GoPubMed: Ontology-based literature search for the life sciences

to receive the academic degree / zum Erreichen des akademischen Grades
Doktor-Ingenieur (Dr.-Ing.)

by / von

Andreas Doms
born on the 1st of July 1976 in Lauchhammer

submitted to / vorgelegt an der
Technical University of Dresden
Department of Computer Science

submitted: 13th of October 2008

Supervisor: Prof. Dr. Michael Schroeder
Referee: Prof. Dr. Uwe Aßmann
External referee: Prof. Dr. Patrick Lambrix
Contents

1 Motivation ................................................................. 1
   1.1 Definition of open problems ........................................ 1
       1.1.1 Open problem 1: Algorithms for the identification of ontological concepts in literature abstracts .............................. 1
       1.1.2 Open problem 2: Methods for knowledge mining in annotated literature databases .................................................. 2
       1.1.3 Open problem 3: Design of a system for ontology-based literature search for the life sciences ..................................... 3
   1.2 Overview .............................................................. 4

2 Background .............................................................. 5
   2.1 Introduction .......................................................... 6
       2.1.1 Recent advances .................................................. 6
       2.1.2 Motivation ......................................................... 8
   2.2 Terminologies and Ontologies in the life sciences .................... 9
       2.2.1 Lexical Resources ................................................ 9
       2.2.2 Terminological Resources ...................................... 10
           The UMLS Metathesaurus ............................................ 11
           Medical Subject Headings ........................................ 11
           Gene Ontology ...................................................... 12
       2.2.3 Ontologies ........................................................ 17
           Upper level ontologies ............................................. 17
           Domain ontologies .................................................. 18
           Formal ontologies .................................................. 18
   2.3 Semantic Web ........................................................ 18
   2.4 Biomedical Search engines .......................................... 20
       2.4.1 Search engines focusing on Information Retrieval .............. 20
           Improved querying .................................................. 20
           Results processing ................................................ 21
       2.4.2 Search engines focusing on Knowledge Retrieval ................ 23
           Tools integration ................................................... 23
           Semantic processing .............................................. 27
       2.4.3 General search engines ......................................... 35
   2.5 Ontology-based tools in the life sciences ........................... 36
       2.5.1 Aligning and Merging Ontologies ................................ 36
       2.5.2 Search in biomedical data ...................................... 36
       2.5.3 Data exchange ................................................... 36
       2.5.4 Information integration ....................................... 37
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.5.5</td>
<td>Knowledge representation</td>
<td>37</td>
</tr>
<tr>
<td>2.5.6</td>
<td>Computer reasoning</td>
<td>37</td>
</tr>
<tr>
<td>2.5.7</td>
<td>Textmining and Natural Language Processing</td>
<td>38</td>
</tr>
<tr>
<td>2.6</td>
<td>Biomedical text corpora</td>
<td>38</td>
</tr>
<tr>
<td>2.6.1</td>
<td>Literature/Citation databases</td>
<td>38</td>
</tr>
<tr>
<td>2.6.2</td>
<td>Full text corpora</td>
<td>40</td>
</tr>
<tr>
<td>2.6.3</td>
<td>Annotation corpora</td>
<td>41</td>
</tr>
<tr>
<td>2.6.4</td>
<td>Database curation projects</td>
<td>47</td>
</tr>
<tr>
<td>2.7</td>
<td>Biomedical textmining methodologies</td>
<td>52</td>
</tr>
<tr>
<td>2.7.1</td>
<td>Lexical methods</td>
<td>52</td>
</tr>
<tr>
<td>2.7.2</td>
<td>Syntactical methods</td>
<td>55</td>
</tr>
<tr>
<td>2.7.3</td>
<td>Semantic methods</td>
<td>56</td>
</tr>
<tr>
<td>2.7.4</td>
<td>Software Resources</td>
<td>58</td>
</tr>
<tr>
<td>2.8</td>
<td>Terminology recognition in the life sciences</td>
<td>58</td>
</tr>
<tr>
<td>2.8.1</td>
<td>Problem definition for Terminology Recognition</td>
<td>59</td>
</tr>
<tr>
<td>2.8.2</td>
<td>Terminology recognition methods</td>
<td>60</td>
</tr>
<tr>
<td></td>
<td>Dictionary-based</td>
<td>60</td>
</tr>
<tr>
<td></td>
<td>Rule-based</td>
<td>60</td>
</tr>
<tr>
<td></td>
<td>Alignment-based</td>
<td>61</td>
</tr>
<tr>
<td></td>
<td>Machine-learning</td>
<td>61</td>
</tr>
<tr>
<td>2.8.3</td>
<td>Word Sense Disambiguation</td>
<td>63</td>
</tr>
<tr>
<td>2.8.4</td>
<td>State-of-the-art in Concept Recognition</td>
<td>64</td>
</tr>
<tr>
<td>2.9</td>
<td>Evaluation Methodologies</td>
<td>69</td>
</tr>
<tr>
<td>2.9.1</td>
<td>Definitions of quality measures</td>
<td>69</td>
</tr>
<tr>
<td>2.9.2</td>
<td>Difficulties of corpus-based evaluations</td>
<td>71</td>
</tr>
<tr>
<td>2.9.3</td>
<td>Back-To-Back Tests</td>
<td>73</td>
</tr>
<tr>
<td>2.9.4</td>
<td>BioCreAtIvE</td>
<td>73</td>
</tr>
<tr>
<td>2.9.5</td>
<td>GOA database</td>
<td>74</td>
</tr>
<tr>
<td>2.9.6</td>
<td>TREC Genomics Track</td>
<td>74</td>
</tr>
<tr>
<td>2.9.7</td>
<td>Author curations</td>
<td>74</td>
</tr>
<tr>
<td>2.10</td>
<td>Outline</td>
<td>75</td>
</tr>
<tr>
<td>3</td>
<td>Algorithms for Concept Recognition</td>
<td>77</td>
</tr>
<tr>
<td>3.1</td>
<td>Introduction</td>
<td>79</td>
</tr>
<tr>
<td>3.2</td>
<td>Methods</td>
<td>80</td>
</tr>
<tr>
<td>3.2.1</td>
<td>Processing of the terminology</td>
<td>80</td>
</tr>
<tr>
<td></td>
<td>Concept labels</td>
<td>81</td>
</tr>
<tr>
<td></td>
<td>Information values</td>
<td>81</td>
</tr>
<tr>
<td></td>
<td>Morphological variants</td>
<td>84</td>
</tr>
<tr>
<td></td>
<td>Lexical variants</td>
<td>85</td>
</tr>
<tr>
<td></td>
<td>Syntax analysis</td>
<td>86</td>
</tr>
<tr>
<td>3.2.2</td>
<td>Candidate identification</td>
<td>87</td>
</tr>
<tr>
<td></td>
<td>Candidate Generation</td>
<td>87</td>
</tr>
<tr>
<td></td>
<td>Candidate Ranking</td>
<td>89</td>
</tr>
<tr>
<td>3.2.3</td>
<td>Disambiguation Methods</td>
<td>89</td>
</tr>
<tr>
<td></td>
<td>Algorithms for Word Sense Disambiguation</td>
<td>90</td>
</tr>
<tr>
<td></td>
<td>Experimental setup</td>
<td>96</td>
</tr>
<tr>
<td></td>
<td>Experimental Results</td>
<td>99</td>
</tr>
<tr>
<td></td>
<td>Discussion of methods</td>
<td>100</td>
</tr>
<tr>
<td>3.2.4</td>
<td>Context Models for Concept Disambiguation</td>
<td>102</td>
</tr>
</tbody>
</table>
Walkthrough of recognition pipeline ........................................ 102
Using Context Models to predict relevant articles ......................... 103
3.3 Evaluation of Concept Recognition algorithms .......................... 106
  3.3.1 BioCreAtIvE Gold Standard ........................................ 106
  3.3.2 Author Curations Benchmark ..................................... 108
3.4 Summary and Discussion ............................................. 113

4 Ontology-based Literature Search ........................................ 117
  4.1 Introduction .................................................................. 119
  4.2 Methods ..................................................................... 120
    4.2.1 Query expansion with terminologies ............................. 121
    4.2.2 Structuring documents ........................................... 122
    4.2.3 Annotating documents ............................................ 125
    4.2.4 Ontology Induction ................................................ 128
    4.2.5 Interpreting Induction Results .................................... 131
      Informative paths ....................................................... 131
      Informative concepts ................................................ 134
    4.2.6 Aspect ranking ..................................................... 140
    4.2.7 Question answering ................................................ 140
    4.2.8 Bibliometric analysis .............................................. 143
  4.3 Evaluation of Ontology-based Literature Search with GoPubMed .......... 143
    4.3.1 Answering TREC Question with GoPubMed ...................... 144
    4.3.2 Ontology-based bibliometric analyses with GoPubMed ........... 149
      Important topic: Apoptosis ............................................ 150
      Important topic: endosome .......................................... 151
      Important place: Dresden ............................................. 152
      Important journal: Which are the 10 most frequently used GO terms in Nature, Cell and Science? .......................... 152
      20 Journals for the Molecular Biologist ................................ 153
      Summary of analysis .................................................. 155
  4.4 Summary and Discussion ............................................. 156
    4.4.1 Induced Ontology .................................................. 156
    4.4.2 Question Answering ............................................... 158
    4.4.3 Bibliometric Statistics ............................................ 158

5 Design of GoPubMed ...................................................... 161
  5.1 Introduction .................................................................. 162
  5.2 Requirements Analysis ................................................ 164
    5.2.1 Stakeholders ......................................................... 164
    5.2.2 Domain Analysis .................................................... 166
    5.2.3 Problem description ............................................... 166
    5.2.4 Goals ................................................................. 167
    5.2.5 User requirements .................................................. 168
      Functional Requirements ............................................. 168
      Quality Requirements ................................................ 169
      External Requirements .............................................. 170
    5.2.6 Acceptance .......................................................... 171
    5.2.7 Use case ............................................................. 171
    5.2.8 GUI Design ........................................................ 172
  5.3 Architectural Design ................................................... 174
5.4 Version history of GoPubMed ............................................. 178
  5.4.1 Community curation effort ........................................ 178
  5.4.2 User acceptance ..................................................... 178
  5.4.3 Reuse of component frameworks ................................. 181
5.5 GoPubMed’s offsprings .................................................. 181
5.6 Summary and Discussion ................................................ 190

6 Conclusion and future work ................................. 193
  6.1 General contributions ............................................... 193
  6.2 Contributions to open problem 1 .................................... 193
  6.3 Contributions to open problem 2 .................................... 194
  6.4 Contributions to open problem 3 .................................... 195
  6.5 Future research ........................................................ 195
    6.5.1 Ontology generation .............................................. 195
    6.5.2 Question answering ............................................... 195
    6.5.3 GoEverywhere ...................................................... 196
List of Figures

2.1 Platforms for proteomics and functional genomics. Methodology is shown in the outer columns, resultant data sets in the middle columns, and model systems in the center. Figure reproduced from Tyers and Mann [269].

2.2 In blue, the number of citations in PubMed grows exponentially, hitting 754,003 new citations in 2007. The yellow line shows the number of review articles in PubMed. PubMed has published usage statistics since 1997 and the numbers have steadily increased by 97 million per year on average. The figure currently stands at 871 million per year.

2.3 The Gene Ontology term lineage beginning from term “peptidyl-prolyl cis-trans isomerase activity” to the root node. The the solid arrows denote direct is.a relations. The dotted arrows denote transitive is.a relations of the term “peptidyl-prolyl cis-trans isomerase activity”.

2.4 (a) A subsumption hierarchy violating the True Path Rule: Here a fly chitin synthase could be annotated as a CHITIN BIOSYNTHESIS, and appear in a query for genes annotated to CELL WALL BIOSYNTHESIS, which makes no sense because flies don’t have cell walls. The dashed nodes are replaced in the revised version of the GO branch shown in (b).

(b) The revised ontology structure ensures the True Path Rule. The parent chitin metabolic process now has the child terms CUTICLE CHITIN METABOLIC PROCESS and CELL WALL CHITIN METABOLIC PROCESS, with the appropriate CATABOLIC PROCESS and BIOSYNTHETIC PROCESS terms beneath them, as dashed nodes.

2.5 The diagram shows the transitions from data to information, to knowledge, and finally to wisdom. Understanding is not a separate level of its own, it is necessary for the transition from each stage to the next. Figure adopted from [19]. Ackoff [1] points out that information ages quickly, knowledge has a longer time span, only wisdom is permanent.

2.6 AliBaba: What are the risk factors of treating G6PD-deficient malaria patients with primaquine? The query entered in Ali Baba was “primaquine malaria g6pd deficiency”. The graph shows the connection between G6PD (deficiency), vivax malaria (patients), and their treatment with primaquine. The risk factors hemolytic anemia/hemolysis are directly connected to G6PD and primaquine. Other nodes in the graph present more information on studied populations, the viral origin, infected cells, and information on the disease. Example from AliBaba’s website.

2.7 Chilibot: Relations visualization in Chilibot. A new hypothesis is shown that CREB interacts indirectly with synaptotagmin via one or more of 22 other genes. This hypothesis is discussed in Plake et al. [204].
LIST OF FIGURES

2.8 An example of a DocBook document. .......................... 43
2.9 A fragment of a GENIA document encoded with TEI. ................. 43
2.10 A sentence with treebanking information. Sentences (S), noun phrases (NP), verb phrases (VP) ................................. 45
2.11 The GENIA ontology (Version 3.01) is a concept tree. The GENIA corpus is only annotated with the leaf concepts printed bold in this figure. ............... 46
2.12 An XML fragment of a structural template for PMID: 9582366 annotated manually for FetchProt. The experiment describes a non-membrane spanning protein tyrosine kinase activity (GO:0004715) .................. 47
2.13 (a) linear case, the hyperplane is a line between the 2D-vectors (b) non-linear case, a Kernel function is needed .................. 62
2.14 Interpolated term extraction precisions for one article. The dotted line marks the interpolated values. .............................. 71

3.1 The table shows the distribution of the word counts per GO concepts. Most GO labels have 2-4 words. Very short labels such are rare and tend to be ambiguous. Very long labels encoding additional information in a subclause are a peculiarity of GO. ........................................ 81
3.2 This plot shows the distribution of GO words in the Gene Ontology. The distribution follows Zipf’s law. A few words occur very frequent while many terms occur rarely. Words with a very high frequency do not give much information as they are part of many labels in the ontology. Highly frequent words have a low information value, low frequent words have a high information value. 82
3.3 Output of the ElivagarConsole program to pre-process the Gene Ontology vocabulary. Line 1 prompts the successful initialization of some taggers. Line 2 prompts the registering of the revisions. Java’s reflections mechanism is used to detect revision plugins. Line 9 executes an external script to load the 57173 labels of the Gene Ontology and run some revisions on them. In line 13 the GO specific trailers like “sensu ...” are removed and stored for later use. Line 14 executes the computation of global word information values. Line 16 executes revisions filtering invalid inflections of words based to the previously assign syntactic categories. Line 25 executes revisions adding lexical variants to word recognized as Roman and Arabic numbers. Line 28 marks mandatory word in each label. Line 30 saves the revised labels into a file. ............... 84
3.4 This algorithm produces only concept label candidates which have a textual token for the required words. For example the term LYTIC VIRUS BUDDING FROM PLASMA MEMBRANE is not selected as a candidate if only inflected forms of the words “from” and “membrane” are detected. The two words combine an information value of less than 16% of the whole label. ............... 88
3.5 The figure shows the path between the UMLS terms BODY PART ORGAN OR ORGAN COMPONENT and AMINO ACID PEPTIDE OR PROTEIN. The edges describe relations between entities (in our case, the subtype-aware-signature and its sub-properties) and nodes consist of classes and relations of the ontology. BODY PART ORGAN OR ORGAN COMPONENT is a subsumption of FULLY FORMED ANATOMICAL STRUCTURE, which belongs to the signature of the relation produces. This relation has as range ORGANIC CHEMICAL which is a super-class of AMINO ACID PEPTIDE OR PROTEIN. The length of this path is 4. ........................................ 92
LIST OF FIGURES

3.6 *Thrush* is an ambiguous term, as its senses include *songbird* or *oral candidiasis*. This figure shows the possibilities for disambiguating ‘thrush’. Solid edges are *is a* relationships.

3.7 The figure shows the measured performance on the BioCreAtIvE corpus for different threshold parameter values. The optimal threshold parameter for Context Model algorithm is between 0.80 and 0.84.

3.8 The screenshot shows the online Curation Tool integrated in the biomedical search engine GoPubMed. The authors can judge the quality of automatic annotations by the system and add missing concepts of MeSH and the Gene Ontology. The picture shows six curated terms for this article in the list on the right side.

3.9 The disambiguation performance for the concept *development* decreases with less training examples reaching a plateau at 100 examples.

