer-Verlag. ISBN 978-3-642-02120-6. doi: 10.1007/978-3-642-02121-3_34.

Hakenberg, Jörg; Royer, Loïc; Plake, Conrad; Strobelt, Hendrik; and Schroeder, Michael. **Me and my friends: Gene mention normalization with background knowledge**. In *Proceedings 2nd BioCreAtIvE Challenge Evaluation Workshop*, number 2, Madrid, April 2007.

Hakenberg, Jörg; Plake, Conrad; Royer, Loïc; Strobelt, Hendrik; Leser, Ulf; and Schroeder, Michael. **Gene mention normalization and interaction extraction with context models and sentence motifs.** *Genome Biology*, 9 Suppl 2:S14, 2008. doi: 10.1186/gb-2008-9-s2-s14.

Halevy, Alon; Norvig, Peter; and Pereira, Fernando. **The Unreasonable Effectiveness of Data**. *Intelligent Systems, IEEE*, 24(2):8–12, March 2009. doi: 10.1109/MIS.2009.36.

Harmelen, Frank van. **Two Obvious Intuitions: Ontology-Mapping Needs Background Knowledge and Approximation**. In *WI '06: Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence*, page 11, Washington, DC, USA, 2006. IEEE Computer Society. ISBN 0-7695-2747-7. doi: 10.1109/WI.2006.179.

Harth, Andreas; Umbrich, Jürgen; and Decker, Stefan. **MultiCrawler: A Pipelined Architecture for Crawling and Indexing Semantic Web Data**. In Cruz, Isabel F.; Decker, Stefan; Allemang, Dean; Preist, Chris; Schwabe, Daniel; Mika, Peter; Uschold, Michael; and Aroyo, Lora, editors, *International Semantic Web Conference*, Lecture Notes in Computer Science, pages 258–271. Springer, 2006. doi: 10.1007/11926078_19.

Hatzivassiloglou, Vasileios; Duboué, Pablo A.; and Rzhetsky, Andrey. **Disambiguating proteins, genes, and RNA in text: a machine learning approach.** *Bioinformatics*, 17 Suppl 1:S97–106, 2001. doi: 10.1093/bioinformatics/17.suppl_1.S97.

Hawking, David. **Web Search Engines: Part 1**. *Computer*, 39(6):86–88, 2006a. ISSN 0018-9162. doi: 10.1109/MC.2006.213.

Hawking, David. **Web Search Engines: Part 2**. *Computer*, 39(8):88–90, 2006b. ISSN 0018-9162. doi: 10.1109/MC.2006.286.

Hawking, David; Crimmins, Francis; Craswell, Nick; and Upstill, Trystan. **How valuable is external link evidence when searching enterprise Webs?** In *ADC '04: Proceedings of the 15th Australasian database conference*, pages 77–84, Darlinghurst, Australia, Australia, 2004. Australian Computer Society, Inc.

Hersh, William R.; Cohen, Aaron M.; Roberts, Phoebe M.; and Rekapalli, Hari Krishna. **TREC 2006 Genomics Track Overview**. In Voorhees, Ellen M. and Buckland, Lori P., editors, *Proceedings of the Fifteenth Text REtrieval Conference, TREC 2006, Gaithersburg, Maryland, November 14-17, 2006*, volume Special Publication 500-272. National Institute of Standards and Technology (NIST), 2006.

Hirschman, Lynette and Gaizauskas, Rob. **Natural language question answering: the view from here**. *Natural Language Engineering*, 7(4):275–300, 2001. ISSN 1351-3249. doi: 10.1017/S1351324901002807.

Hirschman, Lynette; Morgan, Alexander A; and Yeh, Alexander S. **Rutabaga by any other name: extracting biological names.** *Journal of Biomedical Informatics*, 35(4):247–59, Aug 2002.

Hirschman, Lynette; Yeh, Alexander; Blaschke, Christian; and Valencia, Alfonso. **Overview of BioCreAtIvE: critical assessment of information extraction for biology.** *BMC Bioinformatics*, 6 Suppl 1:S1, 2005.

Hull, Duncan; Wolstencroft, Katy; Stevens, Robert; Goble, Carole; Pocock, Mathew R; Li, Peter; and Oinn, Tom. **Taverna: a tool for building and running workflows of services.** *Nucleic Acids Research*, 34 (Web Server issue):W729–W732, Jul 2006. doi: 10.1093/nar/gkl320.

Humphrey, Susanne M.; Rogers, Willie J.; Kilicoglu, Halil; Demner-Fushman, Dina; and Rindflesch, Thomas C. **Word sense disambiguation by selecting the best semantic type based on Journal Descriptor Indexing: Preliminary experiment**. *Journal of the American Society for Information Science and Technology*, 57(1):96–113, 2006. ISSN 1532-2882. doi: 10.1002/asi.v57:1.

Humphreys, Betsy L.; Lindberg, Donald A.; Schoolman, Harold M.; and Barnett, G. Octo. **The Unified Medical Language System: an informatics research collaboration.** *Journal of the American Medical Informatics Association (JAMIA)*, 5(1):1–11, 1998. doi: 10.1136/jamia.1998.0050001.

Hunter, Amy; Kaufman, Metthew H.; McKay, Angus; Baldock, Richard; Simmen, Martin W.; and Bard, Jonathan B. L. **An ontology of human developmental anatomy**. *Journal of Anatomy*, 203(4):347–355, October 2003. doi: 10.1046/j.1469-7580.2003.00224.x.

Jatowt, Adam; Kawai, Yukiko; and Tanaka, Katsumi. *Web Information Systems Engineering – WISE 2005*, chapter Temporal Ranking of Search Engine Results, pages 43–52. Lecture Notes in Computer Science. Springer, Berlin / Heidelberg, 2005. doi: 10.1007/11581062_4.

Jensen, Lars J.; Saric, Jasmin; and Bork, Peer. **Literature mining for the biologist: from information retrieval to biological discovery**. *Nature Reviews Genetics*, 7(2):119–129, 2006. ISSN 1471-0056.

Kaisser, Michael. **The QuALiM Question Answering Demo: Supplementing Answers with Paragraphs drawn from Wikipedia**. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics, ACL 2008*, Columbus, Ohio, 2008.

Kanehisa, Minoru; Araki, Michihiro; Goto, Susumu; Hattori, Masahiro; Hirakawa, Mika; Itoh, Masumi; Katayama, Toshiaki; Kawashima, Shuichi; Okuda, Shujiro; Tokimatsu, Toshiaki; and Yamanishi, Yoshihiro. **KEGG for linking genomes to life and the environment**. *Nucleic Acids Research*, 36(Database-Issue):D480–D484, 2008. doi: 10.1093/nar/gkm882.

Karp, Richard M. and Rabin, Michael O. **Efficient randomized pattern-matching algorithms**. *IBM Journal of Research and Development*, 31(2):249–260, 1987. ISSN 0018-8646.

Katz, Boris; Borchardt, Gary; and Felshin, Sue. **Natural Language Annotations for Question Answering**. In *Proceedings of the 19th International FLAIRS Conference (FLAIRS 2006)*, Melbourne Beach, FL, May 2006.
URL `http://start.csail.mit.edu/`.

Kim, Jin-Dong.; Ohta, Tomoko; Tsuruoka, Yoshimasa; Tateisi, Yuka; and Collier, Nigel. **Introduction to the bioentity recognition task at JNLPBA**. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications (JNLPBA)*, Geneva, Switzerland, 2004.

Kittler, Ralf; Surendranath, Vineeth; Heninger, Anne-Kristin; Slabicki, Mikolaj; Theis, Mirko; Putz, Gabrielle; Franke, Kristin.; Caldarelli, Antonio; Grabner, Hannes; Kozak, Karol; Wagner, Jan; Rees, Effi; Korn, Bernd; Frenzel, Corina; Sachse, Cristoph; Sonnichsen, Birte; Guo, Jie; Schelter, Janell; Burchard, Julia; Linsley, Peter S.; Jackson, Aimee L.; Habermann, Bianca; and Buchholz, Frank. **Genome-wide resources of endoribonuclease-prepared short interfering RNAs for specific loss-of-function studies**. *Nature Methods*, 4(4):337–344, March 2007. doi: 10.1038/nmeth1025.

Kleinberg, Jon M. **Authoritative sources in a hyperlinked environment**. *Journal of the ACM*, 46(5): 604–632, 1999.

Knuth, Donald E.; Morris Jr., James H.; and Pratt, Vaughan R. **Fast Pattern Matching in Strings**. *SIAM Journal on Computing*, 6(2):323–350, 1977. submitted 1974.

Kourtesis, Dimitrios; Paraskakis, Iraklis; Friesen, Andreas; Gouvas, Panagiotis; and Bouras, Athanasios. **Web Service Discovery In A Semantically Extended Uddi Registry: The Case Of Fusion**. In Camarinha-Matos, Luis M.; Afsarmanesh, Hamideh; Novais, Paulo; and Analide, Cesar, editors, *Virtual Enterprises and Collaborative Networks*, volume 243 of *IFIP*, pages 547–554, 2007. doi: 10.1007/978-0-387-73798-0_59.

Krauthammer, Michael; Rzhetsky, Andrey; Morozov, Pavel; and Friedman, Carol. **Using BLAST for identifying gene and protein names in journal articles.** *Gene*, 259(1-2):245–252, Dec 2000. doi: 10.1016/S0378-1119(00)00431-5.

Kwok, Cody C. T.; Etzioni, Oren; and Weld, Daniel S. **Scaling question answering to the Web**. In *In Proceedings of the Tenth International World Wide Web Conference (WWW10)*, pages 150–161, 2001. doi: 10.1145/502115.502117.

Lacovara, Jane E. **When searching for the evidence, stop using Wikipedia!** *MedSurg Nursing articles*, 17(3):153, Jun 2008.

Lai, Jun; Soh, Ben; and Fei, Chai. *Computational Science and Its Applications – ICCSA 2006*, volume 3983 of *Lecture Notes in Computer Science*, chapter A Web Page Ranking Method by Analyzing Hyperlink Structure and K-Elements, pages 179–186. Springer, Berlin / Heidelberg, 2006. doi: 10.1007/11751632_19.

Lambrix, Patrick and Tan, He. *Anatomy Ontologies for Bioinformatics*, chapter Ontology Alignment and Merging, pages 133–149. Springer London, 2008. doi: 10.1007/978-1-84628-885-2_6. ISSN 1568-2684.

Lamprecht, Anna-Lena; Margaria, Tiziana; and Steffen, Bernhard. **Bio-jETI: a framework for semantics-based service composition.** *BMC Bioinformatics*, 10(Suppl 10):S8, 2009.

Lara, Rubén; Polleres, Axel; Lausen, Holger; Roman, Dumitru; de Bruijn, Jos; and Fensel, Dieter. **A Conceptual Comparison between WSMO and OWL-S**. WSMO Deliverable D4.1v0.1, January 2005. URL `http://www.wsmo.org/2004/d4/d4.1/v0.1/20050106/`.

Larkey, Leah S. **A patent search and classification system**. In *DL '99: Proceedings of the fourth ACM conference on Digital libraries*, pages 179–187, New York, NY, USA, 1999. ACM. ISBN 1-58113-145-3. doi: 10.1145/313238.313304.

Lee, Lawrence C.; Horn, Florence; and Cohen, Fred E. **Automatic Extraction of Protein Point Mutations Using a Graph Bigram Association**. *PLoS Computational Biology*, 3(2):e16+, February 2007. doi: 10.1371/journal.pcbi.0030016.

Lee, Minsuk; Cimino, James; Zhu, Hai Ran; Sable, Carl; Shanker, Vijay; Ely, John; and Yu, Hong. **Beyond information retrieval-medical question answering.** In *AMIA Annual Symposium Proceedings*, pages 469–73, 2006.
URL `http://www.ncbi.nlm.nih.gov/pmc/articles/PMC1839371/`.

Leroy, Gondy and Rindflesch, Thomas C. **Effects of information and machine learning algorithms on word sense disambiguation with small datasets**. *International Journal of Medical Informatics*, 74 (7-8):573–585, 2005. doi: 10.1016/j.ijmedinf.2005.03.013.

Li, Xin and Roth, Dan. **Learning question classifiers: the role of semantic information**. *Natural Language Engineering*, 12(3):229–249, 2006. ISSN 1351-3249. doi: 10.1017/S1351324905003955.

Li, Xin; Chen, Hsinchun; Zhang, Zhu; and Li, Jiexun. **Automatic patent classification using citation network information: an experimental study in nanotechnology**. In *JCDL '07: Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries*, pages 419–427, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-644-8. doi: 10.1145/1255175.1255262.

Li, Yaoyong and Bontcheva, Kalina. **Adapting Support Vector Machines for F-term-based Classification of Patents**. *ACM Transactions on Asian Language Information Processing (TALIP)*, 7(2):1–19, 2008. ISSN 1530-0226. doi: 10.1145/1362782.1362786.

Lin, Dekang. **An Information-Theoretic Definition of Similarity**. In Shavlik, Jude W. and Shavlik, Jude W., editors, *ICML*, pages 296–304. Morgan Kaufmann, 1998. ISBN 1-55860-556-8.

Lin, Jimmy. **The Web as a Resource for Question Answering: Perspectives and Challenges**. In *In Proceedings of the Third International Conference on Language Resources and Evaluation (LREC-2002*, 2002.

Lin, Jimmy. **An exploration of the principles underlying redundancy-based factoid question answering**. *ACM Transactions on Information Systems (TOIS)*, 25(2):6, 2007. ISSN 1046-8188. doi: 10.1145/1229179.1229180.

Lin, Jimmy. **Is searching full text more effective than searching abstracts?** *BMC Bioinformatics*, 10: 46, February 2009. doi: doi:10.1186/1471-2105-10-46.

Liu, Hongfang; Johnson, Stephen B.; and Friedman, Carol. **Automatic resolution of ambiguous terms based on machine learning and conceptual relations in the UMLS**. *Journal of the American Medical Informatics Association*, 9(6):621–636, November 2002. doi: 10.1197/jamia.M1101.

Liu, Hongfang; Teller, Virginia; and Friedman, Carol. **A multi-aspect comparison study of supervised word sense disambiguation.** *Journal of the American Medical Association (JAMA)*, 11(4):320–31, 2004. doi: 10.1197/jamia.M1533.

Liu, Ying; Qin, Tao; Liu, Tie-Yan; Zhang, Lei; and Ma, Wei-Ying. **Similarity space projection for web image search and annotation**. In *MIR '05: Proceedings of the 7th ACM SIGMM international workshop on Multimedia information retrieval*, pages 49–56, New York, NY, USA, 2005. ACM. ISBN 1-59593-244-5. doi: 10.1145/1101826.1101837.

Lord, Phillip W.; Stevens, Robert D.; Brass, Andy; and Goble, Carole A. **Investigating Semantic Similarity Measures Across the Gene Ontology: The Relationship Between Sequence and Annotation**. *Bioinformatics*, 19(10):1275–1283, 2003. doi: 10.1093/bioinformatics/btg153.

Lupu, Mihai; Piroi, Florina; Huang, Xiangji (Jimmy); Zhu, Jianhan; and Tait, John. **Overview of the TREC 2009 Chemical IR Track**. In *TREC 2009 Working Notes*, 2009.
URL `http://trec.nist.gov/pubs/trec18/papers/CHEM09.OVERVIEW.pdf`.

Mangold, Christoph. *Konzepte und Realisierung einer kontextbasierten Intranet-Suchmaschine*. PhD thesis, Universität Stuttgart, Germany, 2007.

Manning, Chris D. and Schütze, Hinrich. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, MA, May 1999.
URL `http://nlp.stanford.edu/fsnlp/`.

Manning, Christopher D.; Raghavan, Prabhakar; and Schütze, Hinrich. *Introduction to Information Retrieval*. Cambridge University Press, 2008.
URL `http://nlp.stanford.edu/IR-book/information-retrieval-book.html`.

Marchiori, Massimo. **The quest for correct information on the Web: hyper search engines**. In *Selected papers from the sixth international conference on World Wide Web*, pages 1225–1235, Essex, UK, 1997. Elsevier Science Publishers Ltd. doi: 10.1016/S0169-7552(97)00036-6. submitted 1996.

Marin, Mauricio; Paredes, Rodrigo; and Bonacic, Carolina. **High-performance priority queues for parallel crawlers**. In *WIDM '08: Proceeding of the 10th ACM workshop on Web information and data management*, pages 47–54, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-260-3. doi: 10.1145/1458502.1458511.

Martin, David; Burstein, Mark; Hobbs, Jerry; Lassila, Ora; McDermott, Drew; McIlraith, Sheila; Narayanan, Srini; Paolucci, Massimo; Parsia, Bijan; Payne, Terry; Sirin, Evren; Srinivasan, Naveen; and Sycara, Katia. **OWL-S: Semantic Markup for Web Services**. W3C Member Submission, November 2004.
URL `http://www.w3.org/Submission/OWL-S`.

Mascord, Damien. **Java Regular expression library benchmarks**, April 2005.
URL `http://tusker.org/regex/regex_benchmark.html`.

McCreight, Edward M. **A Space-Economical Suffix Tree Construction Algorithm**. *Journal of the ACM (JACM)*, 23(2):262–272, 1976. ISSN 0004-5411. doi: 10.1145/321941.321946.

McDonald, David D. *Internal and external evidence in the identification and semantic categorization of proper names*, pages 21–39. MIT Press, Cambridge, MA, USA, 1996. ISBN 0-262-02392-X.

McDonald, Sharon and Tait, John. **Search strategies in content-based image retrieval**. In *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 80–87, New York, NY, USA, 2003. ACM. ISBN 1-58113-646-3. doi: 10.1145/860435.860452.

McLeod, Kenneth and Burger, Albert. **Using argumentation to tackle inconsistency and incompleteness in online distributed life science resources**. In *Proceedings of IADIS International Conference Applied Computing 2007, Salamanca, Spain (18th–20th February 2007)*, pages 489–492. IADIS, 2007.

Menczer, Filippo; Pant, Gautam; and Srinivasan, Padmini. **Topical web crawlers: Evaluating adaptive algorithms**. *ACM Transactions on Internet Technology (TOIT)*, 4(4):378–419, 2004. ISSN 1533-5399. doi: 10.1145/1031114.1031117.

Merelli, Emanuela; Armano, Giuliano; Cannata, Nicola; Corradini, Flavio; d'Inverno, Mark; Doms, Andreas; Lord, Phillip W.; Martin, Andrew; Milanesi, Luciano; Möller, Steffen; Schroeder, Michael; and Luck, Michael. **Agents in bioinformatics, computational and systems biology**. *Briefings in Bioinformatics*, 8(1):45–59, 2007. doi: 10.1093/bib/bbl014.

Mihalcea, Rada. **Co-training and Self-training for Word Sense Disambiguation**. In Ng, Hwee T. and Riloff, Ellen, editors, *HLT-NAACL 2004 Workshop: Eighth Conference on Computational Natural Language Learning (CoNLL-2004)*, pages 33–40, Boston, Massachusetts, USA, May 2004. Association for Computational Linguistics.

Mihalcea, Rada and Csomai, Andras. **Wikify!: linking documents to encyclopedic knowledge**. In *CIKM '07: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 233–242, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-803-9. doi: 10.1145/1321440.1321475.

Mollá, Diego and Vicedo, José Luis. **Question Answering in Restricted Domains: An Overview**. *Computational Linguistics*, 33(1):41–61, 2007. ISSN 0891-2017. doi: 10.1162/coli.2007.33.1.41.

Møller, Anders. **dk.brics.automaton, version 1.11-2**, August 2009.
URL `http://www.brics.dk/automaton/index.html`. Department of Computer Science, Aarhus University.

Moreira, José E.; Michael, Maged M.; Da Silva, Dilma; Shiloach, Doron; Dube, Parijat; and Zhang, Li. **Scalability of the Nutch search engine**. In *ICS '07: Proceedings of the 21st annual international conference on Supercomputing*, pages 3–12, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-768-1. doi: 10.1145/1274971.1274975.

Morrison, Donald R. **PATRICIA—Practical Algorithm To Retrieve Information Coded in Alphanumeric**. *Journal of the ACM (JACM)*, 15(4):514–534, 1968. ISSN 0004-5411. doi: 10.1145/321479.321481.

Mukhopadhyay, Debajyoti and Biswas, Pradipta, editors. *Distributed Computing and Internet Technology*, volume 3816, chapter FlexiRank: An Algorithm Offering Flexibility and Accuracy for Ranking the Web Pages , pages 308–313. Springer, Berlin / Heidelberg, 2005. doi: 10.1007/11604655_35.

Müller, Hans-Michael; Kenny, Eimear E; and Sternberg, Paul W. **Textpresso: an ontology-based information retrieval and extraction system for biological literature.** *PLoS Biology*, 2(11):e309, Nov 2004. doi: 10.1371/journal.pbio.0020309.

Needleman, Saul B. and Wunsch, Chistian D. **A general method applicable to the search for similarities in the amino acid sequence of two proteins.** *Journal of Molecular Biology*, 48(3):443–53, Mar 1970. doi: 10.1016/0022-2836(70)90057-4.

Nelson, Stuart J.; Johnston, Douglas; and Humphreys, Betsy. *Relationships in the organization of knowledge*, chapter Relationships in Medical Subject Headings, pages 171–184. Kluwer Academic Publishers, New York, 2001.
URL `http://www.nlm.nih.gov/mesh/meshrels.html`.

Neubauer, Falk; Hoheisel, Andreas; and Geiler, Joachim. **Workflow-based grid applications**. *Future Generation Computer Systems*, 22(1):6–15, 2006. ISSN 0167-739X. doi: 10.1016/j.future.2005.08.002.

Ogren, Philip V.; Cohen, K. Bretonnel; Acquaah-Mensah, George; Eberlein, Jens; and Hunter, Lawrence. **The compositional structure of Gene Ontology terms.** In Altman, Russ B.; Dunker, A. Keith; Hunter, Lawrence; Jung, Tiffany A.; and Klein, Teri E., editors, *Proceedings of the Pacific Symposium on Biocomputing, Hawaii, USA*, pages 214–225, 2004.

Oliver, Helen; Diallo, Gayo; de Quincey, Ed; Alexopoulou, Dimitra; Habermann, Bianca; Kostkova, Patty; Schroeder, Michael; Jupp, Simon; Khelif, Khaled; Stevens, Robert; Jawaheer, Gawesh; and Madle, Gemma. **A user-centred evaluation framework for the Sealife semantic web browsers**. *BMC Bioinformatics*, 10 Suppl 10(S14), 2009. doi: 10.1186/1471-2105-10-S10-S14.

Page, Lawrence; Brin, Sergey; Motwani, Rajeev; and Winograd, Terry. **The PageRank Citation Ranking: Bringing Order to the Web**. Technical Report 1997-0072, Stanford InfoLab, 1997.

Pahikkala, Tapio; Ginter, Filip; Boberg, Jorma; Järvinen, Jouni; and Salakoski, Tapio. **Contextual weighting for Support Vector Machines in literature mining: an application to gene versus protein name disambiguation.** *BMC Bioinformatics*, 6:157, 2005. doi: doi:10.1186/1471-2105-6-157.

Pant, Gautam and Srinivasan, Padmini. **Learning to crawl: Comparing classification schemes**. *ACM Transactions on Information Systems (TOIS)*, 23(4):430–462, 2005. ISSN 1046-8188. doi: 10.1145/1095872.1095875.

Paşca, Marius. **Lightweight web-based fact repositories for textual question answering**. In *CIKM '07: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 87–96, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-803-9. doi: 10.1145/1321440.1321455.

Paschke, Adrian; Boley, Harold; Kozlenkov, Alexander; and Craig, Benjamin. **Rule responder: RuleML-based agents for distributed collaboration on the pragmatic web**. In *ICPW '07: Proceedings of the 2nd international conference on Pragmatic web*, pages 17–28, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-859-6. doi: 10.1145/1324237.1324240.

Pautasso, Cesare; Zimmermann, Olaf; and Leymann, Frank. **RESTful Web Services vs. "Big" Web Services: Making the Right Architectural Decision**. In *WWW '08: Proceeding of the 17th international conference on World Wide Web*, pages 805–814, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-085-2. doi: 10.1145/1367497.1367606.

Pedersen, Ted and Bruce, Rebecca. **Distinguishing Word Senses in Untagged Text**. In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, pages 197–207, 1997.

Pedersen, Ted and Bruce, Rebecca. **Knowledge lean word-sense disambiguation**. In *AAAI '98/IAAI '98: Proceedings of the fifteenth national/tenth conference on Artificial intelligence/Innovative applications of artificial intelligence*, pages 800–805, Menlo Park, CA, USA, 1998. American Association for Artificial Intelligence. ISBN 0-262-51098-7.

Perez-Iratxeta, Carolina; Andrade-Navarro, Miguel A; and Wren, Jonathan D. **Evolving research trends in bioinformatics.** *Briefings in Bioinformatics*, 8(2):88–95, Mar 2007. doi: 10.1093/bib/bbl035.

Pesenhofer, Andreas; Edler, Sonja; Berger, Helmut; and Dittenbach, Michael. **Towards a patent taxonomy integration and interaction framework**. In *PaIR '08: Proceeding of the 1st ACM workshop on Patent information retrieval*, pages 19–24, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-256-6. doi: 10.1145/1458572.1458578.

Plake, Conrad; Doms, Andreas; Dietze, Heiko; Wächter, Thomas; Alvers, M. R.; and Schroeder, Michael. **Ontology-based Assisted Curation**. In *3rd International Biocuration Conference*, page 16, Berlin, Germany, April 16-19 2009.

Pomerantz, Jeffrey. **A linguistic analysis of question taxonomies: Research Articles**. *Journal of the American Society for Information Science and Technology*, 56(7):715–728, 2005. ISSN 1532-2882. doi: 10.1002/asi.20162.

Prud'hommeaux, Eric and Seaborne, Andy. **SPARQL Query Language for RDF**. W3C Recommendation, January 2008.
URL `http://www.w3.org/TR/rdf-sparql-query/`.

Purandare, Amruta and Pedersen, Ted. **Word Sense Discrimination by Clustering Contexts in Vector and Similarity Spaces**. In *HLT-NAACL 2004 Workshop: Eighth Conference on Computational Natural Language Learning (CoNLL-2004)*, pages 41–48, 2004.

Rada, Roy; Mili, Hafedh; Bicknell, Ellen; and Blettner, Maria. **Development and application of a metric on semantic nets**. *IEEE Transactions on Systems, Man and Cybernetics*, 19(1):17–30, 1989. doi: 10.1109/21.24528.

Ranger, Colby; Raghuraman, Ramanan; Penmetsa, Arun; Bradski, Gary; and Kozyrakis, Christos. **Evaluating MapReduce for Multi-core and Multiprocessor Systems**. In *HPCA '07: Proceedings of the 2007 IEEE 13th International Symposium on High Performance Computer Architecture*, pages 13–24, Washington, DC, USA, 2007. IEEE Computer Society. ISBN 1-4244-0804-0. doi: 10.1109/HPCA.2007.346181.

Rebholz-Schuhmann, Dietrich; Kirsch, Harald; Arregui, Miguel; Gaudan, Sylvain; Riethoven, Mark; and Stoehr, Peter. **EBIMed–text crunching to gather facts for proteins from Medline.** *Bioinformatics*, 23 (2):e237–e244, Jan 2007. doi: 10.1093/bioinformatics/btl302.

Rector, Alan; Rogers, Jeremy; Zanstra, Pieter; and Van Der Haring, Egbert. **OpenGALEN: open source medical terminology and tools.** In *AMIA Annual Symposium Proceedings*, page 982, 2003.
URL `http://www.ncbi.nlm.nih.gov/pmc/articles/PMC1480228/`.

Rescorla, Eric and Schiffman, Allan M. **The Secure HyperText Transfer Protocol**. RFC 2660, The Internet Engineering Task Force (IETF) – Network Working Group, 1999.
URL `http://tools.ietf.org/html/rfc2660`.

Resnick, Peter W. **Internet Message Format**. RFC 5322, The Internet Engineering Task Force (IETF) – Network Working Group, October 2008.
URL `http://tools.ietf.org/html/rfc5322`.

Resnik, Philip. **Using information content to evaluate semantic similarity in a taxonomy**. In *IJCAI'95: Proceedings of the 14th international joint conference on Artificial intelligence*, pages 448–453, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc. ISBN 1-55860-363-8, 978-1-558-60363-9.

Richardson, Matthew; Prakash, Amit; and Brill, Eric. **Beyond PageRank: machine learning for static ranking**. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 707–715, New York, NY, USA, 2006. ACM Press. ISBN 1-59593-323-9. doi: 10.1145/1135777.1135881.

Robertson, Stephen E.; Walker, Steve; Jones, Susan; Hancock-Beaulieu, Micheline; and Gatford, Mike. **Okapi at TREC-3**. In *Overview of the 3rd TREC Text REtrieval Conference*, pages 109–126, 1996.

Roussinov, Dmitri; Fan, Weiguo; and Robles-Flores, José. **Beyond keywords: Automated question answering on the web**. *Communications of the ACM*, 51(9):60–65, 2008. ISSN 0001-0782. doi: 10.1145/1378727.1378743.

Şah, Melike; Hall, Wendy; and De Roure, David C. **Dynamic linking and personalization on web**. In *SAC '10: Proceedings of the 2010 ACM Symposium on Applied Computing*, pages 1404–1410, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-639-7. doi: 10.1145/1774088.1774386.

Salton, Gerard; Wong, Anita; and Yang, Chung-Shu. **A vector space model for automatic indexing**. *Communications of the ACM*, 18(11):613–620, 1975. ISSN 0001-0782. doi: 10.1145/361219.361220.

Salz, Jürgen. **Der Pillen-Knick**. *WirtschaftsWoche*, 12:66–67, March 2009.

Scharffe, François; Euzenat, Jérôme; and Fensel, Dieter. **Towards design patterns for ontology alignment**. In *SAC '08: Proceedings of the 2008 ACM symposium on Applied computing*, pages 2321–2325, New York, NY, USA, 2008. ACM. ISBN 978-1-59593-753-7. doi: 10.1145/1363686.1364236.

Schijvenaars, Bob J A; Mons, Barend; Weeber, Marc; Schuemie, Martijn J; Mulligen, Erik M van Mulligen; Wain, Hester M; and Kors, Jan A. **Thesaurus-based disambiguation of gene symbols.** *BMC Bioinformatics*, 6:149, 2005. doi: 10.1186/1471-2105-6-149.

Schlicker, Andreas; Domingues, Francisco S.; Rahnenführer, Jörg; and Lengauer, Thomas. **A new measure for functional similarity of gene products based on Gene Ontology.** *BMC Bioinformatics*, 7:302, 2006. doi: 10.1186/1471-2105-7-302.
URL `http://www.biomedcentral.com/1471-2105/7/302`.

Schroeder, Michael; Burger, Albert; Kostkova, Patty; Stevens, Robert; Habermann, Bianca; and Dieng-Kuntz, Rose. **Sealife: a semantic grid browser for the life sciences applied to the study of infectious diseases.** *Studies in Health Technology and Informatics*, 120:167–178, 2006.
URL `http://www.ncbi.nlm.nih.gov/pubmed/16823135`.

Schuemie, Martijn J; Kors, Jan A; and Mons, Barend. **Word sense disambiguation in the biomedical domain: an overview.** *Journal of Computational Biology*, 12(5):554–65, Jun 2005. doi: 10.1089/cmb.2005.12.554.

Schulz, Stefan; Kumar, Anand; and Bittner, Thomas. **Biomedical ontologies: what part-of is and isn't.** *Journal of Biomedical Informatics*, 39(3):350–61, Jun 2006. doi: 10.1016/j.jbi.2005.11.003.

Schütze, Hinrich. **Automatic word sense discrimination**. *Computational Linguistics*, 24(1):97–123, 1998. ISSN 0891-2017.

Schütze, Hinrich and Pedersen, Jan O. **Information retrieval based on word senses**. In *Proceedings of the 4th Annual Symposium on Document Analysis and Information Retrieval*, pages 161–175, 1995.

Shah, Rajiv; Kesan, Jay; and Kennis, Andrew. **Implementing open standards: a case study of the Massachusetts open formats policy**. In *dg.o '08: Proceedings of the 2008 international conference on Digital government research*, pages 262–271. Digital Government Society of North America, 2008. ISBN 978-1-60558-099-9.

Shao, Jie; Shen, Heng Tao; and Zhou, Xiaofang. **Challenges and techniques for effective and efficient similarity search in large video databases**. *Proceedings of the VLDB Endowment*, 1(2):1598–1603, 2008. doi: 10.1145/1454159.1454232.