3.10 Schematic view of induced concept sets: The outer triangle denotes the whole ontology. Top corner is the root concept *r*. Concept annotations *a*<sub>1</sub> and *a*<sub>2</sub> induce concepts included in all paths up to the root concept, visualized by the corresponding rectangles. The overlapping parts of manual curation induced by *c* and annotations denote the set of induced true positive matches (iTP). Annotations not covered by induced manual curation denote the set of false positive matches (iFP). Unannotated manual curation denote the set of induced false negatives (iFN).

3.11 Recognition performance of the Local Term Alignment algorithm on the author curations benchmark. The best performance is measured at a threshold level of 0.74. The induced precision is generally higher than the standard precision. The reason is that missing ancestors of manual curations are not counted as false positives matches. Also the induced recall is higher than the standard recall. The reason is general concepts with a low confidence are less likely to be counted as false negative matches if specific descendant labels with higher confidence are mentioned.

3.12 The figure shows the performance of the Concept Recognition pipeline on author curated PubMed abstracts. The best performance is achieved at a confidence threshold level of 0.80. The induced precision always higher than the standard precision. Induced recall does not much differ from the standard measure. The recall decreases dramatically for threshold level above 0.96.

3.13 The figure shows the average precision at eleven standard recall levels for the concept recognition using various revision of the concept labels. The lowest performance was achieved when weighting all label words equally important. Using the global word information value compares with the baseline algorithm the Local Term Alignment. The stem revisions remove inflection variants often leading to false positive candidates. Finally the the context models help improving precision at the expense of recall.

4.1 Text is represented as a tree structure. Each node represents a non-overlapping sub-string of the text. This is an example of an abstract segmented into three sentences containing word tokens.

4.2 Size of the result set returned for PubMed queries send to GoPubMed during the last two years.

4.3 Size of the induced ontology for 7 queries at different sampling sizes.
4.4 This figure shows the induced ontology for the PubMed result to the query "rab5". The top level MeSH and GO categories are alphabetically ordered. The GO branch CELLULAR PROCESS is expanded to the concept ENDOCYTOSIS, the topic process most intensively for the protein. The induced ontology shows 16 other biological processes related to the protein rab5. The numbers denote the amount of the linked documents.

4.5 The richest path for the query "ribosome". The symbol "≫" denotes an is-a relation and ∈ denotes a part-a relation. A ribosome can be characterized as a non-membrane-bound intracellular organelle which is an intracellular part of the cell.

4.6 This figure shows the quality of the concept ranking for the topic "pancreas cancer prognosis" at different ranks for the four weighting functions: cf-idf, icf-idf, cf-iiidf and icf-iiidf. Local Induced - Global Count (icf-idf) performs worst compared to all other rankings. Local Induced - Global Induced slightly dominates at high ranks.

4.7 This plot shows the average quality for four ranking functions on 20 queries. The function icf−idf performs worst, icf−iiidf has slightly lower performance that the other to weighting functions. The other two, not inducing locally, rank the importance of concepts for a topic equally good.

4.8 The prototype can answer "Which" questions by decomposing the natural language questions and using the background knowledge of MeSH and GO to present relevant answers in the form of aspects. Each list item is a concept defined in the background knowledge and groups textual passages from PubMed abstracts which mention the keywords and the concept together.

4.9 This screenshot shows the induced ontology for the query "PrnP" in GoPubMed. ENCEPHALOPATHY, BOVINE SPONGIFORM, the official name for "Mad Cow Disease" is selected. It links to 77 publications in PubMed. Additionally SCRAPIE, CREUTZFELDT-JAKOB SYNDROME and other diseases are listed in the disease branch of MeSH.

4.10 This screenshot shows the first text snippet the user sees when clicking on ENCEPHALOPATHY, BOVINE SPONGIFORM. The keyword "PrP" is a synonym of "PrnP" and is highlighted in yellow. The disease’s synonym, BOVINE SPONGIFORM ENCEPHALOPATHY, is highlighted in green. The snippet is ranked highest because it is a recent publication plus keyword and concept is mentioned in the same sentence.

4.11 A full abstract shown in the document view of GoPubMed for the query "BRCA1 AND BARD1". The abstract is shown when selecting concept DNA REPAIR. Note that DNA repair is not mentioned literally in the text. Instead the Concept Recognition algorithm detected INTERSTRAND CROSSLINK REPAIR as a synonym of NUCLEOTIDE-EXCISION REPAIR which is a descendant of DNA REPAIR.

4.12 PubMed abstracts per year mentioning apoptosis (including synonyms and specialisations of the term).

4.14 Topic evolution for the most mention GO concepts since 1972. The list of GO concepts is ordered by the research interest over time. Topics like apoptosis, transduction, donor preference, cell proliferation and necrosis are of increasing research interest whereas liver development, pregnancy and kidney development show an relatively decreasing rate of mentions in PubMed abstracts. .......................................................... 154

5.1 Three search paradigms ................................................................. 164

5.2 The GO sub-hierarchy containing terms related to small GTPases. ▲ symbolizes an is_a relation and □ symbolizes a part_of relation. Nodes marked △ hide more children related to "small GTPase mediated signal transduction". Note: this tree view is stripped down to the concepts of GO necessary to explain our example. The subtree related to regulation of Rho protein signal transductions present two times because this GO term has multiple parents. The relations in GO are graph-shaped, we show here a simplified hierarchical representation. ....................................................... 172

5.3 Four experimental GUI design for GoPubMed. (a) This design was used in the first prototype deployed in 2004, see figure 5.7 for a screenshot. A problem is the used space in the header frame. (b) This design variant includes the input field in the document view. The disadvantage is that the input area moves when the document view is scrolled. (c) This layout is used to the current version of GoPubMed, see screenshot 5.13. Less space is unused in the header and the input area is always visible when documents are scrolled. The disadvantage is that the frame architecture makes communication between the parts of the GUI complex. Another problem is that the frame borders disallow to display information across the borders. e.g. drag and drop of information or and auto-completion box opening in the input frame can not expand across the frameborder. (d) A frameless layout. A draw back is that the ontology view moves when document are scrolled. This can be avoided with a more complex dynamic HTML interface. AJAX technology is required to update parts of the interface independently. .............................................. 173

5.4 This sequence diagram shows the default use case in which a user searches with a keyword query. The result contains cached, previously annotated and new documents. The outcome is a DocumentResultSet which is an iterator over AnnotatedDocuments. .............................................................. 176

5.5 Static Model of the presentation layer of GoPubMed 3.0 .................... 177

5.6 Timeline of GoPubMed releases, foundation of spin-off company and first customers for the ontology-based search platform ......................... 179

5.7 The screenshot shows the search result for authors named “Pizauro”. The result show 26 documents classified using the Gene Ontology. The concept soluble was selected and one abstract is displayed. Ten other Gene Ontology concepts were identified using the Local Term Alignment methode developed in Doms [67]. ....................................................... 180
5.8 The second release of GoPubMed (internal version 3.0) beginning of 2006 included the restructuring of the architecture which provided the basis for the next two versions of the software. The Graphical User Interface was redesigned allowing result sets of size 500-1000 instead of 100 in the first version, static bibliometric pages for single ontology terms, highlighting of semantic links and keywords in the abstract texts and titles. The position of a citation in the original PubMed query was shown beside links to HubMed and Google Scholar. The concept of informative concepts was introduced as Frequent term for the current query. Other improvements included export functions such as BibTex, EndNote and FullText as well as searching in the ontology tree. 183

5.9 This figure shows the static bibliometric statistics provided by GoPubMed 3.0. The example shows the plotted research interest in ARACHIDONIC ACID METABOLISM. The upper plot shows the absolute numbers of citations mentioning the term or a descendant concept. The lower plot shows the relative research interest compared to all other topics covered by PubMed citation in the same year in percent. The red line shows the moving average over 5 years. 184

5.10 In GoPubMed 3.0 the Thompson Impact Factor was used to weight the individual publications of an author. The dynamic tables allowed to sort the rows according to the absolute number of articles of an author, the average impact factor of an authors publications, the maximum or minimum impact factor of an authors publication. One problem approached in a later version of GoPubMed were ambiguous author names. 185

5.11 Beginning of 2007 a release flagged with the internal version 3.0.8 included the following improvements: The list of Top Categories was computed based on the tf – idf ranking described in section 4.2.5, AJAX technologies were introduced to allow for dynamical loading of the ontology tree, beside Gene Ontology term now also Wikipedia categories and MeSH headings were annotated, filtering was now possible with MeSH and GO concepts. Citations in the document view were now re-ranked according to the proximity of keywords and recognized ontology concepts and a dynamic bibliometric analysis for the current query was introduced. This screenshot shows GoPubMed 3.0.8 for a search with the keywords "aspirin". 186

5.12 In GoPubMed 3.0.8 the static bibliometric statistics for each concept were extended with an author collaboration network based on the top 50 authors. The example shown here displays the result for the query "gene ontology". 187
5.13 In December 2007 a press release in 22 languages was published to announce the new release 3.6 of GoPubMed. Interesting new features included Social Networking Features for biomedical experts, the ability to query up to 10,000 documents, the grouping of the organized information into four answer sections. The What section contained the induced ontology. The Who section contained filters for the disambiguated authors identified in the citations. The Where section allowed for filtering the documents according to geographic parameters and journals. The When section allowed filtering on the temporal basis. An eye-catching new feature showed the author profile of the most prominent author for the current query. The profile in this example shows that the author shown here is an internationally leading author in Helicobacter pylori, Gastritis and other topics. Form the publications linked to the author profile the system estimated that the author was at least from 1987 to 2007 affiliated with the Digestive Disease Section of the Veterans Administration Medical Center in Houston, Texas. Authors were able to edit their profiles online. Until today 978 authors edited their profiles.

5.14 GoPubMed 3.6 introduced the Author Curation Tool which allowed users to give feedback on the automatic annotation quality of GoPubMed. Data collected with this tool was used to evaluate the recognition performance of the system and to train disambiguation models to improve the precision of the semantic markups.

6.1 A rule based segmentation algorithm was used to split texts into sentences. The input text is assumed to be tokenized before.
LIST OF FIGURES
List of Tables

2.1 A typical term lineage in the Gene Ontology reveals common substring relations between GO terms. From bottom to top these terms are all linked via the is_a relation. .................................................. 13
2.3 A part-of relation can have four different interpretations. The Gene Ontology uses only the interpretation 2 and 4. ................................. 14
2.5 Categorization of Biomedical Search Engines ................................. 20
2.7 This table compares biomedical search engines according to their features regarding the querying process. ................................. 24
2.9 This table compares biomedical search engines according to their features regarding the processing of the retrieved results. ................................. 25
2.11 This table compares biomedical search engines according to their features regarding the processing of the retrieved results. ................................. 25
2.13 This table compares biomedical search engines according to their features for linking results to external knowledge. ................................. 31
2.15 This table compares biomedical search engines according to their features for linking results to external knowledge. ................................. 33
2.16 Comparision of Biomedical Search Engines, descriptions on the following page 34
2.17 Medline fields .................................................. 39
2.19 Comparision of annotation corpora, part 1 ................................. 49
2.21 Comparision of annotation corpora, part 2 ................................. 50
2.22 The Penn Treebank Part-of-Speech set. .................................. 51
2.24 A typical GOA record. ............................................. 51
2.25 The part of speech tags for the sentence “Pockets of higher lead poisoning rates continue to be a problem in some geographic areas.” ................. 55
2.27 Noun phrases in the sentence “Pockets of higher lead poisoning rates continue to be a problem in some geographic areas.” ................. 56
2.28 Contingency matrix for the outcomes of a term identification system. This shows all possible outcomes of a system which has the task of identifying relevant entities in a text. ............................................. 69

3.1 Table of most frequent words in the Gene Ontology. “Activity” is the most often used word. GO appends the word “activity” to gene products to clarify that it refers to the attribute of the gene product and not to the product itself. 82
3.3 Label Revisions modifying word information values ............................. 83
3.5 Words found in PubMed which are stemmed to “acid” by the PorterStemmer. 85
3.6 Ambiguous terms and their senses in the WSD datasets collected. ....... 97

XV
3.7 The above datasets contain manually collected PubMed articles by one expert (high quality / low quantity), manually curated articles by a group of non-experts (medium quality / medium quantity) and semi-automatically collected articles (low quality / high quantity).

3.9 CS1 column contains the results (% $f$-measure) for the Closest Sense (CS) approach with the use of the classic distance (only subsumption). CS2 column contains the results for the CS approach with the use of the optimized signature together with the subsumption distance. TC1 and TC2 contain the results of the Term Cooc (TC) approach, when the co-occurrences or the inferred co-occurrences are used, respectively. TC3 contains the results for the TC approach with co-occurrences and support vector machines, and TC4 when inferred co-occurrences and SVMs are used. bME column contains the results for the baseline method (classical Maximum Entropy modeling of stems without meta-data or hierarchical information), trained and tested on the high quality corpus in a 5-fold cross validation. MD1 is for the Meta-Data approach, trained and tested on the high quality corpus in a 5-fold cross validation. MD2 is trained on the medium quality/quantity corpus and tested on the high quality one. MD3 was trained on the low quality / high quantity corpus and tested on the high quality corpus. Some terms (spindle, nucleus, transport) are easier to disambiguate than others (development, lead). Overall, all methods perform well between 73-96% $f$-measure.

3.10 Examples of binary features extracted from the PubMed abstract below. The word phrase features were stemmed using the Porter Stemmer algorithm. The journal title is used as it is. The years are accumulated to decades, so the example falls in to the decade 2000-2010.

3.12 GO concepts most often marked as false positive matches in the comparison with the BioCreAtIvE benchset. The curators of this corpus focused only on concepts explicitly relevant for the named protein in the texts. Other general concepts were not marked as GO concepts. This results in a low measured precision on this corpus.

3.13 List of 20 most often curated GO concepts in a time period of two month. The last column shows the disambiguation performance measured using a 5-fold cross validation for Context Models if available. Many of the concepts provide not yet enough training data for disambiguation. Some disambiguation hints are very uneven providing almost only negative or positive hints. The Maximum Entropy models require a reasonable balanced set of training examples. A strategy to improve the systems recognition performance is to focus the curation efforts of concepts with prevailing negative hints, e.g. the concept label nursing behavior. Such as miss-balance indicates systematic miss-classification of the concept. Concept labels with prevailing positive hints tend to be less ambiguous. For example the concept label apoptosis.

4.2 A selection of PubMed queries with and without the query expansion applied by the NCBI. The differences in the size of the result set are large in most cases. Only "alzheimer" and "aspirin" return the same numbers with and without expansion. Both headings have no expansions defined in the NCBI’s translation tables although MeSH knows 16 synonyms for Aspirin and 13 synonyms for Alzheimer Disease.
### LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.4</td>
<td>Same queries as in table 4.2 expanded with synonyms and descendant terms. The last column shows the size of the search result using all synonyms of the descendants and the concept itself.</td>
</tr>
<tr>
<td>4.6</td>
<td>Taggers implemented in the package Elivagar (see supplementary material). The third column names tag types required before application of the tagger.</td>
</tr>
<tr>
<td>4.7</td>
<td>This table shows 20 PubMed queries provided by domain experts. The experts ranked a selection of concepts according to their relevance to the topic. The selection was made using the union of the 40 top ranked concepts by the four different weighting schemes.</td>
</tr>
<tr>
<td>4.9</td>
<td>The ranking of concepts provided an expert for the PubMed query &quot;pancreas cancer prognosis&quot;.</td>
</tr>
<tr>
<td>4.10</td>
<td>Both weighting functions of concepts for the &quot;pancreas cancer prognosis&quot; provide similar rankings.</td>
</tr>
<tr>
<td>4.13</td>
<td>Statistics for endosome (Art. = number of articles containing endosome). Top left: Most prolific journals. Top right: Most prolific authors.</td>
</tr>
<tr>
<td>5.6</td>
<td>Feedback given in an online survey with PubMed authors.</td>
</tr>
<tr>
<td>6.1</td>
<td>Genome sequencing projects statistics, from NCBI's website</td>
</tr>
<tr>
<td>6.2</td>
<td>First level of MeSH Headings</td>
</tr>
<tr>
<td>6.3</td>
<td>OBO Foundry candidate ontologies</td>
</tr>
<tr>
<td>6.4</td>
<td>MeSH Subheadings are used to better define a topic, narrow retrieval, or express a certain aspect of a main MeSH heading.</td>
</tr>
<tr>
<td>6.5</td>
<td>Databases contributing gene product associations to the Gene Ontology Annotations project.</td>
</tr>
</tbody>
</table>
Abstract

Background: Most of our biomedical knowledge is only accessible through texts. The biomedical literature grows exponentially and PubMed comprises over 18,000,000 literature abstracts. Recently much effort has been put into the creation of biomedical ontologies which capture biomedical facts. The exploitation of ontologies to explore the scientific literature is a new area of research.