Signorini, Alessio and Imielinski, Tomasz. **If You Ask Nicely, I will Answer: Semantic Search and Today's Search Engines**. In *ICSC '09: Proceedings of the 2009 IEEE International Conference on Semantic Computing*, pages 184–191, Washington, DC, USA, 2009. IEEE Computer Society. ISBN 978-0-7695-3800-6. doi: 10.1109/ICSC.2009.31.

Simmons, Robert F. **Answering English questions by computer: a survey**. *Communications of the ACM*, 8(1):53–70, 1965. ISSN 0001-0782. doi: 10.1145/363707.363732.

Simmons, Robert F. **Natural language question-answering systems: 1969**. *Communications of the ACM*, 13(1):15–30, 1970. ISSN 0001-0782. doi: 10.1145/361953.361963.

Singhal, Amit; Buckley, Chris; and Mitra, Mandar. **Pivoted document length normalization**. In *SIGIR '96: Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 21–29, New York, NY, USA, 1996. ACM. ISBN 0-89791-792-8. doi: 10.1145/243199.243206.

Smith, Barry; Ashburner, Michael; Rosse, Cornelius; Bard, Jonathan; Bug, William; Ceusters, Werner; Goldberg, Louis J.; Eilbeck, Karen; Ireland, Amelia; Mungall, Christopher J.; The OBI Consortium; Leontis, Neocles; Rocca-Serra, Philippe; Ruttenberg, Alan; Sansone, Susanna-Assunta; Scheuermann, Richard H.; Shah, Nigam; Whetzel, Patricia L.; and Lewis, Suzanna. **The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration.** *Nature Biotechnology*, 25(11):1251–1255, Nov 2007. doi: 10.1038/nbt1346.

Smith, Temple F. and Waterman, Michael S. **Identification of common molecular subsequences.** *Journal of Molecular Biology*, 147(1):195–197, Mar 1981. doi: 10.1016/0022-2836(81)90087-5.

Spasic, Irena; Ananiadou, Sophia; McNaught, John; and Kumar, Anand. **Text mining and ontologies in biomedicine: making sense of raw text.** *Briefings in Bioinformatics*, 6(3):239–251, Sep 2005. doi: 10.1093/bib/6.3.239.

Srihari, Rohini and Li, Wei. **Information Extraction Supported Question Answering**. In *In Proceedings of the Eighth Text REtrieval Conference (TREC-8)*, pages 185–196, 1999.

St. Laurent, Simon; Johnston, Joe; and Dumbill, Edd. *Programming Web Services with XML-RPC*. O'Reilly, June 2001.

Stillman-Lowe, C. **Wikipedia comes second.** *British Dental Journal*, 205(10):525, Nov 2008. doi: 10.1038/sj.bdj.2008.994.

Stock, Mechtild and Stock, Wolfgang G. **Intellectual property information: A comparative analysis of main information providers**. *Journal of the American Society for Information Science and Technology*, 57(13):1794–1803, 2006. ISSN 1532-2882. doi: 10.1002/asi.v57:13.

Studer, Rudi and Sure, York. **Cost Estimation in Ontology Engineering**. In *IST 2006 Conference & Exhibition – Building Semantic Knowledge Applications*, 2006.
URL `http://cordis.europa.eu/ist/kct/event_ist_20061122_sekt.htm`.

Sussna, Michael. **Word sense disambiguation for free-text indexing using a massive semantic network**. In *CIKM '93: Proceedings of the second international conference on Information and knowledge management*, pages 67–74, New York, NY, USA, 1993. ACM. ISBN 0897916263. doi: 10.1145/170088.170106.

Sutherland, Karen; McLeod, Kenneth; and Burger, Albert. **Semantically linking web pages to web services in Bioinformatics**. In *3rd International Applications of Semantic Technologies Workshop*, September 2008.

Sutherland, Karen; McLeod, Kenneth; Ferguson, Gus; and Burger, Albert. **Knowledge-driven enhancements for task composition in bioinformatics**. *BMC Bioinformatics*, 10(Suppl 10)(S12), 2009. doi: 10.1186/1471-2105-10-S10-S12.

Tang, Hangwi and Ng, Jennifer Hwee Kwoon. **Googling for a diagnosis–use of Google as a diagnostic aid: internet based study.** *British Medical Journal*, 333(7579):1143–1145, Dec 2006. doi: 10.1136/bmj.39003.640567.AE.

Taubert, Mark. **Use of Google as a diagnostic aid: bias your search.** *British Medical Journal*, 333(7581): 1270; author reply 1270, Dec 2006. doi: 10.1136/bmj.39058.703194.3A.

Thanh Le, Bach and Dieng-Kuntz, Rose. **A Graph-Based Algorithm for Alignment of OWL Ontologies**. In *WI '07: Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence*, pages 466–469, Washington, DC, USA, 2007. IEEE Computer Society. ISBN 0-7695-3026-5. doi: 10.1109/WI.2007.10.

Theiler, Karl. *The House Mouse: Atlas of Mouse Development*. Springer, New York, 1989.

Thut, Catherine J.; Rountree, Ryan B.; Hwa, Michael; and Kingsley, David M. **A large-scale in situ screen provides molecular evidence for the induction of eye anterior segment structures by the developing lens.** *Developmental Biology*, 231(1):63–76, March 2001. ISSN 0012-1606. doi: 10.1006/dbio.2000.0140.

Tomuro, Noriko and Lytinen, Steven L. **Selecting features for paraphrasing question sentences**. In *Proceedings of the Workshop on Automatic Paraphrasing at Natural Language Processing Pacific Rim Symposium (NLPRS)*, pages 55–62, 2001.

Trißl, Silke and Leser, Ulf. **Querying Ontologies in Relational Database Systems.** In Ludäscher, Bertram and Raschid, Louiqa, editors, *Data Integration in the Life Sciences, Second InternationalWorkshop, DILS 2005*, volume 3615 of *Lecture Notes in Computer Science*, pages 63–79, San Diego, CA, USA, July 2005. Springer. doi: 10.1007/11530084_7.

Trißl, Silke and Leser, Ulf. **GRIPP - Indexing and Querying Graphs based on Pre- and Postorder Numbering**. Technical Report 207, Department for Computer Science, Humboldt-Universität Berlin, 2006.

Tsaparas, Panayiotis. **Using Non-Linear Dynamical Systems for Web Searching and Ranking**. In *Principles of Database Systems (PODS)*, pages 59–70, New York, NY, USA, 2004. ACM. doi: 10.1145/1055558.1055569.

Tseng, Yuen-Hsien and Wu, Yi-Jen. **A study of search tactics for patentability search: a case study on patent engineers**. In *PaIR '08: Proceeding of the 1st ACM workshop on Patent information retrieval*, pages 33–36, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-256-6. doi: 10.1145/1458572.1458581.

Tsuruoka, Yoshimasa and Tsujii, Jun'ichi. **Probabilistic term variant generator for biomedical terms**. In *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 167–173, New York, NY, USA, 2003. ACM. ISBN 1-58113-646-3. doi: 10.1145/860435.860467.

Tumer, Duygu; Shah, Mohammad Ahmed; and Bitirim, Yiltan. **An Empirical Evaluation on Semantic Search Performance of Keyword-Based and Semantic Search Engines: Google, Yahoo, Msn and Hakia**. In *ICIMP '09: Proceedings of the 2009 Fourth International Conference on Internet Monitoring and Protection*, pages 51–55, Washington, DC, USA, 2009. IEEE Computer Society. ISBN 978-0-7695-3612-5. doi: 10.1109/ICIMP.2009.16.

Tummarello, Giovanni; Delbru, Renaud; Oren, Eyal; and Cyganiak, Richard. **Sindice.com: A Semantic Web Search Engine**. Presentation, Digital Enterprise Research Institute National University of Ireland, Galway, November 2007.

Tvarozek, Michal and Bielikova, Mária. **Reinventing the Web Browser for the Semantic Web**. In *WI-IAT '09: Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology*, pages 113–116, Washington, DC, USA, 2009. IEEE Computer Society. ISBN 978-0-7695-3801-3. doi: 10.1109/WI-IAT.2009.243.

Twisselmann, Birte. **Use of Google as a diagnostic aid: summary of other responses.** *British Medical Journal*, 333(7581):1270–1271, Dec 2006. doi: 10.1136/bmj.39059.575556.FA.

Ukkonen, Esko. **On-line construction of suffix trees**. *Algorithmica*, 14(3):249–260, 1995. doi: 10.1007/BF01206331.

Völkel, Max; Krötzsch, Markus; Vrandecic, Denny; Haller, Heiko; and Studer, Rudi. **Semantic Wikipedia**. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 585–594, New York, NY, USA, 2006. ACM. ISBN 1-59593-323-9. doi: 10.1145/1135777.1135863.

Wächter, Thomas. **GoPubMed Ontology Generation Plugin for OBO-Edit 2**. Available as part of OBO-Edit 2, 2009.
URL `http://oboedit.org/`.

Wächter, Thomas; Wobst, André; Schroeder, Michael; Tan, He; and Lambrix, Patrick. **A corpus-driven approach for design, evolution and alignment of ontologies**. In *WSC '06: Proceedings of the 38th conference on Winter simulation*, pages 1595–1602. Winter Simulation Conference, 2006. ISBN 1-4244-0501-7.

Wächter, Thomas; Alexopoulou, Dimitra; Dietze, Heiko; Hakenberg, Jörg; and Schroeder, Michael. *Semantic Web: Revolutionizing Knowledge Discovery in the Life Sciences*, chapter Searching Biomedical Literature with Anatomy Ontologies, pages 177–196. Springer, Berlin, 2007.

Waldvogel, Marcel; Varghese, George; Turner, Jon; and Plattner, Bernhard. **Scalable high-speed prefix matching**. *ACM Transactions on Computer Systems (TOCS)*, 19(4):440–482, 2001. ISSN 0734-2071. doi: 10.1145/502912.502914.

Wang, Minhua. **A Significant Improvement to Clever Algorithm in Hyperlinked Environment**, 2002.

Weber, Nils; Braubach, Lars; Pokahr, Alexander; and Lamersdorf, Winfried. **Agent-based semantic search at motoso.de**. In *MATES'09: Proceedings of the 7th German conference on Multiagent system technologies*, pages 278–287, Berlin, Heidelberg, 2009. Springer-Verlag. ISBN 3-642-04142-6, 978-3-642-04142-6.

Weiner, Peter. **Linear pattern matching algorithm**. In *14th Annual IEEE Symposium on Switching and Automata Theory*, pages 1–11, 1973.

Wentz, Reinhard. **Use of Google as a diagnostic aid: is Google like 10,000 monkeys?** *British Medical Journal*, 333(7581):1270; author reply 1270, Dec 2006. doi: 10.1136/bmj.39058.700266.3A.

Widdows, Dominic; Peters, Stanley; Cederberg, Scott; Chan, Chiu-Ki; Steffen, Diana; and Buitelaar, Paul. **Unsupervised monolingual and bilingual word-sense disambiguation of medical documents using UMLS**. In *Proceedings of the ACL 2003 workshop on Natural language processing in biomedicine*, pages 9–16, Morristown, NJ, USA, 2003. Association for Computational Linguistics. doi: 10.3115/1118958.1118960.

Wilensky, Robert; Chin, David N.; Luria, Marc; Martin, James H.; Mayfield, James; and Wu, Dekai. **The Berkeley UNIX Consultant Project**. Technical Report UCB/CSD-89-520, EECS Department, University of California, Berkeley, 1989.
URL `http://www.eecs.berkeley.edu/Pubs/TechRpts/1989/5896.html`.

Wilkinson, Mark D and Links, Matthew. **BioMOBY: an open source biological web services proposal.** *Briefings in Bioinformatics*, 3(4):331–341, Dec 2002. doi: 10.1093/bib/3.4.331.

Winnenburg, Rainer; Plake, Conrad; and Schroeder, Michael. **Improved mutation tagging with gene identifiers applied to membrane protein stability prediction.** *BMC Bioinformatics*, 10 Suppl 8:S3, 2009. doi: 10.1186/1471-2105-10-S8-S3.

Wolstencroft, Katy; Alper, Pinar; Hull, Duncan; Wroe, Chris; Lord, Phillip; Stevens, Robert; and Goble, Carole. **The myGrid ontology: bioinformatics service discovery**. *International Journal of Bioinformatics Research and Applications (IJBRA)*, 3(3):303–325, 2007. ISSN 1744-5485. doi: 10.1504/IJBRA.2007.015005.

Wren, Jonathan D. **URL decay in MEDLINE–a 4-year follow-up study.** *Bioinformatics*, 24(11):1381–5, Jun 2008.

Yahoo! Inc. **Search BOSS service**.
URL `http://developer.yahoo.com/search/boss/`.

Yang, Tao. **Large Scale Internet Search at Ask.com**. In *First International Conference on Scalable Information Systems, INFOSCALE*, 2006. Keynote.

Yarowsky, David. **One sense per collocation**. In *HLT '93: Proceedings of the workshop on Human Language Technology*, pages 266–271, Morristown, NJ, USA, 1993. Association for Computational Linguistics. ISBN 1-55860-324-7. doi: 10.3115/1075671.1075731.

Yarowsky, David. **Unsupervised word sense disambiguation rivaling supervised methods**. In *Proceedings of the 33rd annual meeting on Association for Computational Linguistics*, pages 189–196, Morristown, NJ, USA, 1995. Association for Computational Linguistics. doi: 10.3115/981658.981684.

Yesilada, Yeliz; Bechhofer, Sean; and Horan, Bernard. **Personalised Dynamic Links on theWeb**. In *SMAP '06: Proceedings of the First International Workshop on Semantic Media Adaptation and Personalization*, pages 7–12, Washington, DC, USA, 2006. IEEE Computer Society. ISBN 0-7695-2692-6. doi: 10.1109/SMAP.2006.26.

Yu, Hong and Sable, Carl. **Being Erlang Shen: Identifying answerable questions**. In *Nineteenth International Joint Conference on Artificial Intelligence on Knowledge and Reasoning for Answering Questions*, 2005.

Zajac, Rémi. **Towards ontological question answering**. In *Proceedings of the workshop on Open-domain question answering*, pages 1–7, Morristown, NJ, USA, 2001. Association for Computational Linguistics. doi: 10.3115/1117856.1117861.

Zheng, Zhiping. **AnswerBus question answering system**. In *Proceedings of the second international conference on Human Language Technology Research*, pages 399–404, San Francisco, CA, USA, 2002. Morgan Kaufmann Publishers Inc.
URL `http://www.answerbus.com/`.

Zhou, Wei; Yu, Clement; Torvik, Vetle; and Smalheiser, Neil R. **A concept-based framework for passage retrieval in Genomics**. In *The Fifteenth Text REtrieval Conference Proceedings (TREC'06)*, Baltimore, WA, 2006.

Zobel, Justin and Moffat, Alistair. **Inverted files for text search engines**. *ACM Computing Surveys (CSUR)*, 38(2):6, 2006. ISSN 0360-0300. doi: 10.1145/1132956.1132959.

Zobel, Justin; Moffat, Alistair; and Sacks-Davis, Ron. **Searching Large Lexicons for Partially Specified Terms using Compressed Inverted Files**. In *VLDB '93: Proceedings of the 19th International Conference on Very Large Data Bases*, pages 290–301, San Francisco, CA, USA, 1993. Morgan Kaufmann Publishers Inc. ISBN 1-55860-152-X.

Zobel, Justin; Moffat, Alistair; and Ramamohanarao, Kotagiri. **Inverted files versus signature files for text indexing**. *ACM Transactions on Database Systems (TODS)*, 23(4):453–490, 1998. ISSN 0362-5915. doi: 10.1145/296854.277632.

APPENDIX A

EVIDENCE DATA FOR GOWEB

A.1 Evidences for GoWeb Results with Google questions

Case 5

Tree: Diseases
→ Cardiovascular Diseases (534)
→ Cardiovascular Infections (9)
→ Endocarditis, Bacterial (7)

- **Pathpots**
  sub-acute bacterial endocarditis, which occurs on . . . stellate scarring of intima, syphilis, angina, aortic regurgitation, hypertrophy and dilatation . . .
  `www.fortunecity.com/bennyhills/mayall/3/pathpots.htm`
- **part ii**
  . . . RIGHT-SIDED FAILURE Pulmonary emboli (acute or chronic) Any disease interfering . . . AORTIC REGURGITATION Old rheumatic fever Bacterial endocarditis Syphilis . . .
  `www.pathguy.com/boildown.txt`
- **Case 29-2004 A 75-Year-Old Woman with Acute Onset of Chest Pain Followed by Fever**
  There was moderate aortic regurgitation as visualized on Doppler ultrasonography. . . . Acute bacterial endocarditis with aortic-root abscess, due to . . .

Case 6

Tree: Diseases (220)
→ Neoplasm (60)
→ Neoplasms by Histologic Type (10)
→ Neoplasms, Glandular and Epithelial (5)
→ Carcinoma (5)
→ Adenocarcinoma (2) → Parents for "Linitis Plastica"

- **U.S. Pharmacist**
  . . . can lead to adenocarcinoma, a cancer that is increasing rapidly in white . . . Nausea or vomiting. Stomach pain. The Pregnant Patient With Heartburn . . .
  `www.uspharmacist.com/index.asp?page=ce/2949/default.htm`

- **Summary of Product Characteristics**
  For small cell lung cancer in multiple-agent chemotherapy, ... manifested as loss of appetite, nausea, sometimes vomiting as well as diarrhoea, and ...
  `www.medac.de/medac_international/data/doc/SPC_Cisplatin.pdf`

Diseases (220)
→ Digestive System Diseases (60)
→ Gastrointestinal Diseases (54)
→ Intestinal Diseases (12)
→ Intestinal Obstruction (6)
→ Ileus (2)
→ Fecal Impaction (1)
→ Afferent Loop Syndrome (1)

- **THE PALLIATIVE CARE HANDBOOK**
  There are many causes of nausea and vomiting and more than one cause may be ... Intestinal obstruction in association with advanced cancer is often complex ...
  `www.aswcs.nhs.uk/sitespecgps/palcare/Palliative%20Care%20 handbook%202007.pdf`

- **Ileus - Ileus symptom, treatment, causes**
  They may report nausea, vomiting, and poor appetite. Abdominal cramping is usually not present. ... Ovarian Cancer. Ovarian Cyst. Parkinson's Disease ...
  `www.healthcentral.com/druglibrary/408/zyprexa-side_effects_drug_ interactions ...`

- **HONG KONG COLLEGE OF RADIOLOGISTS**
  ... with advanced progressive cancer have a right to ... Nausea and Vomiting. Dysphagia. Constipation and Fecal Impaction. Diarrhea. Intestinal obstruction ...
  `www.hkcr.org/edu/trainingregulations/Training%20guidelines%20of%20 Palliative ...`

- **PROBLEMS AS STARTING POINTS FOR TRAINING**
  screening programme on breast cancer. abdominal pain during pregnancy ... oesophageal varices. paralytic ileus. peptic ulcer ...
  `www.mf.uni-lj.si/mf/fakulteta/prenova/medicina/mknjiga.pdf`

Case 7

Top categories: Diseases
→ Cushing Syndrome (41)

- **Adrenal Cancer - Signs, symptoms, complications, diagnosis - urologychannel**
  ... with Cushing's syndrome and an adrenal mass are diagnosed with adrenal cancer. ... High blood pressure (hypertension) Increased blood sugar, diabetes ...
  `www.urologychannel.com/adrenalcancer/symptoms.shtml`

- **Division of General Surgery: Massachusetts General Hospital Perspective on Endocrine Diseases**
  ... the operation is curative, preventing the long-term effects of hypertension. ... tumors, if a patient with Cushing's syndrome is found to have an adrenal mass, ...

- **Adrenal Cortical Carcinoma | Webpathology.com**
  Remarks: This patient presented with Cushing's syndrome and hypertension. ... This patient underwent laparoscopic adrenalectomy for a large left adrenal mass. ...

Case 8

Top categories: Diseases
→ Osteoma, Osteoid (1)

- **Web extra table [posted as supplied by authors]**
  10 yo boy with right thigh. pain and CT showed lytic. R hip lesion. hip lesion, older child, Eosinophilic granuloma, Osteoid osteoma. Osteoid osteoma ...
  `www.bmj.com/cgi/data/bmj.39003.640567.AE/DC1/1`

Case 9

- **Introduction to Case Studies**
  Case 7–Slowly-Growing Nodule. Case 9–Acute Respiratory Failure in an Elderly Man ... HRCT, pleural disease and nodules. Rheumatoid arthritis, nodule ...
  `http://pathhsw5m54.ucsf.edu/introduction.html`

Case 10

Diseases (145)
→ Bacterial Infections and Mycoses (12)
→ Bacterial Infections (6)
→ Gram-Negative Bacterial Infections (4)
→ Tick-Borne Diseases (4)
→ Ehrlichiosis (1)

- **Web extra table [posted as supplied by authors]**
  73 yo fever, thigh pain, urinary frequency, previous statin use. fever, bilateral thigh pain, weakness. Amyotrophy. Ehrlichiosis. No. 11. 30 yo female with fever ...
  `www.bmj.com/cgi/data/bmj.39003.640567.AE/DC1/1`

Case 11

Top categories: Diseases
→ Lymphoma (66)

- **Adult Non-Hodgkin Lymphoma Treatment - National Cancer Institute**
  Primary Central Nervous System Lymphoma Treatment ... with a locally invasive anterior mediastinal mass that may cause respiratory ...
  `www.cancer.gov/cancertopics/pdq/treatment/adult-non-hodgkins/HealthProfessional/page3`

- **UCCC - Cancer Information**
  ... with a locally invasive anterior mediastinal mass that may cause respiratory ... lymphoma marked by extensive necrosis and angioinvasion, most often presenting ...
  `http://www.uch.edu/CancerCenter/content/CancerInfo/Details.asp?Id=CDR0000062707&Type=Summary`

Case 12

Top categories: Diseases
→ Neurofibromatoses (21)

- **Neurofibromatosis Summary**
  Skin tumors can be surgically removed. ... multiple neurofibromas on ... patients with NF2 may also develop other brain tumors, as well as spinal tumors ...
  `www.bookrags.com/Neurofibromatosis`

- **eMedicine - Neurofibromatosis Type 2 : Article by Andrew L Wagner, MD**
  ... syndrome characterized by multiple schwannomas, meningiomas, and ependymomas. ... related to spinal tumors (eg, paraplegia, pain) and skin tumors each occurred in ...
  `www.emedicine.com/radio/topic475.htm`

Case 14

Tree: Diseases (959)
→ Cardiovascular Diseases (72)
→ Vascular Diseases (44)
→ Vasculitis (2)
→ Behcet Syndrome (1)

- **Rheumatoid Arthritis, Multiple Sclerosis, Lupus and Asthma Autoimmune Diet Program.**
  ... swollen joints (arthritis), unexplained fever, skin rashes, and kidney ... Vasculitis, Crohn's Disease, Ulcerative Colitis, Hives, Sjogren's disease (dry ...
  `www.biblelife.org/autoimmune.htm`

- **NewsRx - Behcet Disease News Articles**
  Eye inflammation can cause blurred vision; rarely, it causes pain and redness. ... digestive tract, such as ulcerative colitis and Crohn's disease, careful ...
  `www.newsrx.com/library/topics/Behcet-Disease.html`

Case 15

Top categories: Diseases
→ Amyloidosis (20)

- **4965KN Amyloidosis Consise Rev**
  presenting with congestive heart failure or nephrotic syndrome, symptoms ... urine, by quantitative Bence Jones proteinuria and by percentage and ...
  `www.myeloma.org/spider_articles/pdfs/amyloidosis/alconciseUK.html`

- **Amyloidosis and Waldenström's Macroglobulinemia**
  ... injected with Bence Jones proteins from ... failure or fatigue secondary to restrictive ... with nephrotic syndrome, amyloidosis is present in 10% of ...
  `https://www.amyloidosis.org/download/gertz.pdf`

- **Resolution of Heart Failure in Patients with AL Amyloidosis – Dubrey et al. 125 (6): 481 – Annals of Internal** ...
  ... cases, resolution of the nephrotic syndrome was associated with the ... Nephrotoxicity of human Bence Jones protein in rats: proteinuria and enzymuria profile. ...
  `www.annals.org/cgi/content/full/125/6/481`

Case 16

Tree: Diseases (29)
→ Neoplasms (4)
→ Neoplasms by Histologic Type (1)
→ Neoplasms, Nerve Tissue (1)
→ Neuroectodermal Tumors (1)
→ Neuroendocrine Tumors (1)
→ Paraganglioma (1)
→ Pheochromocytoma (1)

- **Nature Clinical Practice Urology | Modern management of pheochromocytoma | Article**
  ... hypertension.1 Orthostatic hypotension (a result of hypovolemia) ... employed in the past to distinguish adrenal tumors from an upper pole renal mass. ...
  `npg.nature.com/ncpuro/journal/v4/n11/full/ncpuro0962.html`

Case 17

No MeshTerm for Acute chest syndrome (ACS), but finds

- **Endothelin-1 Production during the Acute Chest Syndrome in Sickle Cell Disease – HAMMERMAN et al. 156 (1): 280 –** ...
  ... cough, pleuritic chest pain, and pulmonary infiltrates, occurs in 30-40% of ... yr-old black female with homozygous SS disease presented with diffuse back pain. ...
  `ajrccm.atsjournals.org/cgi/content/full/156/1/280`

- **Does Splinting From Thoracic Bone Ischemia and Infarction Contribute to the Acute Chest Syndrome in Sickle Cell** ...
  ... hospitalized with acute chest or back pain above the diaphragm. ... Incentive spirometry to prevent acute pulmonary complications in sickle cell diseases. ...
  `www.chestjournal.org/cgi/content/full/122/1/6`

Case 18

- **Web extra table [posted as supplied by authors]**
  10 yo boy with right thigh. pain and CT showed lytic. R hip lesion ... fibroma, astrocytoma, tumor, leiomyoma, scoliosis. Tuberous Sclerosis. Endometriosis ...
  `www.bmj.com/cgi/data/bmj.39003.640567.AE/DC1/1`

Case 22

Tree: West Nile Fever but also other

Diseases [658]
→ Virus Diseases [764]
→ Encephalitis, Viral [201]
→ Encephalitis, Arbovirus [99]
→ Encephalitis, Japanese [52]
→ Encephalitis, Tick-Borne [25]
→ Encephalitis, California [5]
→ Encephalitis, St. Louis [4]
→ West Nile Fever [3]

- **Healthcite.com**
  Leukemia - Chronic Lymphocytic Leukemia (CLL) Leukemia - Chronic Myelogenous Leukemia (CML) . . . West Nile Encephalitis (Yale) West Nile Fever Overview . . .
  `http://www.healthcite.com/topic.php?topic_id=43&h=2&topic=112&topic_name=West+Nile+Encephalitis+(Nile+Fever)`

- **Public Health**
  Cll. vg-Na. channel. DDT. kdr. esistance. of . . . fever, yellow fever, encephalitis, filariasis, West Nile fever and chikun . . .
  `www.bayervector.co.za/docs/Public%20Health%20No%2018%20Nov%202006_100.pdf`

Case 25

Tree: Diseases [974]
→ Cardiovascular Diseases [944]
→ Vascular Diseases [943]
→ Vasculitis [10]
→ Phlebitis [9] (Inflammation of a vein.)
→ Thrombophlebitis [9] (Inflammation of a vein associated with thrombus formation.)

- **CancerNetwork**
  Leading oncologists offer research and opinion on the screening, early detection, diagnosis, . . . First-line breast cancer . . . thrombophlebitis, portal vein thrombosis, and . . .
  `www.cancernetwork.com/drugs/femara.htm`

- **Breast Cancer in Canadian Women**
  deaths for Canadian women, breast cancer is the most . . . thrombosis, thrombophlebitis, portal vein thrombosis and pulmonary embolism. . . .
  `www.touchbriefings.com/download.cfm?fileID=4191&action=downloadFile`

- **DESCRIPTION**
  tissues and in the cancer tissue itself can therefore be achieved by . . . thrombophlebitis, portal vein thrombosis and pulmonary embolism. Cardiovascular events . . .
  `www.pharma.us.novartis.com/product/pi/pdf/Femara_T2000-45.pdf`

Case 26

Top categories: Diseases
→ Cardiomyopathy, Hypertrophic [22] (Hypertrophic Obstructive Cardiomyopathy)

- **Grand Rounds - Hammersmith Hospital: Cardiac arrest and hypertrophic cardiomyopathy – Lefroy 309 (6964): 1277 –** . . .
  . . . term prognosis of survivors of out-of-hospital cardiac arrest. . . . related to cardiac arrest and syncope in young patients with hypertrophic cardiomyopathy. . . .
  `www.bmj.com/cgi/content/full/309/6964/1277`

- **Documented exercise-induced cardiac arrest in a paediatric patient with hypertrophic cardiomyopathy – Pedrote et al**. . .
  . . . a recurrence of cardiac arrest during exercise, which was successfully . . . ICD therapy is effective in young patients with HCM and previous cardiac arrest. . . .
  `europace.oxfordjournals.org/cgi/content/full/8/6/430`

- **Hypertrophic Cardiomyopathy**
  . . . role in cardiac arrest and subsequent death in some young professional athletes. . . .

Those who have an abnormal blood pressure response with exercise . . .
`www.metrohealth.org/body.cfm?id=1509`

Case 27

Tree: Diseases [982]
→ Nervous System Diseases [949]
→ Central Nervous System Diseases [254]
→ Brain Diseases [230]
→ Dementia [38]
→ Creutzfeldt-Jakob Syndrome [17]

- **Slow virus infections : Epilepsy.com/Professionals**
  Insomnia, dysautonomia, ataxia. No typical EEG changes. Epidemiology and diagnosis of CJD . . . about sudden unexpected death in epileptic patients (SUDEP) . . .
  `professionals.epilepsy.com/page/infectious_slow.html`
- **CIGNA - Creutzfeldt Jakob Disease**
  . . . of confusion, depression, forgetfulness, sleeping difficulties (insomnia), and . . . voluntary movements (cerebellar ataxia), with associated unsteadiness, . . .
  `www.cigna.com/healthinfo/nord33.html`
- **Hearing loss as the initial presentation of Creutzfeldt-Jakob disease. - Free Online Library**
  . . . year history of progressive dementia, spasticity, ataxia, and startle myoclonus. . . . Variant Creutzfeldt-Jakob disease death, United States.(RESEARCH) . . .
  `www.thefreelibrary.com/Hearing+loss+as+the+initial+presentation+of+Creutzfeldt-Jakob+disease-a0122258001`

Case 28

Top categories: Diseases (61))
→ Churg-Strauss Syndrome [3]

- **Laboratory imposed restrictions on ANCA testing – 63 (5): 594 – Annals of the Rheumatic Diseases**
  The laboratory has performed ANCA testing only when the request form indicated . . . haematuria (requests from the renal/transplant unit), Churg-Strauss syndrome, . . .
  `ard.bmj.com/cgi/content/extract/63/5/594`
- **O OFF IIRREELLAANNDD**
  include haemoptysis (13% of patients), cystic bone lesions (4-20% of . . . Some 70-75% of patients with Churg-Strauss syndrome have ANCA . . .
  `www.hospitaldoctor.ie/hospital_doctor/pdfs/HOS_DOC_MARCH_APRIL_05.pdf`
- **Churg-Strauss Syndrome - Patient UK**
  Pulmonary: asthma, pneumonitis and haemoptysis . . . patients are perinuclear-ANCA (p-ANCA) positive (antimyeloperoxidase antibodies) . . .
  `www.patient.co.uk/showdoc/40024815/`

Case 29

Top five & more:
Diseases → Dermatomyositis (4)

- **eMedicine - Dermatomyositis : Article by Jeffrey P Callen, MD**
  ... is an idiopathic inflammatory myopathy (IIM) with characteristic cutaneous findings. ... terminal attack complex, calcinosis, heliotrope rash, Gottron papules ...
  `www.emedicine.com/med/topic2608.htm`

- **Dermatomyositis**
  ... disclosed myositis and a neurogenic myopathy in another one. ... Heliotrope rash-Purplish discoloration and edema of the periorbital tissues. Poikiloderma ...
  `www.thedoctorsdoctor.com/diseases/dermatomyositis.htm`

Case 30

Top categories:
Diseases → Cat-Scratch Disease (13)

- **NEJM – Recent Featured Images in Clinical Medicine**
  After a second renal transplant failed, he began to undergo dialysis. ... had no fever, urinary tract symptoms, ... Cat Scratch Disease Lymphadenopathy ...
  `content.nejm.org/misc/eicm.shtml`

- **Giessen Research Center in Infectious Diseases**
  ... infections in the genitourinary tract of renal transplant recipients. ... Cat scratch disease due to Bartonella henselae infection mimicking parotid malignancy. ...
  `www.uniklinikum-giessen.de/grid/publication.html`