Motivation: When people search, they have questions in mind. Answering questions in a domain requires the knowledge of the terminology of that domain. Classical search engines do not provide background knowledge for the presentation of search results. Ontology annotated structured databases allow for data-mining. The hypothesis is that ontology annotated literature databases allow for text-mining. The central problem is to associate scientific publications with ontological concepts. This is a prerequisite for ontology-based literature search. The question then is how to answer biomedical questions using ontologies and a literature corpus. Finally the task is to automate bibliometric analyses on an corpus of scientific publications.

Approach: Recent joint efforts on automatically extracting information from free text showed that the applied methods are complementary. The idea is to employ the rich terminological and relational information stored in biomedical ontologies to markup biomedical text documents. Based on established semantic links between documents and ontology concepts the goal is to answer biomedical question on a corpus of documents. The entirely annotated literature corpus allows for the first time to automatically generate bibliometric analyses for ontological concepts, authors and institutions.

Results: This work includes a novel annotation framework for free texts with ontological concepts. The framework allows to generate recognition patterns rules from the terminological and relational information in an ontology. Maximum entropy models can be trained to distinguish the meaning of ambiguous concept labels. The framework was used to develop a annotation pipeline for PubMed abstracts with 27,863 Gene Ontology concepts. The evaluation of the recognition performance yielded a precision of 79.9% and a recall of 72.7% improving the previously used algorithm by 25.7% f-measure. The evaluation was done on a manually created (by the original authors) curation corpus of 689 PubMed abstracts with 18,356 curations of concepts. Methods to reason over large amounts of documents with ontologies were developed. The ability to answer questions with the online system was shown on a set of biomedical question of the TREC Genomics Track 2006 benchmark. This work includes the first ontology-based, large scale, online available, up-to-date bibliometric analysis for topics in molecular biology represented by GO concepts. The automatic bibliometric analysis is in line with existing, but often out-dated, manual analyses.

Outlook: A number of promising continuations starting from this work have been spun off. A freely available online search engine has a growing user community. A spin-off company was funded by the High-Tech Gründerfonds which commercializes the new ontology-based search paradigm. Several off-springs of GoPubMed including GoWeb (general web search), Go3R (search in replacement, reduction, refinement methods for animal experiments), GoGene (search in gene/protein databases) are developed.
Publications

Some ideas and figures have appeared previously in the following publications:

Peer Reviewed Publications

- **GoPubMed: Exploring PubMed with the GeneOntology.**
  Andreas Doms and Michael Schroeder.

- **Biomedical word sense disambiguation with ontologies and meta-data automation meets accuracy**
  Dimitra Alexopoulou, Bill Andreopoulos, Heiko Dietze, Andreas Doms, Fabien Gandon, Jörg Hakenberg, Khaled Khelif, Michael Schroeder and Thomas Wächter.

- **Facts from text**
  Rainer Winnenburg, Thomas Wächter, Conrad Plake, Andreas Doms, and Michael Schroeder.

- **Agents in bioinformatics, computational and systems biology.**

- **SCOWLP: A web-based database for detailed characterization and visualization of protein interfaces.**
  Joan Teyra, Andreas Doms, Michael Schroeder, and M. Teresa Pisabarro.

Conferences and Book Chapters

- **GoPubMed: ontology-based literature search applied to Gene Ontology and PubMed**

- **Ontologies and Text Mining as a Basis for a Semantic Web for the Life Sciences.**

- **GoPubMed: Exploring PubMed with Ontological Background Knowledge**
Workshops and Others

- **How to query the GeneOntology.**
  Andreas Doms, Tim Furche, Albert Burger, and Michael Schroeder.

- **GoPubMed: Answering biomedical questions**
  Andreas Doms.

- **Literature Search with Ontologies**
  Andreas Doms, Heiko Dietze, Thomas Wächter and Michael Schroeder.
  In proceedings of *Web4Web Workshop*, 2008
Chapter 1

Motivation

A major goal in the post-genome era is the exploration of the order and logic of genetic programs [35]. Advances in sequencing technology made genomes of many organisms available. High-throughput experiments create masses of data which can be mined for new insights into biological programs.

Despite the fact that ever more biomedical knowledge is stored in structured databases [269] including genome sequences, molecular structures, biological pathways, protein interactions and gene expression arrays most of the biomedical knowledge available nowadays is still only accessible through unstructured text in scientific publications.

1.1 Definition of open problems

The biomedical literature grows at a tremendous rate and PubMed comprises already over 18,000,000 abstracts. Finding relevant literature is an important and difficult problem with up to 5,000 new citations in PubMed every day. Biomedical text mining aims to manage this information blast [26].

Recently much research has been devoted to the analysis of the biomedical literature. This interest has been sparked by the growth in literature, but also by the availability of abstracts, full papers, and bibliometric data. Researchers have been specifically interested in automatically extracting information from free text such as protein names [91, 290, 124], ontology terms [247, 27, 64], and protein interactions [265, 90, 125]. These techniques are then applied to aid or even automate annotation of proteins.

Ontologies are increasingly used to capture biological knowledge. In Lambrix et al. [154] the authors give examples of biological ontologies and ontology-based knowledge. A prominent example of an ontology is the Gene Ontology (GO) [98], which provides a hierarchical vocabulary for function, processes and cellular locations. GO is used to annotate proteins in biological databases such as the sequence database UniProt [10] and the protein structure databank PDB [21]. In the context of these developments the following problems arise.

1.1.1 Open problem 1: Algorithms for the identification of ontological concepts in literature abstracts

There is no common vocabulary in the biomedical domain. Classical keyword search is not aware of inconsistent naming practices. Therefore classical search is often not yielding the best results and a lot of important information remains buried in the masses of text.
PubMed freely serves the abstracts of most biomedical publications since 1972. While full texts contain all information published in an article including tables, figures, methods used and related work the abstracts contain the keypoints of each work in a comprehensive form. In this work the freely accessible citation abstracts are used instead of the in general non-free full texts containing a lot of background information yielding potentially many irrelevant documents.

**Open problem 1: Scalable algorithms for ontological concept recognition**

The goal is to design an efficient method to annotate free PubMed abstracts with concept labels of the Gene Ontology. Out of a pool of several thousands ontological concepts the system has to detect candidates in a way that tolerates potentially new syntactical variants of known concepts aiming at a high recall. The meanings of ambiguous concept labels have to be distinguished aiming at a high precision. The algorithms have to scale for millions of documents and thousands of concepts.

1.1.2 Open problem 2: Methodes for knowledge mining in annotated literature databases

Semantic markups in unstructured text are a prerequisite for an ontology-based literature search. When people search, they have questions in mind. Answering questions in a domain requires the knowledge of the terminology of that domain. Classical search engines do not provide background knowledge for the presentation of search results.

Biological questions, like "Which anaerobic bacterial strains are resistant to Vancomycin?", as they are given in the TREC Genomics Track 2006 [118], can not easily be answered by classical search engines. Underlying all of the above textmining applications is the literature, which grows overall. But at a closer glance it turns out that some research areas shrink, while others take off. Bibliometric analyses aim to shed light on such developments and to identify emerging trends. Such analyses date back to the 60s [209] and typically focus on research topics [95], specific journals [31], or the researchers themselves [186, 187, 209, 110, 47, 34]. Garfield and Melino [95] investigate e.g. the research on programmed cell death. Despite programmed cell death being described 25 years ago, it took some 15 years until journals such as "Cell Death and Differentiation" emerged and the number of publications in the field in general took off. As an example for an analysis of a specific journal Boyack analysed the emergence and development of topics covered by PNAS [31]. A very active area of research aims to understand the social process of publishing by investigating co-author and co-citation networks [186, 187, 209, 110, 47, 34]. Such analyses allow one to identify authors in an organisation, who work interdisciplinary and connect otherwise unconnected co-author networks [187], to animate citations of key publications over time [47], evolution of author and publication networks [34], and to understand how groups form [110]. All of these analyses are useful to take a birds-eye view onto research.
1.1. DEFINITION OF OPEN PROBLEMS

1.1.2 Open problem 2: Methodes for knowledge mining in annotated literature databases
The goal is to develop reasoning algorithms for mining information from text based on semantic markups in literature abstracts. In contrast to classical searches the aim is to provide a table of content outlining the whole result set and providing links to subsets of documents grouped by topics. The goal is also to link bibliometric analyses to ontology-based literature search to discover trends, prominent authors and research institutions as well as important journals for single research topics.

1.1.3 Open problem 3: Design of a system for ontology-based literature search for the life sciences

Algorithms efficiently linking semantic concepts and free text as well as datamining methods using the background knowledge of ontologies and an annotated corpus to answer questions and reveal trends in the literature are the basis for an ontology-based search engine on the web.

There is currently no ontology-based search engine on the web which can handle hundreds of users, thousands of concepts and millions of documents. Typically web applications are expected to respond within a few seconds. The problem is that typically search results contain from hundreds to thousands of documents which need to be annotated. This disallows an architecture which annotates all documents on the fly. On the other hand new publications are added frequently, up to 5,000 per day in PubMed.

Another problem is the presentation of the results of an ontology-based search. A typical result links to several thousand concept labels. The graph representing the induced ontology including relations and definitions takes several megabytes of memory. This disallows to transfer the whole graph with each server response.

Textmining algorithms are imperfect and more training data is required to improve their accuracy. Manual annotations are time consuming. The goal is to design a community curation tool seamlessly integrated into the web-search to collect necessary training data to let users of the system improve the quality of the results.

Open problem 3: Design of a system for ontology-based literature search for the life sciences

The goal is to design a system that supports the answering of biomedical questions by processing search results of PubMed using the background knowledge of biomedical ontologies. The system must be capable of serving hundreds of users, ten thousands of concepts and millions of documents while responding immediately. A community curation tool is to be designed which allows user driven annotation.
1.2 Overview

The following diagram interrelates research areas relevant for this work. At the top, literature and annotation databases comprise the biomedical knowledge of today. Although more and more information is stored in structured databases, most knowledge is only accessible through text. Terminological Resources such as ontologies or taxonomies capture parts of the biomedical knowledge in a computable form. They are mainly used to annotate structured databases to facilitate datamining. Biomedical search engines support human curators when creating manual annotations and biomedical tools use background knowledge to reason over structured data. A prerequisite for the Semantic Web is the linkage of web resources with formally defined concepts. Textmining algorithms were previously used to automate the annotation process of free text with selected types of entities such as person names, dates, places, institutions as well as gene and protein names. This work applies textmining algorithms on terminological resources to automate the annotation with concepts of ontologies or taxonomies. One result of this work is the first online biomedical search engine using semantic technologies. The performance of the algorithms is measured with evaluation methods known from Information Retrieval.

In the background chapter of this work literature databases, annotation databases, biomedical tools and search engines, the semantic web, terminological resources as well as textmining and evaluation methods are introduced. Chapter 3 builds on terminological resources and textmining algorithms. Chapter 4 uses results of chapter 3 and combines them with reasoning and datamining. Chapter 5 describes the design of the first ontology-based search engine for the life sciences using the methods of the previous chapters.
Chapter 2

Background

This chapter describes related work having bearing on the present study. Ontologies and terminologies in the life sciences as the source of background knowledge for ontology-based search are introduced. Ontological resources build the basis for Semantic Web applications such as semantic search engines. The biomedical publishing domain is a forerunner in terms of semantic markup of unstructured text. One reason is the urgent need to overcome incompatible naming practices. A number of biomedical search engines tackle this problem. Also the life sciences domain is a pioneer in the application of ontologies to real problems. Various ontology-based tools to search, exchange, integrate, reason and visualize data and information have been developed with textmining only being one more way to employ background knowledge. The recently increased interest in large text corpora has been sparked by the increasing amount of freely available medical texts. Biomedical textmining methodologies borrow from previous research, e.g. on named entity recognition, and add terminological knowledge of the domain. Biomedical concepts are similar to Named Entities, which are instances of concepts such as real persons, companies, genes, proteins and others. However from the linguistic perspective, they differ in that Named Entities are mostly proper nouns, while biomedical concepts mostly contain common nouns. Previously developed approaches for Named Entity Recognition are described and standard evaluation methods are introduced. The chapter ends with an outline how this work is arranged and links the chapters to the pertaining related work.
2.1 Introduction

2.1.1 Recent advances

A major goal in the post-genome era [269] is the exploration of the order and logic of genetic programs [35]. Entire genome sequences are available [107, 54, 102, 76] for 662 organisms while currently there are 1280 ongoing projects [277].

These masses of data fuel high-throughput experiments [25] such as DNA microarrays [236]. Figure 2.1 shows methodologies and resultant data sets of proteomics, the large-scale study of proteins and their structures and functions [8], and functional genomics, the study of the dynamics of gene transcription, translation, and protein-protein interactions [121].

Structured biomedical databases. Biomedical knowledge is stored in structured databases [281, 21, 140]. SCOPPI [279] is a database of all domain-domain interactions and their interfaces derived from PDB structure files and SCOP domain definitions. In Teyra et al. [264] we describe SCOWLP, a web-based database describing protein interface interactions at atom, residue and domain level. It substantially enriches the description of protein interfaces by adding detailed interface information of peptidic-ligands and solvent to the existing protein-protein interaction databases. For an overview of structured biomedical databases see the yearly updated “Molecular Biology Database Collection” [94]. However, an overwhelming amount of biomedical knowledge is still recorded in text.

Biomedical literature. The biomedical literature is growing at a tremendous rate. PubMed, the most widely used biomedical literature database, has grown by 754,003 cited documents in the last year and now covers more than 18 million abstracts of scientific literature, although about half of them are retractions and corrections [247]. However, PubMed does not cover all published article citation in the field of biomedicine. Figure 2.2 shows an increase in biomedical literature in the most widely used citation database.

Most pieces of information found in one of the structured databases are also published in research articles. Many of such database entries even provide references in the literature as further evidence. However, not all information published in research articles is stored in structured databases.
2.1. INTRODUCTION

Figure 2.2: In blue, the number of citations in PubMed grows exponentially, hitting 754,003 new citations in 2007. The yellow line shows the number of review articles in PubMed. PubMed has published usage statistics since 1997 and the numbers have steadily increased by 97 million per year on average. The figure currently stands at 871 million per year.

Textmining. The last decade produced a large amount of interest in biomedical literature. Leek [158] aimed at detecting reported gene locations on chromosomes in the literature. Fukuda et al. [91] identified protein names in biological papers. Craven and Kumlien [59] described a method labeling documents mentioning subcellular locations. In Rindflesch et al. [229] the authors seek to find assertion for relations between drugs and genes. Friedman et al. [90] searched for molecular pathways in journal articles. Chang et al. [45] developed an algorithm to identify abbreviations from texts. One task in the work of Hersh [120] was to recover functions of genes from a document collection. Srinivasan [256] was generating biological hypotheses from Medline. Spasic et al. [255] stress the importance of using ontologies together with terminological glossaries for advanced textmining. In Ling et al. [161] a method to generate gene summaries from biomedical literature is described.

A considerable problem has to be solved when searching for the relevant literature about specific biological facts regarding, for example, proteins, genes, mutations, chromosome locations or diseases. Language is highly ambiguous. This adds to the problem that there is no common biomedical terminology. There is, of course, no standardized way to express scientific facts in free text. The naming of biological entities suffers greatly from inconsistencies. Entrez Gene [166] and GeneCards [221] aim to improve labeling consistency of biological entities. With new insights into the molecular functions of genes and proteins, cellular components and biological processes, a new vocabulary has evolved. However, until a few years ago this was an entirely unmanaged process. The Gene Ontology project [13] is a collaborative effort to address the need for consistent vocabulary in describing genes and gene products.

Ontologies. Numerous biomedical taxonomies and ontologies recently became available, such as SNOMED, UMLS, MeSH and the Gene Ontology. The first version of SNOMED was released in 1974. Since 2003, SNOMED CT, the latest release, has been integrated into the Unified Medical Language System (UMLS). UMLS is a project which combines several terminologies such as ICD-10 (International Classification of Diseases), MeSH [165], SNOMED [197], LOINC [85], Gene Ontology [13] and OMIM [113] into one resource. The Metathesaurus contains concepts, concept names, and other attributes from more than 100 terminologies, classifications, and thesauri; some in multiple editions. The Open Biomedical Ontologies (OBO) Foundry is an effort to co-ordinate the evolution of ontologies [253]. It now comprises 70 biomedical ontologies. One major goal of these efforts is to provide common
controlled vocabulary to facilitate data integration in bioinformatics.

Taking the recent advances in biomedical textmining and ontologies an interesting question arises. Can the background knowledge in ontologies be used to facilitate a semantic search in annotated text databases? The motivation of this work is discussed in the following section.

2.1.2 Motivation

Due to the enormous amount of available literature, simple keyword-based text search of the literature often fails to yield the best results, and a lot of important information remains buried in the masses of text.