- **VUMC Research; display faculty**
  ... year-old renal transplant patient with persistent fever, pancytopenia, and ... R D, Edwards, K M. Cat scratch disease: detection of Bartonella henselae ...
  `medschool1.mc.vanderbilt.edu/facultydata/php_files/show_faculty.php?id3=1421`

Case 33

Top categories:
Diseases → Telangiectasia, Hereditary Hemorrhagic (33)

- **Telangiectasia, Hereditary Hemorrhagic, diagnosis**
  ... presented with recurrent epistaxis, telangiectasias and haemangiomas, suggesting ... previous investigations had also shown multiple polyps of the stomach. ...
  `lib.bioinfo.pl/meid:86020`

- **Epistaxis [Nosebleed] (Alarming Signs and Symptoms: Lippincott Manual of Nursing Practice Series) - WrongDiagnosis**...
  Nasal polyps (Handbook of Diseases) ... Epistaxis commonly begins at puberty in patients with hereditary hemorrhagic telangiectasia. ...
  `www.wrongdiagnosis.com/symptoms/nose_symptoms/book-causes-13a.htm`

Case 34

Top categories:
Diseases → Epidermal Necrolysis, Toxic (4)

- **eMedicine - Toxic Epidermal Necrolysis : Article by Gregory P Garra, DO**
  ... from other severe bullous skin diseases such as ... cause GI hemorrhage, respiratory failure, ocular abnormalities, and genitourinary lesions. ...
  `www.emedicine.com/EMERG/topic599.htm`

- **Severe drug-related skin reaction: toxic epidermal necrolysis caused by carbamazepine**
  ... a severe bullous skin disease six weeks after starting carbamazepine therapy due ... systemic inflammatory reaction occurred with subsequent respiratory failure. ...
  `www.medscape.com/medline/abstract/15455296?prt=true`

- **Arch Dermatol – Intravenous Immunoglobulin Treatment for Stevens-Johnson Syndrome and Toxic Epidermal Necrolysis: A**...
  ... were nevirapine (7), carbamazepine (5), cotrimoxazole (3) ... 34-2005 – A 10-Year-Old Girl with a Bullous Skin Eruption and Acute Respiratory Failure. ...
  `archderm.ama-assn.org/cgi/content/full/139/1/33`

Case 37

Tree: Diseases [938]
→ Cardiovascular Diseases [897]
→ Heart Diseases [891]
→ Heart Arrest [875]
→ Death, Sudden, Cardiac [144]
→ Brugada Syndrome [5]

- **eMedicine - Brugada Syndrome : Article by Hugues Abriel, MD, PhD**
  ... ventricular tachyarrhythmias, syncope, cardiac arrest, sudden death, sudden ... In many cases, cardiac arrest occurs during sleep or rest. ...
  `www.emedicine.com/med/topic3736.htm`

- **Brugada syndrome - Genetics Home Reference**
  ... by unexpected cardiac arrest in young adults, usually at night during sleep. ... dominant ; calcium ; cardiac ; cardiac arrest ; cell ; channel ; complication ; ...
  `ghr.nlm.nih.gov/condition=brugadasyndrome`

- **CIGNA - Brugada Syndrome**
  ... individuals, who die from cardiac arrest during sleep with no apparent or identifiable cause. ... individual goes into cardiac arrest and possibly sudden ...
  `www.cigna.com/healthinfo/nord1157.html`

- **Orphanet Journal of Rare Diseases | Full text | Brugada syndrome**
  ... typically occurring at rest or during sleep (in individuals in their third or ... and secondary prophylaxis of cardiac arrest, the identification of high-risk ...
  `www.ojrd.com/content/1/1/35`

A.2 TREC Genomics 2006 Question and GoWeb Answers

#160: What is the role of PrnP in mad cow disease?

Query: `PrnP`
Result:
⇒ 1000 of 5,880
⇒ 29 snippets relate to the term "Encephalopathy, Bovine Spongiform"

378: **Transmissible spongiform encephalopathy: Definition from Answers.com**
Transmissible Spongiform Encephalopathy Bovine spongiform encepalopathy (BSE) is a transmissible, ... Mutations in the PRNP gene cause prion disease. ...
`www.answers.com/topic/spongiform-encephalopathy`

#161: What is the role of IDE in Alzheimer's disease?

Query: `IDE Alzheimer`
Result: 1000 of 12712

1: **Insulin-Degrading Enzyme as a Downstream Target of Insulin Receptor** ...
effect relationship between insulin signaling and IDE upregulation. ... P85) was correlated with reduced IDE in Alzheimer's disease (AD) brains and in ...
`alzheimer.neurology.ucla.edu/pubs/IDEzhao.pdf`

2: **Insulin degrading enzyme - Wikipedia, the free encyclopedia**
1 IDE and Alzheimer's Disease. 2 IDE Structure and Function. 3 References. 4 External links ... between IDE, Aβ degradation, and Alzheimer's disease. ...
`en.wikipedia.org/wiki/Insulin_degrading_enzyme`

#162: What is the role of MMS2 in cancer?

Query: `MMS2`
Result:
⇒ 1000 of 3245
⇒ 9 related to DNA Damage

41: **SGD Curated Paper**
... concerted action of RAD5 with UBC13 and MMS2 in DNA damage repair is given by ... Finally, it is shown that MMS2, like UBC13 and many other repair genes, is ...
`db.yeastgenome.org/cgi-bin/reference/reference.pl?dbid=S000061270`

#163: What is the role of APC (adenomatous polyposis coli) in colon cancer?

Query: `APC adenomatous polyposis coli`
Result: 1000 of 7009

1: **APC - adenomatous polyposis coli - Genetics Home Reference**
The official name of this gene is "adenomatous polyposis coli." APC is the gene's official symbol. ... adenomatous polyposis - caused by mutations in the APC ...
`ghr.nlm.nih.gov/gene=apc`

#164: What is the role of Nurr-77 in Parkinson's disease?

Query: `Nurr-77`
Result:
⇒ 1000 of 1163
⇒ 7 snippets relate to the term "Parkinson Disease"

40: **The aetiology of idiopathic Parkinson's disease**
Nurr 1 was first recognised as a transcription factor that was primarily . . . Its close structural relation to Nur 77 led to its identification in stimulated . . .
`www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1187126`

132: **Concise Review: Therapeutic Strategies for Parkinson Disease Based on** . . .
. . . nuclear related receptor 1 (Nurr-1), thereby withdrawing the cells of the cell . . . in the SVZ and the substantia nigra of the healthy adult rat brain [77, 98] . . .
`stemcells.alphamedpress.org/cgi/content/full/25/2/263`

221: **Parkinson's disease: piecing together a genetic jigsaw – Dekker et al** . . .
. . . study decreased rapidly with later onset: 77% of patients with onset of disease . . . agenesis of mesencephalic dopaminergic neurons in Nurr-1 deficient mice. . . .
`brain.oxfordjournals.org/cgi/content/full/126/8/1722`

#165: How do Cathepsin D (CTSD) and apolipoprotein E (ApoE) interactions contribute to Alzheimer's disease?

Query: `"Cathepsin D" "apolipoprotein E"`
Result:
⇒ 785
⇒ 74 snippets relate to the term "Alzheimer Disease"

14: **Apolipoprotein E and Alzheimer's disease: The protective effects of** . . .
Apolipoprotein E, e4 allele as a major risk factor for sporadic . . . expression of apolipoprotein E and cathepsin D in astrocytes . . .
`ladulab.anat.uic.edu/PDFs/Rebeck review.pdf`

#166: What is the role of Transforming growth factor-beta1 (TGF-beta1) in cerebral amyloid angiopathy (CAA)?

Query: `TGF-beta1 cerebral amyloid angiopathy`
Result: 115

1: **Alzheimer's Disease**
. . . report that TGF-beta1 induces amyloid-beta deposition in cerebral blood vessels . . . cerebral blood vessels of patients with cerebral amyloid angiopathy. . . .
`md1.csa.com/hottopics/alzheimers/97biblio36.html`

#167: How does nucleoside diphosphate kinase (NM23) contribute to tumor progression?

Query: `NM23 tumor progression`
Result: 1000 of 3289

2: **Increased Lung Metastasis in Transgenic NM23-Null/SV40 Mice with** . . .
. . . has a dual role in tumor progression: 1) its overexpression in primary tumors at . . . we studied hepatic tumor progression in NM23-M1 null mice with . . .
`jnci.oxfordjournals.org/cgi/content/full/97/11/836`

#168: How does BARD1 regulate BRCA1 activity?

Query: `BARD1 BRCA1`
Result: 1000 of 1941

1: **BARD1 - BRCA1 associated RING domain 1 - Genetics Home Reference**
The BARD1 and BRCA1 proteins work together to repair damaged DNA. . . . shown that the BARD1 protein binds to the BRCA1 protein, which stabilizes both . . .
`ghr.nlm.nih.gov/gene=bard1`

3: **BARD1 - Wikipedia, the free encyclopedia**
Mutations in the BRCA1-associated RING domain (BARD1) gene in primary breast, . . . Functional interaction of BRCA1-associated BARD1 with polyadenylation factor . . .
`en.wikipedia.org/wiki/BARD1`

#169: How does APC (adenomatous polyposis coli) protein affect actin assembly?

Query: `adenomatous polyposis coli actin assembly`
Result: 1000 of 2,130

3: **The adenomatous polyposis coli protein.**
Adenomatous polyposis coli tumor suppressor protein has signaling activity in . . . associates with microtubules in vivo and promotes their assembly in vitro. . . .
`www.pubmedcentral.nih.gov/articlerender.fcgi?artid=395695`

#170: How does COP2 contribute to CFTR export from the endoplasmic reticulum?

Query: `COP2 CFTR`
Query: `COP2 CFTR endoplasmic reticulum`
no results other than TREC related

#171: How does Nurr-77 delete T cells before they migrate to the spleen or lymph nodes and how does this impact autoimmunity?

Query: `Nurr-77 T cell`
no relevant response

#172: How does p53 affect apoptosis?

Query: `p53 apoptosis`
Result: 1000 of 74,203

4: **Apoptosis - the p53 network – Haupt et al. 116 (20): 4077 – Journal** . . .
However, p53 can also promote apoptosis by a transcription-independent mechanism . . . In turn, p53 induces either viable cell growth arrest or apoptosis. . . .
`jcs.biologists.org/cgi/content/full/116/20/4077`

#173: How do alpha7 nicotinic receptor subunits affect ethanol metabolism?

Query: `alpha7 nicotinic receptor ethanol`
Result: 357

1: **Alpha7 nicotinic receptor activation inhibits etha**. . .**[J Neurochem. 2002** . . .
. . . selectively activates alpha7 nicotinic receptors in a concentration . . . alpha7 nicotinic receptor-mediated protection against ethanol-induced oxidative . . .
`www.ncbi.nlm.nih.gov/pubmed/12065644`

#174: How does BRCA1 ubiquitinating activity contribute to cancer?

Query: `BRCA1 ubiquitinating`
Result:
⇒ 94 hits
⇒ 14 snippets relate to the term “Neoplasms”

35: **The ubiquitin system: pathogenesis of human diseases and drug targeting**
mutations of BRCA1 have been reported in an extremely. high frequency in breast carcinoma . . . by ubiquitinating an actin-associated protein, which in turn . . .
`www.elsevier.com/framework_aboutus/pdfs/ciechanover01.pdf`

#175: How does L2 interact with L1 to form HPV11 viral capsids?

Query: `L1 L2 HPV11`
Result: 171

1: **Interactions between Papillomavirus L1 and L2 Capsid Proteins**
. . . (HPV11) L1 and HPV11 glutathione S-transferase (GST) L2 fusion proteins . . . we sought to define the domain(s) on HPV11 L2 that interact with L1 pentamers. . . .
`www.pubmedcentral.nih.gov/articlerender.fcgi?artid=152166`

#176: How does Sec61-mediated CFTR degradation contribute to cystic fibrosis?

Query: `Sec61 CFTR`
Result:
⇒ 404
⇒ 55 snippets relate to the term “Cystic Fibrosis”

21: **Diffusional Mobility of the Cystic Fibrosis Transmembrane Conductance** . . .
ductance regulator protein (CFTR) cause cystic fibrosis. . . . between the Sec61 translocation system and CFTR that was. enhanced by proteasome inhibition. . . .
`www.uark.edu/campus-resources/mivey/cellphys/cftr_frap.pdf`

#177: How do Bop-Pes interactions affect cell growth?

Query: `Bop Pes cell growth`
Result:
⇒ 226
⇒ 13 snippets relate to the term "cell growth"

226: **Parasitology Research**
Significant role of Bop1 in cellular growth was initially suggested from a . . . biogenesis and cell proliferation via a direct interaction with Pes (Lapik et al. . . .
`parasitology.informatik.uni-wuerzburg.de/login/n/h/...`

#178: How do interactions between insulin-like GFs and the insulin receptor affect skin biology?

Query: `insulin-like GF insulin receptor`
Query: `insulin-like GF insulin receptor skin`
Query: `Insulin-like growth factor insulin receptor skin`
Result:
⇒ 1000 of 9497
⇒ no real answer in the snippets

#179: How do interactions between HNF4 and COUP-TF1 suppress liver function?

Query: `HNF4 COUP-TF1`
Query: `HNF4 COUP-TF1 liver function`
Result:
⇒ 90
⇒ no real answer, all mostly related to Hepatitis virus, but not to question

#180: How do Ret-GDNF interactions affect liver development?

Query: `Ret GDNF liver`
Result:
⇒ 1000 of 1026
⇒ 13 snippets relate to the term "liver development"

1: **The RET–Glial Cell-derived Neurotrophic Factor (GDNF) Pathway** . . .
GDNF–RET pathway regulates cell migration, prolif- eration, survival, . . . tor is essential for liver development. Nature. 373:699?702. Tapon, N., and A. Hall. . . .
`www.jcb.org/cgi/reprint/142/5/1337.pdf`

3: **GDNF promotes tubulogenesis of GFRalpha1-expressing MDCK cells by Src** . . .
The receptor complex for GDNF consists of Ret receptor tyrosine kinase and . . . Scatter factor/hepatocyte growth factor is essential for liver development. Nature. . . .
`www.jcb.org/cgi/content/full/161/1/119`

6: **GDNF and its receptors in the regulation of the ureteric branching**
The signalling receptor complex for GDNF includes a dimer of Ret receptor tyrosine . . . growth factor is essential for liver development. Nature. 373: 699-702. . . .
`www.ijdb.ehu.es/ft413.pdf`

#181: How do mutations in the Huntingtin gene affect Huntington's disease?

Query: `Huntingtin gene`
Result: 1000 of 10752

1: **Huntingtin - Wikipedia, the free encyclopedia**
The Huntingtin gene, also called HD (Huntington disease) gene, or the IT15 . . . Categories: Genes on chromosome 4 | Human proteins | Genes | Genes associated . . .
`en.wikipedia.org/wiki/Huntingtin`

#182: How do mutations in Sonic Hedgehog genes affect developmental disorders?

Query: `Sonic Hedgehog gene`
Result:
⇒ 1000 of 23,086
⇒ 47 articles relating to "Holoprosencephaly" (top disease)

188: **How a Hedgehog might see holoprosencephaly – Roessler and Muenke 12** . . .
. . . effect that the loss of a gene would have on the expression of Hh target genes. . . . and defective axial patterning in mice lacking sonic hedgehog gene function. . . .
`hmg.oxfordjournals.org/cgi/content/full/12/suppl_1/R15`

274: **Emerging Roles for Hedgehog-Patched-Gli Signal Transduction in** . . .
. . . Hedgehog (Hh) genes (Sonic hedgehog, Shh; Indian hedgehog, Ihh; . . . Identification of sonic hedgehog as a candidate gene responsible for holoprosencephaly. . . .
`www.biolreprod.org/cgi/content/full/69/1/8`

#183: How do mutations in the NM23 gene affect tracheal development?

Query: `NM23 tracheal development`
Result:
⇒ 134
⇒ 5 snippets relate to the term "Mutation"

14: **Site-directed Mutation of Nm23-H1. MUTATIONS LACKING MOTILITY** . . .
Nm23-H1P96S, a Drosophila developmental mutation homolog, . . . and functions synergistically with shi/dynamin during tracheal development. Genes & Dev. . . .
`www.jbc.org/cgi/content/abstract/272/9/5525`

#184: How do mutations in the Pes gene affect cell growth?

Query: `Pes gene cell growth`
Query: `Pes "cell growth"`
lots of results, mesh term "mutation", no real answer

#185: How do mutations in the hypocretin receptor 2 gene affect narcolepsy?

Query: `hypocretin receptor 2 narcolepsy`
Result:
⇒ 1,826
⇒ 279 snippets relate to the term "Mutation"

29: **Identification and Functional Analysis of Mutations in the Hypocretin** . . .
. . . mutations of the Hypocretin/Orexin-receptor-2 (Hcrtr2) gene were identified . . . The Hypocretin 2 receptor from a Dachshund that has narcolepsy was generated by . . .
`www.genome.org/cgi/content/full/11/4/531`

#186: How do mutations in the Presenilin-1 gene affect Alzheimer's disease?

Query: `Presenilin-1 Alzheimer`
Result:
⇒ 1000 of 10498
⇒ 296 articles relating to "Mutation"

34: **Association between presenilin-1 Glu318Gly mutation and familial** . . .
. . . presenilin-1 mutations in early-onset Alzheimer's disease . . . A mutation in Alzheimer's disease destroying a splice acceptor site in the presenilin-1 gene. . . .
`www.nature.com/cgi-taf/DynaPage.taf?file=/mp/.../v7/n7/full/4001072a.html`

#187: How do mutations in familial hemiplegic migraine type 1 (FHM1) gene affect calcium ion influx in hippocampal neurons?

Query: `FHM1 calcium neuron`
no real answer

APPENDIX B

ADDITIONAL DATA FOR TREC GENOMICS QUESTION ANSWERING

B.1 Mapping of TREC Genomics 2006 Question to Query Parameters

#160: What is the role of PrnP in mad cow disease?

“mad cow disease” – Term: `mesh#16643`
“Prnp” – Protein:

`P61767, P61766, P51780, Q95176, P52113, P40257, P40252, Q5UJG7, P10279, P67995, P04156, Q95M08, Q5UJH8, Q68G95, Q5UAF1, P04273, O18754, P67990, P49927, P67994, P67987, P40244, Q5UJG3, P40247, P51446, P40245, Q60506, O46501, P67992, Q5UJG1, P40246, Q7JK02, P67996, P40258, P67991, P52114, P47852, P40248, Q95270, P67997, P40255, P67986, Q7JIY2, Q5XVM4, Q5UJI7, P13852, P04925, P67988, Q60468, P67989, Q9Z0T3, Q6EH52, P79141, Q7JIH3, P40249, Q95211, Q95174, P67993, P61761, P23907, P40256, P61762, P40251, Q5UJH0, P61768, P27177`

#161: What is the role of IDE in Alzheimer’s disease?

“Alzheimers disease” – Term: `mesh#544`
“IDE” – Protein: `P14735, P22817, Q24K02, P35559, Q9JHR7`

#162: What is the role of MMS2 in cancer?

“cancer” – Term: `mesh#9369`
“MMC” – Protein: `P0A3S0, P0A3R9`

#163: What is the role of APC (adenomatous polyposis coli) in colon cancer?

“adenomatous polyposis coli” – Term: `mesh#11125`
“colon cancer” – Term: `mesh#3110`

#164: What is the role of Nurr-77 in Parkinson’s disease?

“Parkinsons disease” – Term: `mesh#20734`
“Nur-77” – Protein: `P22829, P12813`

#165: How do Cathepsin D (CTSD) and apolipoprotein E (ApoE) interactions contribute to Alzheimer’s disease?

“Cathepsin D” – Term: `mesh#2402`
“apolipoprotein E” – Term: `mesh#1057`
“Alzheimers disease” – Term: `mesh#544`

#166: What is the role of Transforming growth factor-beta1 (TGF-beta1) in cerebral amyloid angiopathy (CAA)?

“TGF-beta1” – Term: `go#0032905`, `mesh#53773`
“cerebral amyloid angiopathy” – Term: `mesh#16657`

#167: How does nucleoside diphosphate kinase (NM23) contribute to tumor progression?

“tumor progression” – Term: `mesh#9369`
“nucleoside diphosphate kinase” – Protein: `P15531, P15532`

#168: How does BARD1 regulate BRCA1 activity?

“BARD1” – Protein: `Q9QZH2, O70445, Q99728`
“BRCA1” – Protein:
`P48754, O54952, P38398, Q95153, Q6J6I9, Q6J6J0, Q9GKK8, Q6J6I8, Q864U1`

#169: How does APC (adenomatous polyposis coli) protein affect actin assembly?

“adenomatous polyposis coli” – Term: `mesh#11125`
“actin assembly” – Term: `mesh#199`

#170: How does COP2 contribute to CFTR export from the endoplasmic reticulum?

“COP2” – Protein: `Q53496`
“CFTR” – Protein:
`Q07E42, Q7JII8, Q09YJ4, Q00553, Q09YH0, Q2QL74, Q108U0, Q2IBE4, Q07DZ6, Q2QLA3, Q00552, Q07DY5, Q2QLC5, Q07E16, Q07DV2, Q2QLE5, P26363, Q5U820, Q2QLH0, P26362, Q00554, Q2IBB3, P26361, P35071, Q09YK5, Q2IBA1, Q7JII7, Q2QLB4, Q2IBF6, P34158, P13569, Q6PQZ2, Q00555, Q5D1Z7, Q07DW5, Q00PJ2, Q2QL83, Q9TUQ2, Q07DX5, Q2QLF9, Q9TSP5`
“export” – Keyword: `export`

#171: How does Nurr-77 delete T cells before they migrate to the spleen or lymph nodes and how does this impact autoimmunity?

“T cell” – Term: `mesh#13601`
“lymph nodes” – Term: `mesh#8198`
“autoimmunity” – Term: `mesh#15551`

"Nur-77" – Protein: P22829, P12813

#172: How does p53 affect apoptosis?

"apotosis" – Term: go#0006915, mesh#17209
"p53" – Protein:
P79820, P10361, O93379, Q9W678, O09185, Q9TTA1, Q8SPZ3, P61260, O57538, P04637, P02340, P56424, P79892, P56423, Q9TUB2, Q92143, Q9W679, Q42578, O12946, Q29537

#173: How do alpha7 nicotinic receptor subunits affect ethanol metabolism?

"alpha7 nicotinic receptor subunits" – Term: mesh#11978
"ethanol metabolism" – Term: go#0006067, mesh#431

#174: How does BRCA1 ubiquitinating activity contribute to cancer?

"ubiquitinating" – Term: mesh#25801
"cancer" – Term: mesh#9369
"BRCA1" – Protein:
P48754, O54952, P38398, Q95153, Q6J6I9, Q6J6J0, Q9GKK8, Q6J6I8, Q864U1

#175: How does L2 interact with L1 to form HPV11 viral capsids?

"human papillomavirus type 11 viral capsids" – Term: mesh#52140, go#0019028
"L1" – Protein:
P36731, P50822, Q07861, P69899, P50789, P21140, Q02051, P50878, Q07860, P36736, P54669, P36737, P50805, P49165, P03103, O15594, Q02515, P17378, Q89828, P03104, P36738, P50818, P50788, P26537, P11369, P50823, P17377, P17388, P09180, P11326, P26321, P36741, P36742, Q9IR52, P27557, Q02050, P36743, P50807, P50819, P19833, P14117, P50813, P27233, P50806, P36732, P03102, P49669, P03101, P36733, P06416, P06417, P36740, P26535, P06456, P25486, P50791, P50812, Q80961, Q28346, P17376, Q05136, P27232, Q05137, P50793, P04012, Q9SF40, P08341, P26536, P03099, Q29187, Q05138, P50790, Q9NUQ9, Q02480, P06917, Q9D8E6, P06794, Q02514, P50814, P50787, P50786, Q02273, P69898, P50821, Q02274, P50792, Q9XF97, Q801X7, P22424, P22162, Q07874, P49691, P24838, P22163, P50820, P50826, P50816, P50817, P36735, P27964, Q05113, P52956, P36578, P06458, P50824, P50815, P50825, P36734, P08170
"l2" – Protein:
Q80939, P29766, P36756, P50800, P36758, Q9JH45, P26538, P11079, P50795, P27558, Q705F9, P03105, Q8BDG3, P07613, P11327, Q02275, P36762, Q9IR53, Q00273, Q676U7, P09439, P27235, Q07863, P36744, P26539, P27234, P26540, P35679, Q705H5, Q80918, P36747, P36760, O90729, P08342, P36748, P10664, P49626, P20843, Q80925, P22425, P33041, Q80932, P36763, P03110, P06418, Q84297, P03106, Q89892, P36764, P36749, P32553, P50827, P06793, Q80960, Q76RD1, O71024, P36746, P21141, P27965, P36765, P36754, P25487, P36755, P36745, Q89508, P24839, P36761, Q80946, P17389, P22165, Q80905, P36753, P50799, P06918, P50798, Q90WJ8, P24484, P04013, P36751, P03108, Q02276, P22164, P36752, P50801, P50796, P36750, P03109, Q80953, P06419, P06457, P36757, Q80912, Q07875, Q81023, P03107, P50794, P50797, Q07862

#176: How does Sec61-mediated CFTR degradation contribute to cystic fibrosis?

“cystic fibrosis” – Term: mesh#3550

“Sec61” – Protein:

Q752H7, P32915, Q96TW8, Q870W0, Q6FRY3, P78979, Q6CPY9, P79088, Q9P8E3, Q6BN08

“CFTR” – Protein:

Q07E42, Q7JII8, Q09YJ4, Q00553, Q09YH0, Q2QL74, Q108U0, Q2IBE4, Q07DZ6, Q2QLA3, Q00552, Q07DY5, Q2QLC5, Q07E16, Q07DV2, Q2QLE5, P26363, Q5U820, Q2QLH0, P26362, Q00554, Q2IBB3, P26361, P35071, Q09YK5, Q2IBA1, Q7JII7, Q2QLB4, Q2IBF6, P34158, P13569, Q6PQZ2, Q00555, Q5D1Z7, Q07DW5, Q00PJ2, Q2QL83, Q9TUQ2, Q07DX5, Q2QLF9, Q9TSP5

#177: How do Bop-Pes interactions affect cell growth?

“cell growth” – Term: go#0016049

“Bop” – Protein:

P33969, O93740, P60015, P51491, P33971, P97443, P03999, P69052, Q63652, O13092, P60573, P02945, P51490, P33972

“Pes” – Protein: P79741

#178: How do interactions between insulin-like GFs and the insulin receptor affect skin biology?

“insulin-like GF” – Term: go#0005159 mesh#7334

“insulin receptor” – Term: mesh#11972

“skin biology” – Keyword: skin

#179: How do interactions between HNF4 and COUP-TF1 suppress liver function?

“HNF4” – Term: mesh#51557

“COUP-TF1” – Protein: Q60632, P10589, Q9TTR8

“liver function” – Keyword: liver

#180: How do Ret-GDNF interactions affect liver development?

“liver development” – Term: go#0001889

“GDNF” – Term: mesh#51100

“Ret” – Protein: P67876, P35546, P07949

#181: How do mutations in the Huntingtin gene affect Huntington’s disease?

“Huntingtons disease” – Term: mesh#6816

“Huntingtin gene” – Protein: P51111, P42859, P42858, P51112

“mutations” – Keyword: mutation

#182: How do mutations in Sonic Hedgehog genes affect developmental disorders?

“developmental disorders” – Term: mesh#2658

“Sonic Hedgehog genes” – Protein:

Q02936, O13238, O13234, P79709, P79839, P79915, O13245, P56674, O13250, P79864, P79838, P79850, P79691, O13247, P79682, P79858, P79717, P79869, O13235, O13241 "mutations" – Keyword: `mutation`

#183: How do mutations in the NM23 gene affect tracheal development?
"NM23 gene" – Protein: `P15531, P15532`
"tracheal development" – Keyword: `trachea`
"mutations" – Keyword: `mutation`

#184: How do mutations in the Pes gene affect cell growth?
"cell growth" – Term: `go#0016049`
"Pes gene" – Protein: `P79741`
"mutations" – Keyword: `mutation`

#185: How do mutations in the hypocretin receptor 2 gene affect narcolepsy?
"hypocretin receptor 2 gene" – Term: `go#0042324`
Protein: `Q9TUP7, P56719, O62809, P58308, O43614`
"narcolepsy" – Term: `mesh#9290`

#186: How do mutations in the Presenilin-1 gene affect Alzheimer's disease?
"Alzheimers disease" – Term: `mesh#544`
"Presenilin-1 gene" – Term: `mesh#53764`
"mutations" – Keyword: `mutation`

#187: How do mutations in familial hemiplegic migraine type 1 (FHM1) gene affect calcium ion influx in hippocampal neurons?
"hippocampal neurons" – Term: `go#0021852, go#0021859, go#0021860, mesh#17966`
"familial hemiplegic migraine type 1 gene" – Keyword: (FMH1)
"calcium ion influx" – Keyword: `calcium`

B.2 Answers to TREC Genomics 2006 using TREC Genomics 2006 Corpus

#160: What is the role of PrnP in mad cow disease?

PMID: 14645932	POS: 75762-602	SubEvidences: 4
PMID: 15722546	POS: 91610-569	SubEvidences: 8
PMID: 15722547	POS: 71465-532	SubEvidences: 10
PMID: 12655107	POS: 68763-601	SubEvidences: 6
PMID: 12692300	POS: 79101-604	SubEvidences: 2
PMID: 15448379	POS: 31383-601	SubEvidences: 6
PMID: 16141216	POS: 0-982	SubEvidences: 5
PMID: 16099922	POS: 64650-361	SubEvidences: 2
PMID: 11562525	POS: 47470-525	SubEvidences: 2
PMID: 12237446	POS: 79208-521	SubEvidences: 2
PMID: 12388826	POS: 69232-525	SubEvidences: 2
PMID: 14718642	POS: 76066-441	SubEvidences: 2
PMID: 15448380	POS: 48187-362	SubEvidences: 2
PMID: 11752724	POS: 36851-476	SubEvidences: 4
PMID: 14634010	POS: 50894-543	SubEvidences: 4
PMID: 10993946	POS: 59873-481	SubEvidences: 2
PMID: 15483267	POS: 1191-2248	SubEvidences: 2
PMID: 11733532	POS: 5909-972	SubEvidences: 2
PMID: 11792707	POS: 5465-962	SubEvidences: 4
PMID: 9867801	POS: 1291-706	SubEvidences: 2
PMID: 11562524	POS: 672-3260	SubEvidences: 1
PMID: 12759373	POS: 3847-4045	SubEvidences: 1
PMID: 14573822	POS: 4234-6121	SubEvidences: 2
PMID: 15494372	POS: 3738-4121	SubEvidences: 2
PMID: 12082114	POS: 5055-1445	SubEvidences: 2
PMID: 15342797	POS: 14658-830	SubEvidences: 1
PMID: 15604451	POS: 3442-3689	SubEvidences: 2
PMID: 10573173	POS: 2088-2385	SubEvidences: 2
PMID: 11152454	POS: 6107-1427	SubEvidences: 4
PMID: 11278343	POS: 5039-653	SubEvidences: 2
PMID: 11842266	POS: 4821-2392	SubEvidences: 2
PMID: 12138171	POS: 8269-1914	SubEvidences: 4
PMID: 12454014	POS: 5054-1144	SubEvidences: 2
PMID: 14711812	POS: 3713-2191	SubEvidences: 2
PMID: 15483265	POS: 4900-2216	SubEvidences: 2
PMID: 7852415	POS: 4244-1629	SubEvidences: 2
PMID: 9261166	POS: 10022-1823	SubEvidences: 4
PMID: 15123682	POS: 62526-748	SubEvidences: 2
PMID: 15269390	POS: 3412-1813	SubEvidences: 2
PMID: 15557254	POS: 4833-3117	SubEvidences: 2
PMID: 15604453	POS: 3466-3275	SubEvidences: 2
PMID: 11086143	POS: 4766-4266	SubEvidences: 2
PMID: 11125179	POS: 3847-3003	SubEvidences: 1
PMID: 15310752	POS: 4164-3421	SubEvidences: 4
PMID: 15385582	POS: 4157-2329	SubEvidences: 2
PMID: 15831971	POS: 3506-2396	SubEvidences: 1

PMID: 16157591	POS: 3691-3893	SubEvidences: 4
PMID: 10466827	POS: 3620-2523	SubEvidences: 1
PMID: 10967124	POS: 9893-4148	SubEvidences: 2
PMID: 9111077	POS: 4484-1371	SubEvidences: 2

130 more hits found.