Consider the following example: A researcher wants to know which enzymes are inhibited by levamisole. A keyword search for "levamisole inhibitor" produces well over 100 hits in PubMed. To find out about specific functions, the researcher has to go through all these papers. He/she is interested in the relevant enzymatic functions. From the first titles it is immediately evident that levamisole inhibits alkaline phosphatase. A less well-known fact is, however, still hidden among the abstracts. The abstract The effect of levamisole on energy metabolism in Ehrlich ascites tumor cells in vitro with PMID 2947578 is ranked very low (position 116 on 10/7/20081) by PubMed. The abstract states that levamisole also inhibits phosphofructokinases. Most readers will miss this statement.

Even if the user tries to reduce the number of papers by filtering out the ones mentioning "levamisole inhibitor" (e.g. query PubMed for "levamisole inhibitor NOT phosphatase"), he or she would miss the less obvious hits like "phosphofructokinase", if both terms occur in the same abstract. Thus, even advanced PubMed queries with boolean logic cannot always structure the search results properly.

Without knowing specific activities a user is interested in, refining a keyword in classical search is also difficult. For example, a search for "levamisole inhibitor enzymatic activity" produces only 5 hits: enough to learn about alkaline phosphatase, not enough to learn about the phosphofructokinase.

The Gene Ontology covers a broad spectrum of terms in the biomedical field. This vocabulary was initially intended to annotate proteins in structured databases. The associations of proteins with biological processes, molecular functions and cellular components are recorded in such databases as well as in biomedical literature. The terms in the Gene Ontology were partially semi-automatically derived from PubMed citations [157]. MeSH was initially intended to index the literature abstracts of PubMed. Therefore, these two controlled vocabularies provide the best coverage of concepts used in PubMed citations. PubMed makes use of manual MeSH annotations to guide literature searches for its users, for example, queries are expanded using MeSH headings. A user can set up a query for articles related to humans. The relevant articles do not need to mention the word “human”.

Some other vocabulary is used by PubMed as well. PubMed does not use Gene Ontology concepts to index its citations. The aim of this work is to annotate all PubMed citations with GO and MeSH concepts while users query the database in a familiar manner, e.g. users may use exactly the same query syntax and retrieve exactly the same relevant documents. In a second step the ontologies are used to classify documents hierarchically. This hierarchy is presented to the users who can then explore the results by navigating through the branches.

1Please note, that all examples in this paper depend on PubMed’s ranking of search results. Since the literature is growing, PubMed may return different articles for the same query at different time points. This means that GoPubMed may display different papers for the examples in this paper. All queries in this paper were checked on 7 October 2008.
When people search, they have questions in mind. Answering questions in a domain requires the knowledge of the terminology of that domain. Some search engines modify user queries using dictionaries. PubMed for example expands the user query "mouse" into "mouse or mice". The results are presented in the form of citation lists. Classical search engines do not provide background knowledge for the presentation of search results. This work aims to allow users to find answers by presenting search results in a structured way, employing the background knowledge of ontologies.

Research findings are captured in the scientific literature, which is growing exponentially (see figure 2.2). But at a closer glance it turns out that some research areas shrink, while others increase in size. Bibliometric analyses aim to shed light on such developments and to identify emerging trends. Such analyses date back to the 1960s [209] and typically focus on selected research topics [95], journals [31], or researchers [209, 187, 110, 34]. Garfield and Melino [95] investigate amongst other things the research on PROGRAMMED CELL DEATH. Despite programmed cell death being described 25 years ago, it took some 15 years until journals such as “Cell Death and Differentiation” emerged and the number of publications in the field in general took off. As an example for an analysis of a specific journal Boyack analysed the emergence and development of topics covered by PNAS [31]. A very active area of research aims to understand the social process of publishing by investigating co-author and co-citation networks [186, 187, 209, 110, 47, 34]. Such analyses allow one to identify authors in an organisation who work interdisciplinarily and connect otherwise unconnected co-author networks [187]. It becomes possible to animate citations of key publications over time [47], development of author and publication networks [34], for a better understanding of how groups form [110]. All of these analyses are useful to take a birds-eye view onto research.

This work aims to link such analyses to the ontology-based literature search, as implemented in the GoPubMed search engine, to support the discovery of trends on topics of interest to molecular biologists. The scientific vocabulary evolves over time. Advances in research introduce refined vocabulary extending general concepts with sub-concepts. The usage of synonyms differs between research communities. Garfield and Melino [95] point out that during the 1960s and 1970s researchers in the US used “programmed cell death” while their European colleagues used “apoptosis”. Ontologies capture synonymous terminology as well as different levels of abstraction. This work aims to reveal trends spanning over different times and different research communities with the help of underlying ontologies.

2.2 Terminologies and Ontologies in the life sciences

2.2.1 Lexical Resources

In linguistics, the lexicon of a language is its vocabulary, including its words and expressions. More formally, it is a language’s inventory of lexemes [2]. While monolingual dictionaries contain words and expressions from one language, multilingual dictionaries link words and expressions from different languages.

The Lexical Markup Framework defines a common standardized framework for the construction of Natural Language Processing (NLP) and Machine-readable dictionary (MRD) lexicons [86]). The goals of LMF are to provide a common model for the creation and use of lexical resources, to manage the exchange of data between and within these resources, and to enable the merging of large numbers of individual electronic resources to form extensive global electronic resources.
CHAPTER 2. BACKGROUND

WordNet is a lexical database which also provides an application programming interface [81]. WordNet organizes lexical entities in sets of synonyms to describe a concept. For example the synset defined as (biology) the process of an individual organism growing organically; a purely biological unfolding of events involved in an organism changing gradually from a simple to a more complex level contains the lexemes: growth, growing, maturation, development, ontogeny and ontogenesis. Lexemes can be contained in different synsets. The synset defined as the process in which something passes by degrees to a different stage (especially a more advanced or mature stage) contains the lexemes: development and evolution.

A synset may contain spelling variants, such as British or American English. WordNet is organized in categories: nouns, verbs, adjectives and adverbs. The total number of all unique noun, verb, adjective and adverb strings is currently 147,278. The noun synset is organized in a is-a hierarchy, e.g. parenchyma (the primary tissue of higher plants composed of thin-walled cells that remain capable of cell division even when mature) is a “substance-of” leafs (the main organ of photosynthesis and transpiration in higher plants). WordNet has no focus on the biomedical domain in particular and is missing most names for gene product symbols and cellular components [29]. Other dictionaries, such as the UMLS, are more widely used in this domain.

The UMLS Specialist Lexicon provides lexical information for processing natural language in the biomedical domain [36]. Coverage includes both commonly occurring English words and biomedical vocabulary. The lexicon entry for each lexical item records syntactic, morphological, and orthographic information. Currently, it provides 360,688 entries with 625,041 inflected forms. Lexical records in the SPECIALIST lexicon comprise the base form of the entry, e.g. anaesthetic, spelling variants, e.g. anesthetic, a unique entry identifier, a part of speech category, abbreviations and acronyms as well as other annotations. UMLS covers a broad biomedical vocabulary. However, it was not intended to include specific sub-domains in the lexicon. Gene and protein names as well as chemicals and drugs are covered by other lexical resources.

Resources for biomedical entities such as gene and protein names can be found in Genew [274], which provides data for all human genes which have approved symbols, Entrez Gene [166] is the NCBI’s repository for gene-specific information, UniProt [10] is a catalog of information on proteins, and GeneCards [221] is a database offering concise information on human genes and their mouse homologs. Chemicals are listed in PubChem, a free database providing chemical structures of small organic molecules and information on their biological activities. ChemIDPlus is a free, web-based search system that provides access to structure and nomenclature authority files used for the identification of chemical substances cited in National Library of Medicine (NLM) databases. ChEBI is a freely available dictionary of molecular entities focused on small chemical compounds. RxNorm provides standard names for clinical drugs. The National Drug Code (NDC) System provides a directory of selected agents, insulin formulations, prescription drug products, and herbal drugs.

2.2.2 Terminological Resources

Standardization of terminology in the medical field was driven by the need to characterize patient records uniquely. With the advent of molecularly based diagnostics and therapies a new dimension in the naming problem arises. Nomenclatures in bioinformatics have been ad hoc and disconnected from patient records [52]. The UMLS Metathesaurus is a vocabulary database that contains information about biomedical and health related concepts, their synonyms, and the relationships among them [242].
The UMLS Metathesaurus

The Metathesaurus is another part of the UMLS project. It was designed for use by system developers. The Metathesaurus is built from the electronic versions of many different sources of controlled vocabularies used in patient care, health service billing, public health statistics, indexing and cataloging biomedical literature and health service research. Applications using the Metathesaurus link different clinical or biomedical vocabularies, support information retrieval from databases and from free-text, link patient records to related literature or factual databases, facilitate natural language processing and automated indexing research, and structure data entries. The Metathesaurus is not a standardized vocabulary itself. It is designed to maximize the usefulness of existing vocabularies. Two prominent contributors to UMLS are the Medical Subject Headings and the Gene Ontology, which are described in the following sections.

Medical Subject Headings

The Medical Subject Headings (MeSH) thesaurus is a controlled vocabulary comprising biomedical and health-related topics. It is organized in 16 categories, table 6.2 lists the first level of MeSH. The United States National Library of Medicine maintains it mainly for indexing the PubMed literature database.

Main Headings. MeSH Main Headings (or descriptors) are arranged in both an alphabetic and a hierarchical structure. At the most general level of the hierarchical structure are very broad headings such as “Anatomy” or “Diseases.” More specific headings can be found at narrower levels of the eleven-level hierarchy, such as “Hallux” (level 8). MeSH currently defines nearly 25,000 main headings, see table 6.2 for examples. Each descriptor has a number of Entry Terms assigned. They consist of variations in form, word order, spelling, names of drugs or equipment of a descriptor. When found in an article, the descriptor is assigned as a heading for the citation. Currently, 162,887 English Entry Terms and 53,703 German Entry Terms are recorded. PubMed articles are indexed with the most specific headings available. For example, an article on cystic fibrosis will be indexed under the subject heading cystic fibrosis and not under the broader heading pancreatic diseases. Experts examine the full text of each journal article and assign the most specific MeSH terms applicable. The NLM Indexing Initiative supports human indexers by providing a ranked list of candidates [11]. The manual curation is a labor intensive task.

Sub Headings. MeSH provides 99 secondary headings commonly referred to as “Sub Headings”, or “MeSH qualifiers”, see table 6.4. Subheadings are used to define a topic better, narrow retrieval, or express a certain aspect of a main MeSH heading. For example, a user interested in locating information on the “diagnosis of cystic fibrosis” should NOT have to scan the entire list of references on “cystic fibrosis” to locate the few articles that discuss “diagnosis”. By qualifying the main subject heading “cystic fibrosis” with the subheading “diagnosis”, the user can retrieve articles which pertain to this single aspect of the disease.

Supplementary Concept Records (SCRs) In addition to these headings, there are more than 172,000 headings called Supplementary Concept Records in a separate thesaurus. SCRs are typically chemicals that indexers have seen in MEDLINE. Such a substance is not a Main Heading, but is stored as an SCR in MeSH so that the preferred form of the substance name can be controlled and added to MEDLINE citations as part of the regular
CHAPTER 2. BACKGROUND

indexing process. Unlike the Main Headings and Subheadings that are changed on an annual basis, new SCRs are added and existing ones edited on a daily basis.

**Peculiarities.** MeSH prefers clinical terms, such as “neoplasms”, as opposed to more common terms such as “cancer” or “tumors”. MeSH vocabulary is derived from a variety of languages. When searching for MeSH headings, users also have to consider Latin and Greek terms. MeSH permits term inversion. In an attempt to list headings in the same alphabetical sequence, some compound terms are NOT listed in natural word order, but in inverted order. For example, the term JUVENILE RHEUMATOID ARTHRITIS becomes ARTHRITIS, JUVENILE, RHEUMATOID in order to keep all arthritis subject headings in the same alphabetical sequence.

MeSH is not designed as an ontology, i.e. the relations of neighboring tree node are not strictly is-a relations in contrast to other ontologies which connect all concepts via a subsumption relation. However because MeSH’s headings are organized from general concepts in the upper levels to specific concepts in the lower levels which is the main assumption about the structure of the used ontologies MeSH is named an ontology in this work.

The scope of MeSH Headings is rather broad. Another resource is the Gene Ontology, which is classified in this work as a terminology. In Smith *et al.* [249] the authors point out that, in spite of its name, GO is not an ontology as the term is commonly used by information scientists or by philosophers. The GO Consortium itself names it a controlled vocabulary. The concepts in GO are more finely grained than those of MeSH, but the scope of GO is narrower.

**Gene Ontology**

Biological systems are complex and our knowledge of such systems is incomplete. There is no agreed vocabulary used for describing biological systems. Names can have different meanings and different names can label the same concept. The Gene Ontology (GO) is designed as a controlled vocabulary that can be applied to all organisms even if knowledge of gene and protein roles in cells is accumulating and changing.

The project has two parts: the hierarchically organized vocabulary and the database of gene and protein annotations. The vocabulary consists of three orthogonal (mutually independent) ontologies: BIOLOGICAL PROCESSES, MOLECULAR FUNCTIONS and CELL COMPONENTS. Molecular functions comprise elemental activities or tasks, e.g. DNA BINDING or a CATALYTIC ACTIVITY. Biological Process describe broad objectives or goals, e.g. MITOSIS or SIGNAL TRANSDUCTION. Cellular components list locations or complexes, e.g. NUCLEUS or RIBOSOME. Gene Ontology Annotation (GOA) is a project run by the European Bioinformatics Institute (EBI) that aims to provide assignments of terms from the GO to gene products in a number of databases [41].

The GO project was started in 1998 by a consortium of researchers studying the genome of three model organisms: Drosophila melanogaster (fruit fly), Mus musculus (mouse), and Saccharomyces cerevisiae (yeast) [13]. Many other model organism databases have contributed since then. See table 6.5 in the appendix for a list of participating databases. GO currently contains 27,863 terms applicable to a wide variety of biological organisms.

GO terms/concepts refer to the mental idea represented by one or more GO labels/names. In the electronic form of this document all GO terms are marked-up and linked to the term information page at [www.gopubmed.org](http://www.gopubmed.org). Most terms have textual descriptions, 98%, synonymous labels, 53%, and database references. Most labels of GO terms consist of multiple words, 98%. Section 2.7 discusses how labels are split into words. A typical GO term lineage is shown in table 2.1.
2.2. TERMINOLOGIES AND ONTOLOGIES IN THE LIFE SCIENCES

<table>
<thead>
<tr>
<th>Name of Gene Ontology term</th>
<th>ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>molecular function</td>
<td>GO:0003674</td>
</tr>
<tr>
<td>catalytic activity</td>
<td>GO:0003824</td>
</tr>
<tr>
<td>isomerase activity</td>
<td>GO:0016853</td>
</tr>
<tr>
<td>cis-trans isomerase activity</td>
<td>GO:0016859</td>
</tr>
<tr>
<td>peptidyl-prolyl cis-trans isomerase activity</td>
<td>GO:0003755</td>
</tr>
<tr>
<td>juglone-sensitive peptidyl-prolyl cis-trans isomerase activity</td>
<td>GO:0042028</td>
</tr>
</tbody>
</table>

Table 2.1: A typical term lineage in the Gene Ontology reveals common substring relations between GO terms. From bottom to top these terms are all linked via the is_a relation.

The compositional structure of GO terms is discussed in Ogren et al. [192]. The author suggests the addition of new GO terms based on the substring relations as shown in table 2.1. Some conventions established for the construction of GO terms actually explain its structure and may be used by textmining systems.

The Gene Ontology does not describe gene products, instead it defines particular attributes of gene products. For example, cytochrome c is a protein. GO does not contain this gene product itself but attributes of it, such as the cellular component PLASMA MEMBRANE RESPIRATORY CHAIN COMPLEX III, the biological process of CYTOCHROME COMPLEX ASSEMBLY and the molecular function OXIDOREDUCTASE ACTIVITY. This explains why so many (9543) terms in GO end with the word “activity”. Alcohol dehydrogenase can refer to the enzyme (EC 1.1.1.1) or to the molecular function of this gene product. Whenever this might be ambiguous GO appends the word “activity” to clarify that it refers to the attribute of the gene product. The reason for this is that a single gene product might have several molecular functions and many gene products can share a single molecular function. Names of gene products are only used in synonyms or in conjunction with the word class, e.g. DNA DAMAGE RESPONSE, SIGNAL TRANSDUCTION BY P53 CLASS MEDIATOR, to indicate that the term is not restricted to the gene product named or to the species in which the gene product is found.

The GO guidelines suggest generic term definitions. That means the definition of a parent term is used to define the discriminating characteristics which mark instances of the specific term as being different from sibling terms. This “aristotelian” pattern explains why so many (19.6%) terms extend a parent label by adding a prefix, as shown in table 2.1.

**Synonyms.** A GO synonym may be broader or narrower than the term string; it may be a related phrase; it may be alternative wording, spelling or may use a different system of nomenclature; or it may be a true synonym. While this flexibility allows GO synonyms to serve as valuable search aids, it creates ambiguity for semantic textmining systems. GO records a relationship type for each synonym. These relationships are stored in the OBO format GO file. Four synonym types are used: (1) “ornithine cycle” is an exact synonym of UREA CYCLE, (2) “cell division” is a broad synonym of CYTOKINESIS, (3) “pyrimidine-dimer repair by photolyase” is a narrow synonym of PHOTOREACTIVE REPAIR and (4) “virulence” is a related synonym of PATHOGENESIS. The related synonym scope is the default type.

**Relation types.** The GO ontology is structured as a directed acyclic graph (DAG), that means a term can have multiple parents, but no cycles are allowed. There are two types of relations used in GO: the is_a (subsumption) relation and the part_of relation.

The subsumption relation is transitive. All concepts of the subsumption hierarchy denote classes but not instances of biological concepts. The part_of relation has four possible
CHAPTER 2. BACKGROUND

<table>
<thead>
<tr>
<th>Type</th>
<th>Example</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>nucleus is part of a cell</td>
<td>exact synonym: cells may or may not have a nucleus</td>
</tr>
<tr>
<td>2</td>
<td>replication fork is part of chromosome</td>
<td>broad synonym: whenever replication fork occurs, it is a part of a chromosome, but chromosome does not necessarily have part replication fork</td>
</tr>
<tr>
<td>3</td>
<td>chromosome is part of nucleus</td>
<td>narrow synonym: nucleus always has chromosome as a part, but chromosome is not necessarily part of the nucleus</td>
</tr>
<tr>
<td>4</td>
<td>nuclear membrane is part of nucleus</td>
<td>related synonym: nucleus always has the part nuclear membrane, and nuclear membrane is always a part of the nucleus</td>
</tr>
</tbody>
</table>

Table 2.3: A part-of relation can have four different interpretations. The Gene Ontology uses only the interpretation 2 and 4.

interpretations: (1) something may or may not be part of something else, (2) something is a necessary part of something else, (3) something necessarily has another part and (4) two things do not exist without each other. Table 2.3 shows an example for each possible interpretation. In GO only type 2 and 4 are used.

A subsumption path of a term is complete if there exists at least one path via is_a relations to the root node. Not all terms of the current version of the Gene Ontology have a complete subsumption path. It is currently one effort of the consortium to complete all subsumption paths. This is motivated by the following reasons: (1) Ontological correctness: Ontologies describe existing things in the real world. Logically, everything that exists is a kind of something else. The Gene Ontology describes existing biological concepts. (2) Queries and reasoning: Queries to the ontology which rely on the complete subsumption paths fail to return the correct (complete) result. A query such as “show me all the different kinds of membrane” will miss those membranes classes which have no path to the concept membrane. (3) Tool support: Most ontology editors such as Protégé-Frames\(^2\), Protege-OWL\(^3\) and SWOOP\(^4\) assume a complete subsumption hierarchy. The GO consortium aims to improve the compatibility with such tools. (4) Visualization: A complete subsumption hierarchy allows tools to filter out other relation types. While still all concepts are linked, the navigation in the hierarchy is more intuitive. This will also allow ontology-based tools, such as GoPubMed, to use GO more consistently. The current version of GoPubMed treats part_of and is_a relations as equals for simplicity.

**True Path Rule.** The True Path Rule states that “the pathway from a child term all the way up to its top-level parent(s) must always be true”. The transitive character of the is_a relation must always reflect biologically correct statements. Taking the examples from table 2.1: the statement “peptidyl-prolyl cis-trans isomerase activity is a cis-trans isomerase activity and cis-trans isomerase activity is a isomerase activity” is biologically correct as well as “peptidyl-prolyl cis-trans isomerase activity is a isomerase activity”. Figure 2.3 illustrates the term lineage. The True Path Rule is

\(^2\)protege.stanford.edu/overview/protege-frames.html  
\(^3\)protege.stanford.edu/overview/protege-owl.html  
\(^4\)www.mindswap.org/2004/SWOOP/
2.2. TERMINOLOGIES AND ONTOLOGIES IN THE LIFE SCIENCES

Figure 2.3: The Gene Ontology term lineage beginning from term “peptidyl-prolyl cis-trans isomerase activity” to the root node. The solid arrows denote direct is_a relations. The dotted arrows denote transitive is_a relations of the term “peptidyl-prolyl cis-trans isomerase activity”.

also valid for the part_of relation. If the RHOPTRY MEMBRANE is part of the RHOPTRY and RHOPTRY is part of the APICAL COMPLEX. Then it must also be biologically correct to state: “RHOPTRY MEMBRANE is part of the APICAL COMPLEX”. However, if a reasoner does not know if all part_of links are of type 4 the weaker type 2 for the transitive part_of relation must be assumed.

In order to add new concepts while guaranteeing the True Path Rule additional auxiliary concepts have been introduced. Considering the example from a previous version of GO shown in figure 2.4a: CHITIN METABOLISM is a subclass of CUTICLE SYNTHESIS and CELL WALL BIOSYNTHESIS. A fly chitin synthase could be annotated to chitin biosynthesis and appear in a query for genes annotated to cell wall biosynthesis and its children. However, the reasoning a fly chitin synthase is a cell wall biosynthesis is wrong as flies do not have cell walls. To resolve this problem auxiliary terms have been introduced. Figure 2.4b shows the current version of this GO branch. The terms CUTICLE CHITIN METABOLIC PROCESS and CELL WALL CHITIN METABOLIC PROCESS along with their appropriate descendants were introduced as children of CHITIN METABOLIC PROCESS. Cuticle chitin metabolism terms now do not trace back to cell wall terms, guaranteeing the True Path Rule.

Word Information value. The anatomy of GO terms can be used to estimate the information value per word in a term name [67, 57]. Words like “activity” or “process” signal rather the association with the ontology branch BIOLOGICAL PROCESS than discriminating siblings or descendant concepts from each other. They clarify the general semantics of concepts. On the contrary, the mentions of those concepts in publications do not contain such general words, yet they refer to this exact meaning. It is a disambiguation task of a
A subsumption hierarchy violating the True Path Rule: Here a fly chitin synthase could be annotated as a CHITIN BIOSYNTHESIS, and appear in a query for genes annotated to CELL WALL BIOSYNTHESIS, which makes no sense because flies don’t have cell walls. The dashed nodes are replaced in the revised version of the GO branch shown in (b).

(b) The revised ontology structure ensures the True Path Rule. The parent chitin metabolic process now has the child terms CUTICLE CHITIN METABOLIC PROCESS and CELL WALL CHITIN METABOLIC PROCESS, with the appropriate CATABOLIC PROCESS and BIOSYNTHETIC PROCESS terms beneath them, as dashed nodes.
textmining system to separate mentions of gene products from mentions of attributes of the
gene product. The Gene Ontology merely describes the latter.

**Related research.** The GO Consortium stated that it “is increasingly difficult to maintain
the semantic consistency we desire without software tools that perform consistency checks
and controlled updates” [55]. Due to its popularity and fast growth it became more difficult
to maintain its semantic integrity manually. In Smith et al. [249] the authors pointed
out semantic inconsistencies of which some were resolved in more recent versions of GO.
Attempts to transfer GO’s knowledge into a more suitable form for use by computers are
the Gene Ontology Next Generation (GONG) project [280] and Protégé 2000 [289]. GO does
not cover all aspects of biological or evolutionary relationships. Complementary vocabularies
are covered by other ontologies, see the Open Biomedical Ontologies obo.sourceforge.net
described in section 2.2.3.

### 2.2.3 Ontologies

In philosophy ontology is the study of being or existence, the discipline describing the reality.
The term comes from the Greek *ontos*: of being and *logos*: word. Computer scientists
use the term for an explicit specification of a conceptualization [109]. This can be a hierar-
chy of entities, relations and rules. Computer scientists use the term loosely to name any
classification scheme. Some of them are more taxonomies rather than true ontologies. The
primary goal of biomedical ontologies is to provide an organizational framework of concepts,
axioms, and relationships that allows reasoning about biomedical knowledge. Ontologies
are implemented using knowledge representation (KR) languages. The frame-based biomed-
ical ontologies, e.g. Foundational Model of Anatomy [230], and description logics-based
ontologies, e.g. GALEN [224] and SNOMED-CT [44], are two prominent representations.
Reasoners are programs used to draw conclusions from facts by applying the knowledge
encoded in axioms and relations of one or more ontologies, for example, if we know that
all cellular components are located somewhere in the cell, particularly physically within or
attached to the cell membrane and we know endosomes are necessarily part of cells, then
we must not look for endosomes outside of cells. The Semantic Web [22] is an approach to
use ontologies to reason over facts stored as resources in the World Wide Web. Agents are
software programs that assist users and will act on their behalf in performing non-repetitive
computer-related tasks. In Doms et al. [65] we report on the usage of agents in bioinformat-
ics. It is believed that the integration of agent technology and ontologies can improve the
ability to perform tasks for users more efficiently and with less human intervention [116].

With the increasing usage of ontologies tools to develop and maintain became necessary.
A comparison of four different ontology editors can be found in Lambrix et al. [153]. Protégé
[191] and OBO-Edit [60] are among the most used among researchers while TopBraid Com-
poser5 is the commercial successor of Protégé.

**Upper level ontologies**

An upper ontology is limited to concepts that are meta, generic, abstract and philosophical.
The concepts are general theories such as the theory of parts and wholes, the theory of
dependence, and the theory of boundaries [28]. IEEE’s Standard Upper Ontology (SUO)
working group is developing such a standard [188]. The authors in Smith et al. [251] pro-
pose the OBO Relation Ontology providing the basis for the domain ontologies in OBO to
interoperate.

---

5 [www.topbraidcomposer.com/](http://www.topbraidcomposer.com/)
Domain ontologies

Domain ontologies represent knowledge in a restricted area of research. The Edinburgh Mouse Atlas Anatomy Ontology [38] is another domain ontology. It is part of the Edinburgh Mouse Atlas Project (EMAP). EMAP is a digital atlas of mouse development and serves as a framework for spatiotemporal data such as in-situ gene expression and cell lineage. It consists of 3D reconstructions (3D gray-level voxel images) of embryos at various stages of development, an anatomy ontology and mappings between the two. The authors in Lambrix et al. [154] describe among other examples of biological ontologies the Open Biomedical Ontologies. The project was started as an umbrella web address for ontologies for use within the genomics and proteomics domains. The member ontologies are required to be open, to be written in a common syntax, to be orthogonal to each other, to share a unique identifier space and to include textual definitions. Many bio-ontologies are already available via OBO.

The OBO Foundry [253] is a subset of the OBO ontologies and puts four additional requirements on the members: ontologies must (1) be developed in a collaborative effort, (2) use common relations that are unambiguously defined, (3) provide procedures for user feedback and for identifying successive versions and (4) have a clearly bounded subject-matter. Currently the OBO Foundry lists 51 candidate ontologies, see table 6.3 in the appendix.

Formal ontologies

A formal ontology defines axioms which are propositions that cannot be proved or derived otherwise. Formal ontologies do not aim at representing entities in a particular application scope. Broad distinctions condition which relations can be defined for categories. An example is the distinctions between “continuant” and “occurant” things. While continuant things can be perceived completely at any point in time occurant things can only be perceived in parts at any point in time. Material object are continuant. Processes are occurant. Such distinction restrict the application of qualities. For example a process, like a jump, can not have a color. Qualities do not exist on their own, they need another thing to exist. Examples of formal ontologies are the OBO Relation Ontology, Galen, SNOMED and the phenotype ontology.

The life sciences, unlike a discipline such as physics, has not yet reduced its laws and principles to mathematical formulas. The majority of life science ontologies are non-formal and heavily rely on human interpretation which requires deep domain knowledge. Computational use of the knowledge is also difficult due to the ambiguity of natural language.

Semantic Web applications build on the structured knowledge stored in machine processable ontologies. The following section introduces the vision of the Semantic Web.

2.3 Semantic Web

The current Web is essentially a collection of documents that are interconnected by links and it is used as a portal to applications. For instance, the biological data sources available on the Web provide access through Web pages. To query the data sources the users often fill out forms. The results are again presented to the users as Web pages. The Web pages are presented to the users based on mark-up. This mark-up mainly represents rendering information, such as the font and color of the text, and links to other Web pages. Therefore, the current Web is mostly a medium of documents for people rather than for information
that can be processed automatically by computers [22]. In Doms et al. [65] we report on the usage of software agents in bioinformatics.

**Semantic Web Vision** The Semantic Web is a vision of a further development of the World Wide Web “in which information is given well-defined meaning, better enabling computers and people to work in cooperation” [22]. The World Wide Web Consortium states it as follows [Consortium]: “The Web can reach its full potential only if it becomes a place where data can be shared and processed by automated tools as well as by people. ... The Semantic Web is a vision: the idea of having data on the Web defined and linked in a way that it can be used by machines not just for display purposes, but for automation, integration and reuse of data across various applications.”

**Semantic Annotation.** As a first step toward this vision of making the content of Web pages machine-understandable, people have started to use semantic annotation. One way to do this is to annotate the Web pages with ‘meaningful’ tags. In this case we annotate the Web pages with XML mark-up to distinguish the meaningful parts of the document. For instance, in a Web page about a protein we may distinguish between its name, coding DNA, three dimensional structure, family, function, source organism, etc. Then, we use programs that recognize the mark-up, and the different parts of the document can then be used in other programs based on the meaning represented by the mark-up. However, for this approach to be successful, there is a need for agreement on the annotation. A solution to this is to use ontologies to specify the meaning of the annotations. The ontologies define a vocabulary, specify the meaning of the terms and define how new terms can be formed by combining existing terms.

**Technology.** In the current Web service [Consortium] approach, data sources and tools can be seen as service providers and announce their services. Data sources, for instance, can announce their content and query capabilities. Users can be seen as consumers that request services based on their task. User requests and services are matched by service matchers. When we semantically enable the Web service approach, service providers are able to use ontologies to describe their services and users can use ontologies when formulating their requests. The service matchers will then find relevant services more easily. By using ontologies during information retrieval, it is possible to reduce the amount of non-relevant information in the returned results. For instance, when looking for information about jaguars, users may use an ontology to state that they are interested in the animal. The result will then only include documents about the animal. It is also possible to find more relevant information. For instance, when looking for information about signal transducers, we may take into account information from an ontology that states that receptors are a special kind of signal transducers. Therefore, also documents about receptors will be returned. Finally, semantic annotations can enhance the integration process. Entities in different data sources that are annotated with the same or related ontology terms are likely to be related. Relations between data items could be derived from relations (e.g. equivalent, is-a, part-of) between the ontology terms they are annotated with. Search engines are a particular kind of information retrieval systems. The following section characterizes a list of recently developed search engines in the biomedical field.
2.4 Biomedical Search engines

With the fast growth of the biomedical literature the number of specialized search engines tailored to the needs of medicals and biologists has increased. In 1997, the US government decided to make MEDLINE, the citation catalog of the National Library of Medicine, publicly accessible via the World Wide Web. In 2001, a new URL was introduced www.pubmed.gov. Since then it has become the most popular literature database online. With the introduction of the Entrez Programming Utilities and the availability of citations in XML format since 2000 the number of alternative PubMed interfaces has increased quickly. Biomedical search engines can be classified according to their focus on Information Retrieval support and Knowledge retrieval support [137]. However, it is not always possible to separate clearly. Table 2.5 shows how the biomedical search engines discussed in the following sections are categorized in this work. The categorization was done according to the most important features of each system which are further discussed in the sections and tables 2.7, 2.9, 2.11, 2.13 and 2.15.

<table>
<thead>
<tr>
<th>Information Retrieval</th>
<th>Knowledge Retrieval</th>
</tr>
</thead>
<tbody>
<tr>
<td>Improved Querying</td>
<td>Tools Integration</td>
</tr>
<tr>
<td>Results Processing</td>
<td>Semantic Processing</td>
</tr>
<tr>
<td>askMedline</td>
<td>BioIE</td>
</tr>
<tr>
<td>PubMedInteract</td>
<td>ReleMed</td>
</tr>
<tr>
<td>PICO Linguist</td>
<td>PubMed PubReMiner</td>
</tr>
<tr>
<td>BabelMeSH</td>
<td>ClusterMed</td>
</tr>
<tr>
<td>PubFinder</td>
<td>BioMetaCluster</td>
</tr>
<tr>
<td>CiteXplore</td>
<td></td>
</tr>
<tr>
<td></td>
<td>EBIMed</td>
</tr>
<tr>
<td></td>
<td>PubFocus</td>
</tr>
<tr>
<td></td>
<td>Harvester</td>
</tr>
<tr>
<td></td>
<td>XploreMed</td>
</tr>
<tr>
<td></td>
<td>AliBaba</td>
</tr>
<tr>
<td></td>
<td>Textpresso</td>
</tr>
<tr>
<td></td>
<td>Chilibot</td>
</tr>
<tr>
<td></td>
<td>MedStory</td>
</tr>
<tr>
<td></td>
<td>iHop</td>
</tr>
</tbody>
</table>

Table 2.5: Categorization of Biomedical Search Engines

2.4.1 Search engines focusing on Information Retrieval

Information Retrieval is the process of searching for documents or information in documents executed by a human user or automated agent. A system supporting a human user querying is aimed at increasing the ratio between relevant and non-relevant documents upon a query, e.g. web search engines. An automatic systems task is the aggregation of filtered information to reduce the number of documents requiring further processing, e.g. customizable RSS feed services.