#161: What is the role of IDE in Alzheimer's disease

PMID: 15615772	POS: 104503-699	SubEvidences: 4
PMID: 10973971	POS: 37541-937	SubEvidences: 3
PMID: 15944156	POS: 71001-1956	SubEvidences: 6
PMID: 14764623	POS: 66784-612	SubEvidences: 8
PMID: 12566388	POS: 54002-747	SubEvidences: 2
PMID: 15749695	POS: 7568-1815	SubEvidences: 4
PMID: 11809755	POS: 49484-2804	SubEvidences: 2
PMID: 15277398	POS: 5487-982	SubEvidences: 6
PMID: 15347685	POS: 31843-2121	SubEvidences: 4
PMID: 15331662	POS: 4174-4282	SubEvidences: 2
PMID: 15100223	POS: 4693-4797	SubEvidences: 2
PMID: 12105192	POS: 71692-813	SubEvidences: 4
PMID: 15642747	POS: 3610-465	SubEvidences: 2
PMID: 16154999	POS: 4386-3621	SubEvidences: 4
PMID: 16162502	POS: 57279-1126	SubEvidences: 2
PMID: 15322125	POS: 5266-5364	SubEvidences: 2
PMID: 9499430		
PMID: 16027115	POS: 74132-1088	SubEvidences: 2
PMID: 10092675		
PMID: 12941771	POS: 5915-1915	SubEvidences: 6
PMID: 12765971	POS: 34011-10304	SubEvidences: 2
PMID: 14976159		
PMID: 12716770	POS: 33265-12952	SubEvidences: 2
PMID: 10075647		
PMID: 11893734		
PMID: 11532993		
PMID: 10818098		
PMID: 12464614		
PMID: 12754258		
PMID: 15781953	POS: 62609-24642	SubEvidences: 16
PMID: 9300660		
PMID: 9668163		
PMID: 15941712		
PMID: 11010978		
PMID: 11312271		
PMID: 10469843		
PMID: 12471021		
PMID: 15699037		
PMID: 10762618	POS: 53703-20429	SubEvidences: 1
PMID: 11285366	POS: 48377-17690	SubEvidences: 2
PMID: 15149955	POS: 51915-30284	SubEvidences: 2
PMID: 9545270		

PMID: 10764738
PMID: 10915790
PMID: 10952980
PMID: 11152691
PMID: 11679584
PMID: 11724769
PMID: 11729199
PMID: 11823454

70 more hits found.

#162: What is the role of MMS2 in cancer?

PMID: 9345010 POS: 45628-1335 SubEvidences: 2
PMID: 7836395
PMID: 10330050
PMID: 12775568
PMID: 12807822
PMID: 15611321
PMID: 9461625
PMID: 10784616
PMID: 11157488
PMID: 12928478
PMID: 14500812
PMID: 16033977
PMID: 9922457
PMID: 11675368
PMID: 11683386
PMID: 15632205
PMID: 10793105 POS: 107049-49794 SubEvidences: 2
PMID: 9192774
PMID: 12655091
PMID: 10212194
PMID: 10419890
PMID: 11861261
PMID: 12490721
PMID: 12591928
PMID: 12876075
PMID: 14759987
PMID: 9275079
PMID: 9890167
PMID: 11245607
PMID: 11356705
PMID: 12384403
PMID: 14668260
PMID: 15242874
PMID: 9427689

#163: What is the role of APC (adenomatous polyposis coli) in colon cancer?

PMID: 7890674	POS: 5093-1219	SubEvidences: 3
PMID: 8626604	POS: 5107-1457	SubEvidences: 4
PMID: 8872463	POS: 3271-1548	SubEvidences: 6
PMID: 8968744	POS: 1411-2069	SubEvidences: 30
PMID: 8977385	POS: 57561-19029	SubEvidences: 2
PMID: 9002677	POS: 21525-1214	SubEvidences: 6
PMID: 9002979	POS: 3604-1827	SubEvidences: 2
PMID: 9015311	POS: 0-227	SubEvidences: 12
PMID: 9020180	POS: 10240-2006	SubEvidences: 3
PMID: 9024214	POS: 4761-1241	SubEvidences: 15
PMID: 9024696	POS: 7753-788	SubEvidences: 1
PMID: 9024698	POS: 0-356	SubEvidences: 10
PMID: 9030594	POS: 38260-1107	SubEvidences: 2
PMID: 9045685	POS: 4561-2214	SubEvidences: 2
PMID: 9049250	POS: 68483-1789	SubEvidences: 2
PMID: 9054414	POS: 58580-2021	SubEvidences: 3
PMID: 9063750	POS: 29176-74	SubEvidences: 1
PMID: 9110993	POS: 2106-3038	SubEvidences: 4
PMID: 9116273	POS: 56897-551	SubEvidences: 2
PMID: 9139698	POS: 1814-1125	SubEvidences: 6
PMID: 9141569	POS: 5612-11952	SubEvidences: 2
PMID: 9151695	POS: 91813-19848	SubEvidences: 3
PMID: 9166410	POS: 92129-31096	SubEvidences: 1
PMID: 9182547	POS: 70213-1948	SubEvidences: 2
PMID: 9182672	POS: 71811-32235	SubEvidences: 1
PMID: 9199178	POS: 2196-3380	SubEvidences: 7
PMID: 9211858	POS: 29326-3150	SubEvidences: 2
PMID: 9235921	POS: 73845-1559	SubEvidences: 2
PMID: 9259273	POS: 64988-459	SubEvidences: 2
PMID: 9261095	POS: 25013-1469	SubEvidences: 2
PMID: 9265655	POS: 72883-1035	SubEvidences: 1
PMID: 9268294	POS: 0-266	SubEvidences: 4
PMID: 9285776	POS: 40107-317	SubEvidences: 2
PMID: 9314542	POS: 1652-4696	SubEvidences: 7
PMID: 9334345	POS: 128076-33110	SubEvidences: 1
PMID: 9348288	POS: 5949-1006	SubEvidences: 7
PMID: 9354661	POS: 39457-1536	SubEvidences: 2
PMID: 9361031	POS: 26877-1020	SubEvidences: 4
PMID: 9362520	POS: 49530-2331	SubEvidences: 3
PMID: 9362521	POS: 12759-5084	SubEvidences: 4
PMID: 9382877	POS: 9418-5446	SubEvidences: 4
PMID: 9396760	POS: 98990-38215	SubEvidences: 1
PMID: 9399837	POS: 20264-345	SubEvidences: 2
PMID: 9399850	POS: 1378-691	SubEvidences: 2
PMID: 9405493	POS: 19709-1360	SubEvidences: 7
PMID: 9405496	POS: 44621-473	SubEvidences: 2
PMID: 9417046	POS: 9294-708	SubEvidences: 2
PMID: 9425166	POS: 14794-1651	SubEvidences: 1
PMID: 9435542	POS: 56429-2332	SubEvidences: 4

PMID: 9435686	POS: 100753-18116	SubEvidences: 2

1110 more hits found.

#164: What is the role of Nurr-77 in Parkinson's disease?

No answers found for this topic.

#165: How do Cathepsin D (CTSD) and apolipoprotein E (ApoE) interactions contribute to Alzheimer's disease?

PMID: 9302273	POS: 28592-513	SubEvidences: 2
PMID: 9328480	POS: 22128-525	SubEvidences: 2
PMID: 9700208	POS: 51937-535	SubEvidences: 2
PMID: 11005793	POS: 166343-566	SubEvidences: 2
PMID: 11709540	POS: 129718-563	SubEvidences: 2
PMID: 12867662	POS: 32775-476	SubEvidences: 2
PMID: 15258178	POS: 43441-422	SubEvidences: 2
PMID: 15003956	POS: 1049-3599	SubEvidences: 2
PMID: 9516475	POS: 59725-1419	SubEvidences: 2
PMID: 10522973		
PMID: 11551970		
PMID: 11912196		
PMID: 14645225		
PMID: 14970212		
PMID: 15220353		

#166: What is the role of Transforming growth factor-beta1 (TGF-beta1) in cerebral amyloid angiopathy (CAA)?

PMID: 12626500
PMID: 15632190
PMID: 12668602

#167: How does nucleoside diphosphate kinase (NM23) contribute to tumor progression?

PMID: 9593706	POS: 47915-1428	SubEvidences: 2
PMID: 11694515	POS: 52667-829	SubEvidences: 2
PMID: 10552965	POS: 49544-1523	SubEvidences: 2
PMID: 12515726	POS: 2375-3445	SubEvidences: 4
PMID: 15247270	POS: 60591-3034	SubEvidences: 2
PMID: 9362334	POS: 72482-39481	SubEvidences: 3
PMID: 9449709		
PMID: 10446260		
PMID: 15498789	POS: 78608-20818	SubEvidences: 2
PMID: 7890749		
PMID: 9013567		
PMID: 16085756		

PMID: 7768941		
PMID: 12225991	POS: 70990-39966	SubEvidences: 3
PMID: 9892021		
PMID: 9020170		
PMID: 15928304		
PMID: 12488479		
PMID: 12756286		
PMID: 11139609		
PMID: 10451442	POS: 65417-28324	SubEvidences: 4
PMID: 9105051	POS: 108271-37657	SubEvidences: 2
PMID: 14718631		

#168: How does BARD1 regulate BRCA1 activity?

PMID: 9425226	POS: 1960-1528	SubEvidences: 65
PMID: 9525870	POS: 5537-1514	SubEvidences: 14
PMID: 9832560	POS: 4249-1605	SubEvidences: 103
PMID: 15855157	POS: 84181-2358	SubEvidences: 120
PMID: 15886201	POS: 9310-760	SubEvidences: 181
PMID: 10026184	POS: 2537-2581	SubEvidences: 144
PMID: 10764811	POS: 68281-1466	SubEvidences: 130
PMID: 10945975	POS: 7037-1224	SubEvidences: 20
PMID: 11278247	POS: 4993-1103	SubEvidences: 108
PMID: 11504763	POS: 6958-438	SubEvidences: 9
PMID: 11773071	POS: 8089-2790	SubEvidences: 146
PMID: 11925436	POS: 0-573	SubEvidences: 242
PMID: 11927591	POS: 38004-1037	SubEvidences: 108
PMID: 12431996	POS: 46891-1587	SubEvidences: 248
PMID: 12582233	POS: 30091-2966	SubEvidences: 35
PMID: 12915465	POS: 81772-743	SubEvidences: 13
PMID: 14638690	POS: 7468-983	SubEvidences: 111
PMID: 14976165	POS: 85041-538	SubEvidences: 268
PMID: 15166217	POS: 28629-1494	SubEvidences: 195
PMID: 15184379	POS: 5818-850	SubEvidences: 65
PMID: 15385441	POS: 43216-2126	SubEvidences: 32
PMID: 15632137	POS: 20012-3104	SubEvidences: 80
PMID: 15964842	POS: 85427-1114	SubEvidences: 6
PMID: 10869340	POS: 66643-1253	SubEvidences: 15
PMID: 10910891	POS: 106283-378	SubEvidences: 2
PMID: 15569676	POS: 10177-1626	SubEvidences: 134
PMID: 14610072	POS: 54834-827	SubEvidences: 5
PMID: 14871887	POS: 63157-1503	SubEvidences: 7
PMID: 14981089	POS: 77530-3454	SubEvidences: 5
PMID: 15159397	POS: 32649-1518	SubEvidences: 127
PMID: 11555636	POS: 25733-652	SubEvidences: 17
PMID: 11707511	POS: 11045-3867	SubEvidences: 10
PMID: 12730115	POS: 47726-1081	SubEvidences: 7
PMID: 15087457	POS: 78982-634	SubEvidences: 12
PMID: 9890972	POS: 4334-1310	SubEvidences: 8
PMID: 10722742	POS: 62638-1620	SubEvidences: 10

PMID: 10938285	POS: 6325-664	SubEvidences: 5
PMID: 11739404	POS: 25557-1851	SubEvidences: 5
PMID: 12611903	POS: 36628-431	SubEvidences: 13
PMID: 12670957	POS: 58100-762	SubEvidences: 6
PMID: 15872055	POS: 58096-453	SubEvidences: 2
PMID: 12700228	POS: 13224-1075	SubEvidences: 65
PMID: 15811849	POS: 36732-2223	SubEvidences: 14
PMID: 12082091	POS: 5909-1771	SubEvidences: 8
PMID: 15454491	POS: 8183-1634	SubEvidences: 11
PMID: 15466891	POS: 90885-482	SubEvidences: 10
PMID: 15547178	POS: 23928-2054	SubEvidences: 9
PMID: 15591324	POS: 56031-1931	SubEvidences: 9
PMID: 16079148	POS: 38519-1900	SubEvidences: 12
PMID: 15983032	POS: 44461-2465	SubEvidences: 16

12 more hits found.

#169: How does APC (adenomatous polyposis coli) protein affect actin assembly

PMID: 9049250	POS: 68483-1789	SubEvidences: 2
PMID: 16157700	POS: 2943-2269	SubEvidences: 1
PMID: 10026156	POS: 89168-3620	SubEvidences: 1
PMID: 9885247	POS: 113429-27672	SubEvidences: 1
PMID: 9660865	POS: 113000-19961	SubEvidences: 1
PMID: 9334345	POS: 128076-33110	SubEvidences: 1
PMID: 9024698	POS: 131680-22564	SubEvidences: 1
PMID: 9744888	POS: 80683-26406	SubEvidences: 1
PMID: 12055095		
PMID: 9382877	POS: 84917-30316	SubEvidences: 1
PMID: 9971746	POS: 119086-27844	SubEvidences: 1
PMID: 9435686	POS: 100753-18116	SubEvidences: 1
PMID: 10607757		
PMID: 9182672	POS: 71811-32235	SubEvidences: 1
PMID: 11408275	POS: 78917-23507	SubEvidences: 1
PMID: 15075229		
PMID: 15492045		
PMID: 15673685		
PMID: 16247021		
PMID: 11756422		
PMID: 12176738		
PMID: 12077140		
PMID: 9628899	POS: 120102-40456	SubEvidences: 1
PMID: 11950877		
PMID: 16314429		
PMID: 15456841		
PMID: 10747098		
PMID: 9396760	POS: 98990-38215	SubEvidences: 1
PMID: 10712269		
PMID: 15809307		
PMID: 12058019		
PMID: 15579911		

PMID: 15020669
PMID: 15657074
PMID: 16046480
PMID: 12006613
PMID: 12496241
PMID: 15504907
PMID: 11719546
PMID: 9348288
PMID: 15976449
PMID: 12154083
PMID: 14610057
PMID: 15955847
PMID: 10424893
PMID: 11564752
PMID: 12213835
PMID: 10681393
PMID: 16263762
PMID: 10748202

92 more hits found.

#170: How does COP2 contribute to CFTR export from the endoplasmic reticulum?

PMID: 15479737	POS: 32033-710	SubEvidences: 40
PMID: 15713669	POS: 67582-1193	SubEvidences: 3
PMID: 11799116	POS: 68758-1784	SubEvidences: 18
PMID: 15944403	POS: 58243-4468	SubEvidences: 1
PMID: 16037065		
PMID: 12107161		
PMID: 15078901		
PMID: 12538638		
PMID: 12105183		
PMID: 15987769		
PMID: 12711700		
PMID: 11673477		

#171: How does Nurr-77 delete T cells before they migrate to the spleen or lymph nodes and how does this impact autoimmunity?

PMID: 11956287	POS: 84343-24218	SubEvidences: 1
PMID: 9139721		
PMID: 9348309		
PMID: 10859336		
PMID: 10837419		
PMID: 12855571		

#172: How does p53 affect apoptosis?