**Improved querying**

PubMed expands user queries using MeSH headings and additional vocabularies such as drugs or chemicals. If a query contains such a term the query is expanded with the option to include also articles which were manually annotated with this term. In the PubMed interface this expanded query can be reviewed by the user. Also the E-Utilities can be called to compute this expansion. This query expansion helps retrieving relevant articles which otherwise would be missed.
Some tools aim to improve the querying of PubMed by supporting the user during query formulation. Features reach from language translation over graphical aims to pre-processing full English questions:

**askMedline.** The text-based website askMEDLINE [83] takes a natural language question as input. The system removes irrelevant words and the remaining words are tested to relate to MeSH headings by querying PubMed. Terms classified as “other eligible entries” are eliminated as well if the remaining search results are few. The result is always a list of citation titles and links to the abstract and full text.

**PubMedInteract.** PubMedInteract [183] is a web interface to PubMed and presents slider bars to set PubMed search limits and parameters. A “Preview Count” option computes the number of articles to be expected with the current settings.

**PICO Linguist.** PICO Linguist [84] offers non English medicals the option to build a structured clinical query with medical terms that may be difficult to express in English by using the PICO framework. The user may specify the patient’s problem, the therapy and alternative therapies and the outcome in his/her own language. Primary sources of vocabularies for translation are UMLS, MeSH, WHO EMRO and UMLF.

**BabelMeSH.** The BabelMeSH [84] website maps search terms to a multilingual MeSH in 12 different languages. Only terms listed in the multilingual vocabulary can be used for the query.

**PubFinder.** This service [101] aims to automatically extract Pubmed abstracts that deal with a specific scientific subject. The user enters a representative set of PubMed ids. Based on the abstracts, a list of discriminating words is calculated which is used for ranking Pubmed abstracts for their probability of belonging to the user defined topic. The first 100 words exhibiting the highest difference in occurrence between both the global PubMed frequency of a word in a reference dictionary and the frequency of a word in the selected abstracts make the list of discriminating words. A set of abstracts dealing with literature mining contains, for example, these words: abstracts, medline, information, articles, names, precision, database, recall, protein, literature, databases, references, system, automatically, interactions, set, mining, scientific, automated, motivation and others.

**CiteXplore.** CiteXplore indexes documents from sources like Medline, European Patent Office, Chinese Biological Abstracts and Citeseer using the Lucene full text index. Advanced searches such as wildcard search on selected attributes is offered. Another option is the expansion with synonyms. Information gathered from other applications such as Inter-Pro, SwissProt/Trembl and Alternative Splicing is cross referenced. The external WhatIsIt textmining service is used to highlight proteins, genes and protein-protein interactions. The references can be exported to EndNote, RIS and Bibtex format.

**Results processing**

Some systems process search results further to facilitate browsing of a large number of documents or link to further related citations based on the content of the search result. Examples are evidence highlighting, document re-ranking and information organization. Evidence highlighting visually emphasizes text passages in source documents. For example, the word
in a sentence stating a relation of two entities is underlined. Readers are supported when scanning through relevant text passages. Documents can be sorted according to selected criteria such as date, type of citation, usage of vocabulary or reputation. Hyperlinks to documents not in the original search result, for example referenced papers or papers with similar content are linked, support researchers in finding all relevant material. Information organization is the process of organizing information such that it becomes useful. For example tables or network graphs support understanding.

**BioIE.** BioIE is a rule-based system that extracts informative sentences from MEDLINE document or uploaded texts. Informative sentences refer to structures, functions, diseases and therapeutic compounds, localisations or familial relationships of biological entities, particularly proteins. The selected text base can be visualized in tabular form as word, MeSH term and word phrase frequency tables. Textual templates are used to identify informative sentences of a selected type, e.g. functional descriptions. The sentences can be further filtered for cooccurrence with additional keywords.

**ReleMed.** ReleMed [246] expands a users query automatically using UMLS and MeSH. Names of proteins and genes are expanded as well. Also lexical variants of words are generated. The user has the option to undo this expansions selectively. Matches in separate sentences are highlighted. ReleMed uses the relational MySql database to implement a full text index over single sentences. MeSH headings associated with the abstracts are concatenated and treated as an additional sentence. The relevance of an article is defined in eight levels depending on the cooccurrence of all keywords in one or more sentences.

**PubMed PubReMiner.** PubMed PubReMiner [146] shows the user journals in which his/her keywords are mentioned the most. It displays authors publishing the most articles mentioning the keywords. It shows words that have been used most in the title and abstract of the articles. Queries can be refined based on document attributes such as address, substances, MeSH headers, publication year, author and others.

**ClusterMed.** Vivisimo® applies clustering methods in ClusterMed and BioMetaCluster [263]. In ClusterMed PubMed results are clustered in various ways. Document distances are computed based on strings in (1) title, Abstract, and Medical Subject Headings, (2) title, abstract only, (3) MESH only, (4) authors name only, (5) affiliation only and (6) date of publication only. Vivisimo uses words found in this document’s attributes to label clusters. The clusters are ordered by the number of documents contained in them. The cluster hierarchy is computed using statistical language processing. For the query ”rab5” ClusterMed returned several clusters such as Vacuoles, Phagosomes, Rabaptin-5, Rab5a and others. The labels are computed on the basis of word occurrence statistics in the retrieved article abstracts. The cluster “Rabaptin-5” contains sub-clusters such as “Ubiquitin”, “GAT domain”, “vesicular transport”, “Nucleotide exchange”, “Dimerization Of Rabaptin-5”, “Endocytic membrane fusion”, “Correlated, Tissue”, “FRET microscopy”, “Cleaved in apoptotic” and other labels. Most of the labels could be categorized in the context of biomedicine as proteins, cellular components, molecular functions, diseases, techniques and others.

ClusterMed gives the option to compute the clusters only on the MeSH headings. The same string based clustering techniques are applied but using only words from MeSH. A clustering result for the query ”rab5” displays clusters labeled with MeSH headings such as “Guanine Nucleotide Exchange Factors”, “Virology” and “Pathology” but also concatenated labels such as “Analysis, Liver”, “Chromatography, Affinity, Cattle”, “Phagosomes,
2.4. BIOMEDICAL SEARCH ENGINES

Microbiology etc. which do not correspond to a single MeSH heading or sub-heading but to a combination of them. A cluster does not necessarily comprise sub-clusters reflected by a relation in the UMLS. In the examples the cluster “Guanine Nucleotide Exchange Factors” comprises labels of cellular components, diseases, peptides, proteins and algorithms. The clustering algorithm grouped them on the basis of statistical co-occurrence in the result set. No information about relations between headings is used.

Another feature of ClusterMed is the clustering by authors. Here, the strings of the last name plus the initials are clustered. Sub-clusters contain co-authors. The clusters may contain PubMed citations of different authors with same last name and initials.

BioMetaCluster. is a meta search engine based on the Vivisimo clustering architecture. It queries 22 web resources relevant for the biomedical domain using string based clustering of the search results.

2.4.2 Search engines focusing on Knowledge Retrieval

In Yao et al. [286] the authors compare knowledge retrieval systems and define their task as finding knowledge from information and organizing it into structures that humans can use. Following Ackoff [1], the human mind can be classified into data, information, knowledge, understanding and wisdom. (1) Data simply exists and has no significance beyond its existence. Symbols such as raw numbers are data. (2) Information is data that has been given meaning by way of relational connection. It adds context. This meaning does not need be useful. A relational database holds structured relational data. (3) Knowledge is the appropriate collection of information, such that its intention is to be useful. Computer programs modeling or simulating some process apply knowledge. (4) Understanding is an interpolative and probabilistic process. With understanding one can synthesize new knowledge or at least information. Bellinger et al. [19] argue that some artificial intelligence systems can generate new knowledge and are therefore “understanding”. (5) Wisdom is an extrapolative and non-deterministic, non-probabilistic process. Read the essay from Sharma [245] for an interesting philosophical discussion. Bellinger et al. [19] suggest an interpretation of the concepts as shown in figure 2.5. The transitions from data to information, to knowledge, and finally to wisdom is achieved by understanding.

Tools integration

Some biomedical search engines focus on the integration of external programs to process search results. Such tools call API methods or web services to analyze single citations or batches of citations. These features are influenced by the ideas of the Semantic Web, as discussed in section 2.3. Some services allow for various data exchange formats such as RDF, XML, BibTex, RIS, Endnote and plain text.

HubMed. HubMed [72] offers a battery of external tools to process PubMed results. The search behaves exactly like the original in PubMed if one chooses to use the option sort by date. An alternative is the option sort by relevance. Here, an Apache Lucene index is employed but no MeSH headings are expanded as is done via the PubMed search. The option sort by relevance seems to favor citations containing all query keywords. Some internal tools help managing references. A clipboard stores a set of citations for later reuse. The history function enables the recovery of previous searches. Citations can be tagged with arbitrary keywords for later filtering for them. Moreover, tags of other users may be used.
<table>
<thead>
<tr>
<th>Biomedical Search Engine</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>GoPubMed</td>
<td>Query transformation, Explore controlled vocabulary, Personalization, Link to original resultset</td>
</tr>
<tr>
<td>PubMed</td>
<td>Query expansion, retrieval, PMID + sentence + link, Personalized search, Tree exploration, Advanced search (using GO)</td>
</tr>
<tr>
<td>BabelMesh</td>
<td>Language translation, Personalized search, PMID + sentence + link</td>
</tr>
<tr>
<td>EBIMed</td>
<td>Meta-search in web resources, Text snippets</td>
</tr>
<tr>
<td>ClusterMed</td>
<td>Via PubMed, Title + Abstract + Links</td>
</tr>
<tr>
<td>XplorMed</td>
<td>Via PubMed, DB entry references, PMCID + sentence + link</td>
</tr>
<tr>
<td>Expert Mapper</td>
<td>Predefined topics, Title + Link</td>
</tr>
<tr>
<td>Pubfinder</td>
<td>Replacement by representative list of PMIDs, Results can be stored in an account, PMID + sentence + link</td>
</tr>
<tr>
<td>iHOP</td>
<td>Gene/protein name search in PubMed sentences, Personalized search, PMID + sentence + link</td>
</tr>
<tr>
<td>PubFocus</td>
<td>Via PubMed, Title + Abstract, Clipboard</td>
</tr>
<tr>
<td>HubMed</td>
<td>With MeSH terms and related words, Clipboard, Title + Abstract + Links + Tools</td>
</tr>
<tr>
<td>ReleMed</td>
<td>MySQL full text index on sentences, PMCID + sentence + link</td>
</tr>
<tr>
<td>Alibaba</td>
<td>Bypassed/replaced PubMed, stored queries, nodes in graph link back to evidence sentences</td>
</tr>
</tbody>
</table>

Table 2.7: This table compares biomedical search engines according to their features regarding the querying process.
## 24. BIOMEDICAL SEARCH ENGINES

<table>
<thead>
<tr>
<th>Biomedical Search Engine</th>
<th>Evidence explanation</th>
<th>Re-ranking</th>
<th>Connection to other related documents</th>
<th>Information aggregation</th>
</tr>
</thead>
<tbody>
<tr>
<td>PubMed, GoPubMed, askMedline, PubMed Interact, PICO Linguist, BabelMesh, Pubfinder</td>
<td>highlighted keywords and focused concepts</td>
<td>sentence cooccurrence of keywords and focused terms</td>
<td>via PubMed</td>
<td>concept graph, (partially) directed relations</td>
</tr>
<tr>
<td>RelMed, AliBaba, PubMed PubReMiner</td>
<td>highlighted keywords</td>
<td>sentence cooccurrence of keywords</td>
<td>via PubMed</td>
<td>external tools</td>
</tr>
<tr>
<td>ClusterMed, BioMetaCluster</td>
<td>highlighted strings of cluster labels</td>
<td>distance of strings in labels</td>
<td>via PubMed</td>
<td>most prominent authors according to different indexes</td>
</tr>
<tr>
<td>iHOP</td>
<td>highlighted/hyperlinked genes/proteins and interaction types in full abstract</td>
<td>IE confidence and external evidences</td>
<td>via PubMed/Lucene Index</td>
<td>significantly concuring words</td>
</tr>
<tr>
<td>HubMed, PubFocus</td>
<td>Word concurrence in sentences</td>
<td>Lucene Index based</td>
<td>via PubMed/Lucene Index based</td>
<td>hyperlinked graph of directed relations</td>
</tr>
<tr>
<td>Expert Mapper, EBI Med, XplorMed, Textpresso, Chilibot</td>
<td>highlighted matches</td>
<td>number of concurrences</td>
<td>via PubMed</td>
<td></td>
</tr>
</tbody>
</table>
Figure 2.5: The diagram shows the transitions from data to information, to knowledge, and finally to wisdom. Understanding is not a separate level of its own, it is necessary for the transition from each stage to the next. Figure adopted from [19]. Ackoff [1] points out that information ages quickly, knowledge has a longer time span, only wisdom is permanent.

The search can be narrowed or widened with the most closely related words. The relatedness is computed using a tf-idf ranking [134] of the words of the first 500 citations. Another option to manipulate the original query is the clustering feature of HubMed. The Lingo algorithm [156] is used to cluster the first 200 citations of the original query. The clusters are linked back to the first 20 citation of each cluster.

An external utility called by HubMed is the Entrez’ ELink utility. For some articles, generally if an article refers to the discovery or sequencing of a gene or protein, the inter-database links are presented by HubMed. Another tool employed is the Whatizit web service [223], which recognizes terms such as protein names and biological processes, linking them to services such as UniProt and Gene Ontology. Citations in the clipboard can be visualized with the TouchGraph Java applet\(^6\). HubMed focuses on improved querying but also on tool integration and thereby supporting knowledge retrieval.

**PubFocus.** PubFocus [205] is a bibliometric statistics tool integrating external data and web services to process PubMed searches and provide ranked lists of prominent authors, cities and journals. The ranking of citations is based on journal impact factors, volume of forward references, referencing dynamics and authors’ contribution level. PubFocus uses the non-free Journal Citation Reports\(^\text{®}\) Impact Factors published by Thomson Scientific. Forward citation information is based on PubMed Central and Google Scholar. The data retrieval is executed online by parsing the external websites HTML output. The authors define several indexes such as citations-over-age index, the Combined Impact Factor, the Cumulative Impact Factor and the Author’s Rank. Terms of the NCI thesaurus and the MGD mammalian gene ontology database occurring within titles and/or abstracts of citations are

\(^6\)sourceforge.net/projects/touchgraph
extracted using a MySQL full text search. Statistics for such terms are also integrated in the web interface and serve for further refinement of the initial query.

**Harvester.** Harvester crawls and cross-links the following bioinformatic sites: BLAST, CDART, CDD, ensEMBL, Entrez, GenomeBrowser, gfp-cDNA, Google-Scholar, GoPubMed, H-inv, HomoloGene, Hwki-Forum, iHOP, IPI, MapView, Mitocheck, OMIM, Polymeta, PSORT II, SMART, SOSUI, SOURCE, STRING, Unigene, UniprotKB, Wikipedia. Harvester cross-links public bioinformatic databases and prediction servers to provide fast access to protein specific bioinformatic information.

**Semantic processing**

Various techniques to overcome the semantic gap between text and its meaning are employed by biomedical search engines. A strong assumption made by some tools is that co-occurrence of biological entities in a sentence potentially indicates an observation or hypothesis of an interaction in vivo. Biological entities have a highly ambiguous terminology. Disambiguation techniques aim at solving this problem. Some tools make use of relations of concepts in taxonomies or ontologies and employ reasoning techniques. If findings can be confirmed to be significant such information is aggregated and can be seen as knowledge and used for question answering. A form of dialog can guide knowledge retrieval by directing what kind of knowledge is requested. Finally, some tools experiment with hypothesis generation, used knowledge and some statistical signals to present potentially new knowledge which needs to be confirmed.

**EBIMed.** Rebholz-Schuhmann et al. [222] identify associations between protein/gene cellular components, biological processes, molecular functions, drugs and species. Results are presented in tabular form. Sentences supporting the associations are cited. The tabular form of presenting many relations between biological entities ordered by their frequency is a way of providing the user with a quick overview. Such an accumulated form of information can support literature search for associations because the user reads mainly relevant sentences. The authors claim that EBIMed is complementary to PubMed in the sense that large result sets in PubMed are tedious to read through while EBIMed’s tables are expected to be of more help for larger sets of citations.

**GOAnnotator.** This tool aims to support database curators. Their task is to confirm automatic database curations by linking annotations to experimental results described in peer-review publications. The tool task protein identifiers and automatically annotated Gene Ontology concepts, e.g. by sequence similarity. In a following step the most similar GO concept which can be traced back in a text mentioning also the protein are displayed. Curators can now manually curate the annotations. The precision of the system is high due to the focused search with the previously annotated similar concept.

**Info-PubMed.** Info-PubMed provides information from Medline on protein-protein interactions. Given the name of a gene or protein, it shows a list of the names of other genes/proteins which co-occur in sentences from Medline, along with the frequency of co-occurrence. Information Extraction techniques are used to identify a set of sentences which clearly indicate interactions. The user interface allows to collect statements in a drag&drop manner and to visualize them using an external tool.
XplorMed. Perez-Iratxeta et al. [200] allow the user to explore a set of Medline abstracts. Three entry points are offered: Medline search, a set of document PMIDs or via a database entry with associations to Medline. The system then classifies the abstracts based on the top hierarchy of the MeSH terms associated to each MEDLINE entry. One or more top level categories of MeSH can be selected. The associated articles are now analyzed for words that are significantly related to others in this subset of articles. A main contribution of XplorMed is the functionality to explore the vocabulary used in a set of articles. The context of prominent words from the abstracts can be visualized. This is a list of other words in the texts which appear frequently together with such words. Based on this analysis, chains of words, an ordered set of words where the 2nd depends on the 1st, the 3rd depends on the 2nd, and so on, can be selected. The selected chain of words may be used to re-rank the initial Medline citations. The more chain words appear in an abstract the higher it is ranked. The authors claim that XplorMed can be used for sets of articles for which the user does not initially know what to expect. Prominent vocabulary is revealed, which later is used for re-ranking.

iHOP. Information Hyperlinked over Proteins [126] is a website offering hyperlinked navigation of PubMed abstracts via gene/protein mentions in sentences. Upon a search the user is presented a list of sentences containing concurrent gene/protein mentions. An interaction of the concurring entities is assumed and the predicted type of the interaction is highlighted. In case of existence of large scale experimental evidence of an interaction this is indicated as well by a link to the experimental results. Gene/protein name disambiguation is a difficult task [290]. iHOP enables users to verify its findings by highlighting the entity names in the original sentences. Thus, research will be able to confirm the findings. Three levels of confidence of the algorithm are indicated. The user may create gene models by appending interactions of genes/proteins to a graphical representation. While iHOP is an Information Retrieval tool as it displays PubMed sentences for gene/protein mentions it additionally disambiguates such entities. This enables a semantic connection between sentences also via synonymous labels of the same entity. Furthermore, this can be used to create interaction networks manually or automatically. This is a potential source of new insights into previously published data potentially supporting new hypotheses. Therefore, iHOP is recognized as a Knowledge Retrieval tool as well.

ExpertMapper. For a selection of 105 topics Expert Mapper computed prominent authors from Medline citations of the years 1997 to 2006 grouped into geographical regions. The main contribution of Expert Mapper is the accumulation of affiliation information for an author so that it becomes possible to make a reasonable manual prediction about the identity of an individual.

Textpresso. Textpresso [179] is an ontology-based search engine built of scientific literature on C. elegans and selected others domains. The texts are indexed with biological concepts and relations. The labels fall into 33 categories that comprise the Textpresso ontology. On a second level Textpresso maintains ca. 14,500 regular expressions representing known formulations of relations of a parent category with other entities. A selection of full text articles of selected species is indexed. The user can retrieve sentences mentioning keywords and concepts. Currently, the list comprises 101 concepts, some of which are known from the Gene Ontology. For each category Textpresso maintains a list of regular expressions used to index the texts. Textpresso can retrieve abstracts mentioning a life stage in C. elegans and a cell part. The indexation of an article with a descendant concept implies
the indexation also with the ancestor concepts. Each concept in the ontology has its own identification algorithm. Textpresso provides ten-thousands of indexed articles containing more than 222,000 facts.

**AliBaba.** AliBaba [204] visualizes PubMed as a graph, see figure 2.6. It parses PubMed abstracts for biological objects and their relations as mentioned in the texts. Ali Baba visualizes the resulting network in graphical form, thus presenting a quick overview over all information contained in the abstracts. A variety of relations between proteins, (sub)cellular locations, genes, drugs, tissues, diseases and others are detected. The extracted relations have a confidence value which can be used for filtering less likely correct associations. The interactive graphical representation allows for human interpretation of high dimensional data.

**Chilibot.** Chen and Sharp [46] search PubMed abstracts for specific relationships between proteins, genes, or keywords. The user enters a list of two or more genes or other keywords. PubMed searches for citations mentioning those entities or a synonym in one sentence. The resulting sentences are later categorized into six types: (1) Interactive relationship (stimulative), (2) Interactive relationship (inhibitory), (3) Interactive relationship (both stimulative and inhibitory), (4) Interactive relationship (neutral), (5) Non-interactive (i.e. parallel) relationship and (6) Abstract co-occurrence only. The relations are then visualized in a 2D graph with colored nodes and edges. The nodes denote the biological entity or keywords, and the edges denote the observed type of relation and its count. The graph is hyperlinked with the original set of sentences the relations were derived from. The user can confirm each relation and remove false edges from the graph.

Another feature of Chilibot is the generation of new hypotheses. Two nodes which are not directly connected to each other can be searched for the missing link. A graph of potential indirect interactions is drawn, see figure 2.7. The hypothesis is made on the basis of common connections to other entities, built by association, a principle previously used in gene function studies [214]. PubGene [132] is similar to Chilibot relying on non-directional interactions. PubGene Webtools allow users to analyze gene expression data with literature network information, browse literature neighbors of a given gene, search literature articles for a set of genes, search ontology terms related to a given gene, search MeSH terms found with a set of genes, and search for official nomenclature.

**Medstory.** Medstory⁷ groups result items into categories. The categories show users how the results distribute. Each category suggests further topics. Selecting of the sub-topics gives the choice of starting a new search with a narrowed query. The main researchers in the area are listed. Medstory is focused on the non-medical expert.

---

⁷medstory.com
Figure 2.6: AliBaba: What are the risk factors of treating G6PD-deficient malaria patients with primaquine? The query entered in Ali Baba was “primaquine malaria g6pd deficiency”. The graph shows the connection between G6PD (deficiency), vivax malaria (patients), and their treatment with primaquine. The risk factors hemolytic anemia/hemolysis are directly connected to G6PD and primaquine. Other nodes in the graph present more information on studied populations, the viral origin, infected cells, and information on the disease. Example from AliBaba’s website.

Figure 2.7: Chilibot: Relations visualization in Chilibot. A new hypothesis is shown that CREB interacts indirectly with synaptotagmin via one or more of 22 other genes. This hypothesis is discussed in Plake et al. [204].
<table>
<thead>
<tr>
<th>Biomedical Search Engine</th>
<th>Special features</th>
<th>Knowledge Retrieval Tools integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>PubMed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GoPubMed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>askMedline</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PubMed Interact</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PICO Linguist</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BabelMesh</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pubfinder</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ReleMed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AliBaba</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2D graph visualization</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sorted statistics</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PubMed PubReMiner</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ClusterMed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BioMetaCluster</td>
<td></td>
<td></td>
</tr>
<tr>
<td>iHOP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HubMed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PubFocus</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Expert Mapper</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EBiMed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>XplorMed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Textpresso</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chilibot</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2.11: This table compares biomedical search engines regarding their integration of external web-services.
### Table 2.13: This table compares biomedical search engines according to their features for linking results to external knowledge.

<table>
<thead>
<tr>
<th>Feature Description</th>
<th>PubMed</th>
<th>GoPubMed</th>
<th>via advanced searches for ontology terms</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Word sense disambiguation</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Author disambiguation</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Annotation propagation</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Top categories of leading experts on the query topic</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Gene lists ordered by frequency of concurrence</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- ν Entities denote edges in graph of concurrence of entities and categories</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Acronym disambiguation</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Edges have explaining synopsis of evidence (conclamative)</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Directed edges of evidence (conclamative)</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Edges correspond to top ranked MeSH terms of evidence</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Relations denote edges in graph of concurrence of entities and directed relations of entities</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- CBM (ChiliBots)</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- EBM (Evidence-Based Medicine)</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Expert Manager</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- PubMed Linkout</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- BabelMesh</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Pubfinder</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- ReleMed</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- AliBaba</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Relations denote edges in graph of concurrence of entities and categories</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Acronym disambiguation</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Edges have explaining synopsis of evidence (conclamative)</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Directed edges of evidence (conclamative)</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Edges correspond to top ranked MeSH terms of evidence</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Relations denote edges in graph of concurrence of entities and directed relations of entities</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- CBM (ChiliBots)</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- EBM (Evidence-Based Medicine)</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Expert Manager</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- PubMed Linkout</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- BabelMesh</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Pubfinder</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- ReleMed</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- AliBaba</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Relations denote edges in graph of concurrence of entities and categories</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Acronym disambiguation</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Edges have explaining synopsis of evidence (conclamative)</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Directed edges of evidence (conclamative)</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Edges correspond to top ranked MeSH terms of evidence</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Relations denote edges in graph of concurrence of entities and directed relations of entities</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- CBM (ChiliBots)</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- EBM (Evidence-Based Medicine)</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Expert Manager</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- PubMed Linkout</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- BabelMesh</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Pubfinder</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- ReleMed</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- AliBaba</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Relations denote edges in graph of concurrence of entities and categories</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Acronym disambiguation</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Edges have explaining synopsis of evidence (conclamative)</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Directed edges of evidence (conclamative)</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Edges correspond to top ranked MeSH terms of evidence</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Relations denote edges in graph of concurrence of entities and directed relations of entities</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- CBM (ChiliBots)</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- EBM (Evidence-Based Medicine)</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Expert Manager</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- PubMed Linkout</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- BabelMesh</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Pubfinder</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- ReleMed</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- AliBaba</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Relations denote edges in graph of concurrence of entities and categories</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Acronym disambiguation</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Edges have explaining synopsis of evidence (conclamative)</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Directed edges of evidence (conclamative)</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Edges correspond to top ranked MeSH terms of evidence</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>- Relations denote edges in graph of concurrence of entities and directed relations of entities</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
</tbody>
</table>
### Biomedical Search Engines

<table>
<thead>
<tr>
<th>Biomedical Search Engine</th>
<th>Question answering</th>
<th>Dialog</th>
<th>Hypotheses generation</th>
</tr>
</thead>
<tbody>
<tr>
<td>PubMed</td>
<td>implicit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GoPubMed</td>
<td>implicit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>askMedline</td>
<td>implicit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PubMed Interact</td>
<td>implicit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PICO Linguist</td>
<td>implicit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BabelMesh</td>
<td>implicit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Publinder</td>
<td>implicit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RekMed</td>
<td>implicit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AliBaba</td>
<td>implicit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PubMedPubMedMiner</td>
<td>implicit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ClusterMed</td>
<td>implicit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BioMetaCluster</td>
<td>implicit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>iHOP</td>
<td>implicit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HubMed</td>
<td>implicit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PubFocus</td>
<td>implicit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Expert Mapper</td>
<td>implicit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EBIMed</td>
<td>implicit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>XplorMed</td>
<td>implicit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Textpresso</td>
<td>implicit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chilibot</td>
<td>implicit</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2.15: This table compares biomedical search engines according to their features for linking results to external knowledge.
## CHAPTER 2. BACKGROUND

<table>
<thead>
<tr>
<th>Biomedical Search Engines</th>
<th>Information Retrieval</th>
<th>Knowledge Retrieval</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Querying</td>
<td>Result processing</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PubMed</td>
<td>1</td>
<td>14, 15, 16</td>
</tr>
<tr>
<td>askMedline</td>
<td>5</td>
<td>15</td>
</tr>
<tr>
<td>PubMed InterAct</td>
<td>1</td>
<td>23, 24</td>
</tr>
<tr>
<td>PICO Linguist</td>
<td>49</td>
<td>23, 25</td>
</tr>
<tr>
<td>BabelMesh</td>
<td>45</td>
<td>23, 24</td>
</tr>
<tr>
<td>PubFinder</td>
<td>6</td>
<td>25, 26</td>
</tr>
<tr>
<td>RelxMed</td>
<td>7</td>
<td>23, 25</td>
</tr>
<tr>
<td>PubMed PuReMiner</td>
<td>13</td>
<td>23, 25</td>
</tr>
<tr>
<td>ClusterMed</td>
<td>1</td>
<td>23, 25</td>
</tr>
<tr>
<td>BioMetaCluster</td>
<td>99</td>
<td>28</td>
</tr>
<tr>
<td>PubFocus</td>
<td>1</td>
<td>23, 24</td>
</tr>
<tr>
<td>HubMed</td>
<td>81</td>
<td>21, 22</td>
</tr>
<tr>
<td>BioIE</td>
<td>1</td>
<td>21, 22</td>
</tr>
<tr>
<td>CiteXplor</td>
<td>23</td>
<td>22, 23</td>
</tr>
<tr>
<td>IHOP</td>
<td>2.7</td>
<td>19</td>
</tr>
<tr>
<td>Info-PubMed</td>
<td>12</td>
<td>19</td>
</tr>
<tr>
<td>EBIMed</td>
<td>1</td>
<td>24, 25</td>
</tr>
<tr>
<td>GoPubMed</td>
<td>12</td>
<td>24, 25</td>
</tr>
<tr>
<td>AllBaba</td>
<td>1</td>
<td>25, 25</td>
</tr>
<tr>
<td>XplorMed</td>
<td>6</td>
<td>23, 24</td>
</tr>
<tr>
<td>GOAnnotator</td>
<td>12</td>
<td>23, 24</td>
</tr>
<tr>
<td>Textpresso</td>
<td>17</td>
<td>23, 24</td>
</tr>
<tr>
<td>Chihibit</td>
<td>11</td>
<td>23, 24</td>
</tr>
</tbody>
</table>

Table 2.16: Comparison of Biomedical Search Engines, descriptions on the following page.
2.4. BIOMEDICAL SEARCH ENGINES

Summary of Comparison. Table 2.16 summarizes the features of all compared search engines. The features are as follows: (1) PubMed query expansion/refinement: expands MeSH headings and additional vocabularies such as drugs or chemicals, citation metadata, (2) expands gene/protein names with synonyms, (3) offers narrowing/expanding with ontology concepts, (4) language translation of terms, (5) full natural language questions handled, (6) querying with other documents/database cross-references, (7) alternative full text index (Lucene/MySQL), (8) refinement based on metadata derived from initial result set, (9) meta search in separate databases, (10) refinement based on keywords derived from initial result set, (11) bypassed normal PubMed query expansion/special PubMed queries, (12) entity specific (genes/proteins), (14) Search in UMLS, (15) Search in MeSH, (16) Search in Gene Ontology, (17) Browse within Taxonomy/Ontology hierarchy, (18) Browse within identified text occurrences, (19) Query history, (20) Permanent profile, (21) Session clipboard, (23) Title, (24) Abstract, (25) external Links, (26) PMID, (27) Evidence sentence, (28) Text snippets, (29) Call external web services, (31) highlighted keywords from query, (32) highlighted biomedical entities/relations, (33) highlighted ontology concepts detected, (34) highlighted vocabulary (cluster labels/significant words), (36) Re-ranking based on concurrence of keywords, (37) Re-ranking based on concurrence of identified entities, (38) Re-ranking based on external database references or precomputed statistics, (39) Language structure (e.g. conclusive sentences), (41) Cosine similarity based, (42) based on co-authorship, (43) via author name, (44) hierarchical classification based on distance metrics, (45) hierarchical classification using taxonomies/ontologies, (46) 2D concept graph, (47) tabular statistics, (48) Call external service, (50) graphical sliders, (51) email communication, (52) social tagging, (53) special query language, (54) batch processing, (55) drag&drop GUI, (57) external markup tool, (58) import literature references from external databases curations, (59) visualization using an external tool, (60) external large scale experimental metadata used, (62) XML, (63) RDF, (64) BibTex, (65) Endnote (RIS), (67) biomedical entities (e.g. gene/proteins), (68) Taxonomy/Ontology terminology, (69) Wikipedia terminology, (71) within abstracts, (72) within sentences, (74) disambiguation for bio-entities, (75) disambiguation for taxonomy/ontology terminology, (76) disambiguation for authors, (78) is-a generalization, (80) significant strings, (81) significant taxonomy/ontology concepts, (82) expert profiles, (83) significant bio-entities, (84) textual synopsis, (87) explicit question answering, (89) question categories, (90) graphical interaction, (92) explicit hypothesis generation.

2.4.3 General search engines

Some general purpose search engines use clustering to provide an overview of search results.