PMID: 16275757	POS: 106293-488	SubEvidences: 13

```
PMID: 12655031    POS: 83473-494
PMID: 10562313    POS: 54409-378     SubEvidences: 12
PMID: 10573167    POS: 54681-404     SubEvidences: 1
PMID: 10620618    POS: 45283-331     SubEvidences: 11
PMID: 10688817    POS: 76275-457     SubEvidences: 1
PMID: 11158189    POS: 58805-379     SubEvidences: 2
PMID: 11159853    POS: 52740-438     SubEvidences: 14
PMID: 11337497    POS: 36519-1176    SubEvidences: 1
PMID: 11381085    POS: 95796-485     SubEvidences: 1
PMID: 11739155    POS: 28564-232     SubEvidences: 13
PMID: 11739635    POS: 75368-516
PMID: 12354779    POS: 127282-495
PMID: 12393587    POS: 100777-456    SubEvidences: 43
PMID: 12514185    POS: 5370-613      SubEvidences: 8
PMID: 12529339    POS: 7820-675      SubEvidences: 3
PMID: 15537749    POS: 0-413         SubEvidences: 7
PMID: 15687234    POS: 41451-229     SubEvidences: 4
PMID: 8626804     POS: 5230-1259     SubEvidences: 1
PMID: 8995241     POS: 49827-1186
PMID: 9020141     POS: 4361-458      SubEvidences: 19
PMID: 9023107     POS: 2340-740      SubEvidences: 1
PMID: 9058703     POS: 37753-939     SubEvidences: 10
PMID: 9060424     POS: 3385-974
PMID: 9108389     POS: 39970-505     SubEvidences: 27
PMID: 9129041     POS: 73265-508
PMID: 9148891     POS: 0-196         SubEvidences: 10
PMID: 9153226     POS: 32918-114     SubEvidences: 7
PMID: 9160668     POS: 62446-365     SubEvidences: 4
PMID: 9182545     POS: 0-211         SubEvidences: 11
PMID: 9207463     POS: 77972-362     SubEvidences: 5
PMID: 9214390     POS: 0-213         SubEvidences: 9
PMID: 9235895     POS: 36038-461
PMID: 9265655     POS: 37210-86      SubEvidences: 10
PMID: 9269753     POS: 45333-499     SubEvidences: 8
PMID: 9292505     POS: 88137-510     SubEvidences: 2
PMID: 9345011     POS: 62810-473
PMID: 9354678     POS: 84639-513     SubEvidences: 5
PMID: 9373250     POS: 43618-761     SubEvidences: 15
PMID: 9376567     POS: 349482-515    SubEvidences: 1
PMID: 9414264     POS: 231807-365    SubEvidences: 4
PMID: 9414295     POS: 102129-510    SubEvidences: 2
PMID: 9425165     POS: 8149-4973     SubEvidences: 6
PMID: 9456323     POS: 0-183         SubEvidences: 8
PMID: 9473234     POS: 65748-995     SubEvidences: 2
PMID: 9479003     POS: 6972-1299     SubEvidences: 6
PMID: 9497367     POS: 38527-958     SubEvidences: 2
PMID: 9531600     POS: 80798-545
PMID: 9531611     POS: 65847-546     SubEvidences: 4
PMID: 9531612     POS: 65837-4129    SubEvidences: 9
```

3481 more hits found.

#173: How do alpha7 nicotinic receptor subunits affect ethanol metabolism?

No answers found for this topic.

#174: How does BRCA1 ubiquitinating activity contribute to cancer?

PMID: 15833741	POS: 60691-885	SubEvidences: 2
PMID: 11181458		
PMID: 10223177	POS: 49796-69941	SubEvidences: 1
PMID: 15878912		
PMID: 12538348		
PMID: 15073042		
PMID: 15465831		
PMID: 14652236		
PMID: 14699124		
PMID: 10910891		
PMID: 11818503		

#175: How does L2 interact with L1 to form HPV11 viral capsids?

PMID: 15302958	POS: 65464-418	SubEvidences: 2
PMID: 12533712	POS: 45199-1135	SubEvidences: 2
PMID: 10026203	POS: 6348-2009	SubEvidences: 2
PMID: 12560332	POS: 64084-977	SubEvidences: 2
PMID: 12117707		
PMID: 10567657	POS: 3602-3495	SubEvidences: 3
PMID: 16169845		
PMID: 10725437		
PMID: 12692285		
PMID: 10501500		
PMID: 11602792		
PMID: 12771418		
PMID: 10644830		
PMID: 11086127		
PMID: 12185286		
PMID: 14602899		
PMID: 10580054		
PMID: 14573795		
PMID: 11312251		
PMID: 12692268		
PMID: 10501503		
PMID: 10793105	POS: 107049-49794	SubEvidences: 3
PMID: 15831940		
PMID: 11304544		
PMID: 11181775		
PMID: 15914847		
PMID: 14573808		

#176: How does Sec61-mediated CFTR degradation contribute to cystic fibrosis?

PMID: 11054417	POS: 10679-3023	SubEvidences: 1
PMID: 10601334	POS: 69691-1727	SubEvidences: 1
PMID: 11717308	POS: 9020-945	SubEvidences: 2
PMID: 16166089	POS: 716-3383	SubEvidences: 2
PMID: 9642271	POS: 104912-2563	SubEvidences: 2
PMID: 15483315	POS: 18028-1584	SubEvidences: 1
PMID: 12186867		
PMID: 10944517		
PMID: 11973342		
PMID: 9915789		
PMID: 10644560	POS: 77689-20482	SubEvidences: 3
PMID: 15028726		
PMID: 15769847		
PMID: 10570223		
PMID: 10910976		
PMID: 11812794		
PMID: 11877404		
PMID: 9500786		
PMID: 10831608		
PMID: 11022045		
PMID: 14583632		
PMID: 10698171	POS: 62994-16378	SubEvidences: 2
PMID: 14607830		
PMID: 9545309		
PMID: 10725333		
PMID: 9417117		
PMID: 12082160		
PMID: 16103111		
PMID: 15078901		
PMID: 14593114		
PMID: 12754295		
PMID: 11381090		
PMID: 15252059		
PMID: 9679137		
PMID: 11673477		
PMID: 12711700		

#177: How do Bop-Pes interactions affect cell growth?

No answers found for this topic.

#178: How do interactions between insulin-like GFs and the insulin receptor affect skin biology?

PMID: 11181540	POS: 1492-2891	SubEvidences: 2
PMID: 9003010	POS: 27219-1714	SubEvidences: 8
PMID: 9582324	POS: 18468-729	SubEvidences: 3
PMID: 11423485	POS: 722-3379	SubEvidences: 2
PMID: 10465280	POS: 46737-19642	SubEvidences: 1

PMID: 11404219	POS: 68818-19098	SubEvidences: 1
PMID: 15576456	POS: 73189-30801	SubEvidences: 1
PMID: 12217880	POS: 7031-1929	SubEvidences: 1
PMID: 11796507	POS: 64194-19016	SubEvidences: 1
PMID: 14514645	POS: 53231-18645	SubEvidences: 1
PMID: 15705592		
PMID: 9048589	POS: 70921-22535	SubEvidences: 1
PMID: 10379878	POS: 29043-19499	SubEvidences: 1
PMID: 10698198	POS: 60854-23506	SubEvidences: 1
PMID: 12399424	POS: 61235-20407	SubEvidences: 1
PMID: 15192040	POS: 67271-24714	SubEvidences: 1
PMID: 10067859	POS: 71445-26717	SubEvidences: 1
PMID: 11956170	POS: 63605-28588	SubEvidences: 1
PMID: 12554765	POS: 79209-26935	SubEvidences: 1
PMID: 15070850	POS: 75648-27846	SubEvidences: 1
PMID: 15701676	POS: 52423-33787	SubEvidences: 1
PMID: 16249442	POS: 17867-3464	SubEvidences: 1
PMID: 12620890	POS: 9216-2763	SubEvidences: 1
PMID: 11272134	POS: 81922-30474	SubEvidences: 1
PMID: 8977385		
PMID: 11440896	POS: 48915-8946	SubEvidences: 1
PMID: 12538623	POS: 43735-13455	SubEvidences: 1
PMID: 10084556	POS: 55178-39053	SubEvidences: 1
PMID: 9357792		
PMID: 11726660		
PMID: 9745395		
PMID: 11289049		
PMID: 14970008		
PMID: 10826997	POS: 62775-47421	SubEvidences: 1
PMID: 11574665		
PMID: 11724898		
PMID: 11147784		
PMID: 15247278		
PMID: 15604210		
PMID: 9398687		
PMID: 9435535		
PMID: 10599698		
PMID: 11756245		
PMID: 11821295		
PMID: 15533996		
PMID: 9421397		SubEvidences: 1
PMID: 15919742		
PMID: 10644537		
PMID: 10655455		
PMID: 10698186		

113 more hits found.

#179: How do interactions between HNF4 and COUP-TF1 suppress liver function?

PMID: 12626717	POS: 88676-567	SubEvidences: 2

PMID: 11121483	POS: 66946-575	SubEvidences: 2
PMID: 11222753	POS: 57634-569	SubEvidences: 2
PMID: 11522818	POS: 92517-570	SubEvidences: 2
PMID: 11376119	POS: 79101-17755	SubEvidences: 1
PMID: 9547266		
PMID: 9658405	POS: 66785-23373	SubEvidences: 2
PMID: 11934848		
PMID: 9580705		
PMID: 12086970	POS: 34009-9692	SubEvidences: 1
PMID: 9461615		
PMID: 9421506		
PMID: 12522137		
PMID: 12105231		
PMID: 10652338		
PMID: 15855320		
PMID: 10070050	POS: 72458-17063	SubEvidences: 2
PMID: 11981036	POS: 94041-33011	SubEvidences: 2
PMID: 10887110		
PMID: 12040019	POS: 122817-35063	SubEvidences: 1
PMID: 15831710	POS: 62778-18091	SubEvidences: 1
PMID: 9171235		
PMID: 15928313	POS: 91981-29511	SubEvidences: 2
PMID: 10330009	POS: 113511-26019	SubEvidences: 2
PMID: 9454748		
PMID: 9139807	POS: 62396-21764	SubEvidences: 2
PMID: 14988251	POS: 46210-31102	SubEvidences: 1
PMID: 9886825	POS: 68661-22387	SubEvidences: 1
PMID: 16051671	POS: 121461-40974	SubEvidences: 4
PMID: 11861507		
PMID: 7768950		
PMID: 10893424		
PMID: 11574686		
PMID: 9651383		
PMID: 7559437		
PMID: 8995295		
PMID: 11929845		

#180: How do Ret-GDNF interactions affect liver development?

PMID: 9732293	POS: 65034-22533	SubEvidences: 1
PMID: 9786961	POS: 84018-46651	SubEvidences: 1
PMID: 11546745		
PMID: 12682085		
PMID: 12414989		

#181: How do mutations in the Huntingtin gene affect Huntington’s disease?

PMID: 15829505	POS: 97013-601	SubEvidences: 3
PMID: 15911879	POS: 89855-488	SubEvidences: 3
PMID: 12925571	POS: 62300-606	SubEvidences: 2

```
PMID: 15155855    POS: 65008-545     SubEvidences: 5
PMID: 15198993    POS: 77774-555     SubEvidences: 4
PMID: 15576360    POS: 126463-577    SubEvidences: 2
PMID: 11030759    POS: 0-336         SubEvidences: 4
PMID: 11285250    POS: 78015-488     SubEvidences: 2
PMID: 11487572    POS: 88883-488     SubEvidences: 3
PMID: 11673400    POS: 55563-488     SubEvidences: 2
PMID: 11912178    POS: 64451-515     SubEvidences: 2
PMID: 12384601    POS: 114767-520    SubEvidences: 3
PMID: 12490531    POS: 72977-519     SubEvidences: 2
PMID: 12554681    POS: 61592-517     SubEvidences: 2
PMID: 12952864    POS: 53042-522     SubEvidences: 2
PMID: 14654691    POS: 53510-520     SubEvidences: 2
PMID: 15494455    POS: 62159-540     SubEvidences: 2
PMID: 12058016    POS: 4966-2352     SubEvidences: 1
PMID: 9328463     POS: 26933-642     SubEvidences: 2
PMID: 9580659     POS: 63950-648     SubEvidences: 2
PMID: 10196373    POS: 2298-590      SubEvidences: 2
PMID: 10332038    POS: 43690-606     SubEvidences: 2
PMID: 11063736    POS: 53393-540     SubEvidences: 4
PMID: 14982954    POS: 89893-573     SubEvidences: 3
PMID: 9931325     POS: 5940-868      SubEvidences: 2
PMID: 14962977    POS: 4415-1605     SubEvidences: 1
PMID: 9158152     POS: 1580-1330     SubEvidences: 1
PMID: 9361024     POS: 3608-968      SubEvidences: 2
PMID: 10325409    POS: 1783-1144     SubEvidences: 1
PMID: 10769019    POS: 5739-749      SubEvidences: 1
PMID: 9536082     POS: 2040-891      SubEvidences: 2
PMID: 10958656    POS: 4205-2173     SubEvidences: 1
PMID: 15983033    POS: 4178-2078     SubEvidences: 1
PMID: 15964845    POS: 93874-1160    SubEvidences: 1
PMID: 8824873     POS: 1724-1373     SubEvidences: 2
PMID: 9147654     POS: 2457-1810     SubEvidences: 1
PMID: 9466992     POS: 1917-2270     SubEvidences: 1
PMID: 11152658    POS: 7627-965      SubEvidences: 1
PMID: 11092756    POS: 3536-2159     SubEvidences: 1
PMID: 12393793    POS: 4904-1884     SubEvidences: 2
PMID: 9694864     POS: 52759-877     SubEvidences: 1
PMID: 9887328     POS: 2551-727      SubEvidences: 2
PMID: 12620967    POS: 6144-2189     SubEvidences: 1
PMID: 15843398    POS: 4233-3208     SubEvidences: 2
PMID: 11689489    POS: 3773-2656     SubEvidences: 2
PMID: 11152661    POS: 3710-2509     SubEvidences: 1
PMID: 10655548    POS: 5256-2562     SubEvidences: 1
PMID: 9300654     POS: 1967-1393     SubEvidences: 1
PMID: 9535906     POS: 6228-1567     SubEvidences: 1
PMID: 8789437     POS: 26990-1598    SubEvidences: 1
```

360 more hits found.

#182: How do mutations in Sonic Hedgehog genes affect developmental disorders?

PMID: 15843416

#183: How do mutations in the NM23 gene affect tracheal development?

No answers found for this topic.

#184: How do mutations in the Pes gene affect cell growth?

No answers found for this topic.

#185: How do mutations in the hypocretin receptor 2 gene affect narcolepsy?

PMID: 12419707	POS: 141584-650	SubEvidences: 6
PMID: 11796702	POS: 35617-1939	SubEvidences: 2
PMID: 15310763	POS: 8915-1632	SubEvidences: 1
PMID: 15271651		
PMID: 15746258		
PMID: 11147774		
PMID: 11459774		
PMID: 12485808		
PMID: 15172887		
PMID: 15687100		
PMID: 15961555		
PMID: 12639903		
PMID: 14656716		

#186: How do mutations in the Presenilin-1 gene affect Alzheimer's disease?

PMID: 10401002	POS: 0-158	SubEvidences: 11
PMID: 14506131	POS: 72190-500	SubEvidences: 4
PMID: 10369872	POS: 28771-469	SubEvidences: 4
PMID: 10607841	POS: 57977-533	SubEvidences: 2
PMID: 16141195	POS: 129453-631	SubEvidences: 2
PMID: 11063718	POS: 75226-615	SubEvidences: 2
PMID: 16079288	POS: 123554-440	SubEvidences: 2
PMID: 14645205	POS: 113889-577	SubEvidences: 2
PMID: 12493737	POS: 0-914	SubEvidences: 4
PMID: 10748035	POS: 85819-1326	SubEvidences: 1
PMID: 9384602	POS: 62893-441	SubEvidences: 2
PMID: 11001931	POS: 65694-494	SubEvidences: 2
PMID: 10075646	POS: 2678-3784	SubEvidences: 1
PMID: 15115757	POS: 1104-2586	SubEvidences: 1
PMID: 11823322	POS: 3106-1341	SubEvidences: 1
PMID: 11278808	POS: 7552-2176	SubEvidences: 1
PMID: 12556527	POS: 2982-2895	SubEvidences: 1
PMID: 14970196	POS: 78757-1470	SubEvidences: 1
PMID: 10899157	POS: 2914-3369	SubEvidences: 2
PMID: 9575187	POS: 8052-2102	SubEvidences: 2
PMID: 12431992	POS: 6049-2107	SubEvidences: 1

PMID: 11084029	POS: 3374-4529	SubEvidences: 1
PMID: 15961413	POS: 5102-6949	SubEvidences: 1
PMID: 8817335		
PMID: 9334350	POS: 83748-24943	SubEvidences: 1
PMID: 15210705		
PMID: 15485850		
PMID: 9536100		
PMID: 15286082		
PMID: 9668120		
PMID: 11487570		
PMID: 10366599	POS: 109833-32264	SubEvidences: 1
PMID: 9065468		
PMID: 10085142		
PMID: 10652302		
PMID: 11134059		
PMID: 9452432		
PMID: 10816583		
PMID: 9575200		
PMID: 11581107		
PMID: 12119298		
PMID: 15123598		
PMID: 10811821		
PMID: 10748144		
PMID: 15051718		
PMID: 11912199		
PMID: 15615772		
PMID: 12551931		
PMID: 12771124		
PMID: 11983636		

120 more hits found.

#187: How do mutations in familial hemiplegic migraine type 1 (FHM1) gene affect calcium ion influx in hippocampal neurons?

No answers found for this topic.