Vivisimo. Vivisimo [263] displays categorization of search results in a tree structure. With the product “Clusty” Vivisimo® offers a meta search engine which groups similar results together into clusters. Clusty does not use a controlled vocabulary to index documents. Representative labels are identified from each cluster.

Powerset. Powerset 8 uses natural language processing to answer real English questions based on Wikipedia articles. The user provides a full sentence question and the system tries to parse the question. Then Wikipedia articles are parsed and relevant passages are listed as answers to the user’s question.

---

8powerset.com
2.5 Ontology-based tools in the life sciences

The biological knowledge is currently developing very fast. Ontologies are being developed to describe this knowledge. The massive amounts of information nowadays available to scientists makes it difficult to stay up to date. Various ontology-based tools have been developed to support biologists in that task. Ontologies as described in section 2.2.3 can be used in a number of ways. Rubin et al. [233] discuss the functionalities such as searching in biomedical data, exchange of data between programs, integration of information, knowledge representation, computer reasoning and natural language processing or text mining. Another new functionality of ontologies is discussed in the chapter 4, structured searching in domain knowledge. This section discusses examples of current state of the art ontology-based tools.

2.5.1 Aligning and Merging Ontologies

Many bio-ontologies have already been developed and many of these ontologies contain overlapping information. The user wants to be able to use multiple ontologies. For instance, companies may want to use community standard ontologies and use them together with their own ontologies. Ontology developers may want to use existing ontologies as the basis for new ontologies. Tools such as SAMBO [152] are used to align ontologies by defining relations between terms in different ontologies. The aligned ontologies can be merged into a new ontology.

An interesting approach for ontology alignment in the context of this work was proposed in Lambrix et al. [155]. The authors suggest to use a literature corpus classified with ontology concepts to use for the alignment of two ontologies. The idea is the a similarity measure between concepts in different ontologies can be defined based on relationships between the documents in which they are used.

2.5.2 Search in biomedical data

Some ontologies were primarily created to annotate biomedical data. The Gene Ontology [98] is used for functional annotation of known genes. The AmiGO browser [13] allows to search the GO database for genes and gene products annotated with GO terminology. The GO is also used in numerous web-based [71, 171] and standalone tools [291, 293] analysing microarray experimental results were a list of differentially expressed genes is given and a functional description of the given genes is the output. Khatri and Drăghici [141] compare 14 tools and discusses current limitations.

The NCI Thesaurus [114] indexes cancer related research data with categories such as findings, drugs, therapies, anatomy, genes, pathways, cellular and subcellular processes, proteins, and experimental organisms.

BioPrompt [56] is an ontology-based clustering tool for searching in biological databases. BioPrompt defines documents as a biological sequences plus the associated meta-data. Several ontology-based hierarchical clustering strategies offer different views on large set of database entries.

2.5.3 Data exchange

The MGED Ontology is used to describe microarray experiments. MAGE-ML is an XML grammar used for the interchange of microarray experimental data between researchers and programs [18]. BioPAX is a collaborative effort to create a data exchange format for biological pathway data. Pathway databases such as KEGG [138], BioCyc [139] and
Reactome [135] export their data in BioPAX format. Pathways in BioPAX format can be visualized with tools such as PATIKA [63], a web-based integrated environment dealing with pathways, and Cytoscape [244], a visualization software for graphs and networks.

2.5.4 Information integration

The TAMBIS ontology [259] can be used to formulate high level queries over biological entities stored in different databases. The system translates the query to source model queries which were mapped to the concept model and executes the queries on the selected databases. Wrapper for each datasource were developed manually. Information Integration using ontologies in biology and medicine is a wide field. Pérez-Rey et al. [201] discuss methods of information integration. Approaches pertaining ontologies are single conceptual schemes, were a global conceptualization exists, which covers all information. This is the approach of TAMBIS. Any change in the system might require the change of the global conceptualization. In contrast to this multiple conceptual schemes describe the semantics of different databases separately. Here the difficulty is moved to the mapping between the schemes. It is not trivial to map concepts with similar or equal meaning. OBSERVER [175] is a system which maps concepts of different domain ontologies. Hybrid approaches develop independent domain ontologies based on a common ontology defining base concepts. Ontologies of the OBO Foundry [253] are based on the common Relation Ontology [251].

2.5.5 Knowledge representation

The Foundational Model of Anatomy ontology (FMA) is a representation of the phenotypic structure of the human body [231]. It comprises 75,000 classes. In contrast to the Gene Ontology which instantiates only two types of relations the FMA defines 168 types of relations with 2.1 million instances. Mainly macroscopic anatomical structures such as cells, tissues and organs are represented. FMA is developed using Protégé and can be viewed on the web with the Foundational Model Explorer. Zhang and Bodenreider [292] used the FMA as one of two large ontologies for the development of lexical methods to map large ontologies to one another, taking into account their semantic structure as well as their terms.

The Edinburgh Mouse Atlas Project (EMAP) [38] is another effort of an anatomy ontology. It adds to the symbolic representation, the structured collection of terms each corresponding to a particular anatomical concept, an iconic representation of mouse embryos. A 3D voxel representation of 26 theiler stages of mouse developmental stages were created as well as a mapping between the textual anatomical classes and the volumes in the 3D representation. Transition relations between structures of subsequent stages allow for following the development of organs and tissues. The modeling of knowledge is a prerequisite for reasoning over it.

2.5.6 Computer reasoning

Computer reasoning uses methods to infer new facts from knowledge stored in ontologies and asserted facts. In Rubin et al. [232] the authors showed that FMA can be useful as a reference knowledge source to predict the anatomic consequences of penetrating injury. Hybrow [215] is a system to test consistency of hypotheses with observed data and prior knowledge by applying constraints and rules.
2.5.7 Textmining and Natural Language Processing

Ontology-based textmining provides the methodological basis of the work described in this thesis. Some other tools in the life sciences employ ontologies for the processing of natural language. Textpresso, introduced in section 2.4.2, is an ontology-based search engine built of scientific literature on C. elegans and selected others domains. Textpresso maintains a list of regular expressions to identify concepts of its ontology. Textpresso maintains a flat list of 101 concepts. Each concept in the ontology has its own identification algorithm.

Whatizit [Kirsch et al.] is webservice which processes any free text or list of PubMed abstracts. The user can select between textmining pipelines. The most relevant pipeline pertaining ontology-based textmining tools is the “whatizitGo” pipeline. Any given text to this pipeline is marked up with Gene Ontology cross-references. The XML output is translated into HTML containing hyperlinks to a copy of the original databases. Whatizit provides four categories of modules: (1) basic NLP modules for syntactical information, e.g. sentence splitting, part-of-speech tagging, (2) modules matching controlled vocabularies, e.g. protein names from Uniprot and GO terms (3) syntax pattern matching modules and, e.g. identification of abbreviations, definitions, mutations, (4) modules for shallow parsing based on cascaded patterns. Modules identify entities using a finite state automate build from a large set of regular expressions. The protein name “colla1” was added to the list of accepted expressions by “(COL1A1—[cC]olla1)”. The patterns “the X protein”, “the protein X”, “T domain of NP” and “NP is a protein” are used to detect explicit mentions of proteins. Mutations are detected in the text by the pattern “AA [0-9]+ AA” where AA denotes all variants of an amino acid or nucleic acid. Protein-protein interactions are detected by finding two noun phrases connected by a verb phrase containing one of 21 predefined verbs. At least one of the noun phrases must name a protein or gene name. The modules process an XML stream in a UNIX pipe manner. Each module processes the appropriate elements and passes the rest unchanged. Elements may not overlap and may not recursively contain elements of the same type. The Java components implement the Runnable interface and must read an InputStream and write an OutputStream.

2.6 Biomedical text corpora

In the recent years large databases of text became available online. Literature databases such as PubMed, Scopus, ACM, Google Scholar, Citeseer and DBLP allow for searching articles for keywords in the title, abstract or other sections of the documents mainly for the purpose of information retrieval, see section 2.4.1. Some databases offer full text versions of articles in standardized document formats such as XML (PubMed Central, BioMed Central). Such resources are highly valuable for research on textmining. Another important source for Natural Language Processing are annotated text corpora (Brown, GENIA, MedTag, PennBioIE) providing structural and semantic annotations suitable for training and testing of algorithms, see table 2.19 for a comparison of annotation corpora.

2.6.1 Literature/Citation databases

Literature databases index millions of documents for the purpose of document retrieval. The main purpose of citation databases is the identification of publications based on keyword search in the publications meta data. Mostly comprising a summary of the articles content in the form of a title sentence, a summarization of the achievements in the form of a short abstract, some keywords plus meta information about the authors, their affiliation, the publication history and references to other articles.
PubMed is the most widely used literature database in the biomedical field. It is operated by the United States National Library of Medicine (NLM) at www.pubmed.org. The number of citations in this database increase exponentially. PubMed has grown by 754,003 entries in 2007 alone. In sum PubMed contains now 17.8 million citations of which 9.6 million actually have an abstract. A citation consists of textual values for the following types of information: title (one sentence), abstract text (approx. 9 sentences), authors (on average 2.7 per citation), affiliated institution, MeSH headings, publication date and others. See table 2.17 for a list entry types in the MEDLINE format.

Figure 2.2 shows the growth of PubMed in the recent decades. PubMed publishes usage statistics. Since 1997 the numbers steadily increase. The numbers grow by 97 million per year on average, currently exceeding 871 million per year.

Entries in PubMed come from MEDLINE, the NLMs database of indexed journal citations of more than 4600 journals in the life sciences, OLDMEDLINE, printed and digitized citations from before 1966, citations in the original MEDLINE which were considered out of scope for biomedicine, citations just added and not yet manually index with MeSH headings, citations of journals covered by NLM which were published before the journals registration with NLM, and citations of journals covered in PubMedCentral.

Besides biomedicine other fields covered by PubMed are nursing, dentistry, veterinary medicine, the health care system, and the preclinical sciences. PubMed contains bibliographic citations and author abstracts from journals published in 70 countries. Abstracts date back to the mid-1960’s. Coverage is worldwide, but most records are from English-language sources or have English abstracts.

**ISI Web of Knowledge Service** is a commercial service on the web. The ISI Citation Databases contains multidisciplinary databases of bibliographic information gathered from thousands of scholarly journals. Articles can be search by subject, author, journal, and/or author address. In contrast to PubMed the full bibliography is stored with each article, so users may search the databases for cited articles or publications by co-authors. A citation index stores cross-references between publications, allowing the user to easily establish which later documents cite which earlier documents. Eugene Garfield first outlined the idea of a unified citation index to the literature of science [96]. The Science Citation Index (SCI), originally produced by the Institute for Scientific Information, now owned by Thomson Scientific, is the basis for the computation of Thomson Impact Factor (IF), which is frequently used as a measure for the importance of a journal. The impact factor of a journal is calculated based on a three-year period. It can be viewed as an approximation of the average

Table 2.17: Medline fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Affiliation</td>
<td>[AD]</td>
</tr>
<tr>
<td>Article Identifier</td>
<td>[AID]</td>
</tr>
<tr>
<td>All Fields</td>
<td>[ALL]</td>
</tr>
<tr>
<td>Author</td>
<td>[AU]</td>
</tr>
<tr>
<td>Pagination</td>
<td>[PG]</td>
</tr>
<tr>
<td>Corporate Author</td>
<td>[CN]</td>
</tr>
<tr>
<td>EC/RN Number</td>
<td>[RN]</td>
</tr>
<tr>
<td>Entrez Date</td>
<td>[EDAT]</td>
</tr>
<tr>
<td>Filter</td>
<td>[FILTER]</td>
</tr>
<tr>
<td>First Author Name</td>
<td>[1AU]</td>
</tr>
<tr>
<td>Full Author Name</td>
<td>[FAU]</td>
</tr>
<tr>
<td>Full Investigator Name</td>
<td>[FIR]</td>
</tr>
<tr>
<td>Grant Number</td>
<td>[GR]</td>
</tr>
<tr>
<td>Investigator</td>
<td>[IR]</td>
</tr>
<tr>
<td>Issue</td>
<td>[IP]</td>
</tr>
<tr>
<td>Journal Title</td>
<td>[TA]</td>
</tr>
<tr>
<td>Language</td>
<td>[LA]</td>
</tr>
<tr>
<td>Last Author</td>
<td>[LASTAU]</td>
</tr>
<tr>
<td>MeSH Date</td>
<td>[MHDA]</td>
</tr>
<tr>
<td>MeSH Major Topic</td>
<td>[MAJR]</td>
</tr>
<tr>
<td>MeSH Subheadings</td>
<td>[SH]</td>
</tr>
<tr>
<td>MeSH Terms</td>
<td>[MB]</td>
</tr>
<tr>
<td>NLM Unique ID</td>
<td>[JID]</td>
</tr>
<tr>
<td>Other Term</td>
<td>[OT]</td>
</tr>
<tr>
<td>Personal Name</td>
<td>[PN]</td>
</tr>
<tr>
<td>Subset</td>
<td>[SB]</td>
</tr>
<tr>
<td>Substance Name</td>
<td>[NM]</td>
</tr>
<tr>
<td>Title</td>
<td>[TI]</td>
</tr>
<tr>
<td>Title/Abstract</td>
<td>[TIAB]</td>
</tr>
<tr>
<td>Transliterated Title</td>
<td>[TT]</td>
</tr>
<tr>
<td>UID</td>
<td>[PMID]</td>
</tr>
<tr>
<td>Location ID</td>
<td>[LID]</td>
</tr>
</tbody>
</table>

**PubMed** is the most widely used literature database in the biomedical field. It is operated by the United States National Library of Medicine (NLM) at www.pubmed.org. The number of citations in this database increase exponentially. PubMed has grown by 754,003 entries in 2007 alone. In sum PubMed contains now 17.8 million citations of which 9.6 million actually have an abstract. A citation consists of textual values for the following types of information: title (one sentence), abstract text (approx. 9 sentences), authors (on average 2.7 per citation), affiliated institution, MeSH headings, publication date and others. See table 2.17 for a list entry types in the MEDLINE format.

Figure 2.2 shows the growth of PubMed in the recent decades. PubMed publishes usage statistics. Since 1997 the numbers steadily increase. The numbers grow by 97 million per year on average, currently exceeding 871 million per year.

Entries in PubMed come from MEDLINE, the NLMs database of indexed journal citations of more than 4600 journals in the life sciences, OLDMEDLINE, printed and digitized citations from before 1966, citations in the original MEDLINE which were considered out of scope for biomedicine, citations just added and not yet manually index with MeSH headings, citations of journals covered by NLM which were published before the journals registration with NLM, and citations of journals covered in PubMedCentral.

Besides biomedicine other fields covered by PubMed are nursing, dentistry, veterinary medicine, the health care system, and the preclinical sciences. PubMed contains bibliographic citations and author abstracts from journals published in 70 countries. Abstracts date back to the mid-1960’s. Coverage is worldwide, but most records are from English-language sources or have English abstracts.

**ISI Web of Knowledge Service** is a commercial service on the web. The ISI Citation Databases contains multidisciplinary databases of bibliographic information gathered from thousands of scholarly journals. Articles can be search by subject, author, journal, and/or author address. In contrast to PubMed the full bibliography is stored with each article, so users may search the databases for cited articles or publications by co-authors. A citation index stores cross-references between publications, allowing the user to easily establish which later documents cite which earlier documents. Eugene Garfield first outlined the idea of a unified citation index to the literature of science [96]. The Science Citation Index (SCI), originally produced by the Institute for Scientific Information, now owned by Thomson Scientific, is the basis for the computation of Thomson Impact Factor (IF), which is frequently used as a measure for the importance of a journal. The impact factor of a journal is calculated based on a three-year period. It can be viewed as an approximation of the average
number of citations in a year, given to those papers in a journal that were published during the two preceding years.

\[ \text{ImpactFactor}(\text{year}_i) = \frac{\text{indexed citations}}{\text{citable items}} \]

were \text{indexed citations} is the number of times articles published in \text{year}_{i-1} and \text{year}_{i-2} were cited in indexed journals during \text{year}_i and \text{citable items} is the number of “citable items” published in \text{year}_{i-1} and \text{year}_{i-2}.

**SCIImago Journal Rank** is based on the transfer of prestige from a journal to another one. The indicator is computed on the data of the Scopus database. In Falagas et al. [77] the authors compare the Thomson Impact Factor and the SCIImago Journal Rank. The SCIImago Journal Rank indexes twice as much journals, 12,751, as the Thomson Impact Factor. In contrast to Thomson it weights external references according to their impact. Thus publications referenced by important other publications become more important.

**Citeseer** is a freely accessible citation database with focus on the literature in computer and information science. The citation index is automatically computed. Citeseer automatically extracts and provides metadata from all indexed articles. When possible CiteSeer automatically links to other metadata resources such as DBLP, a bibliography service for computer science, and the ACM Digital Library, journals, magazines, and conference proceedings of members of the Association for Computing Machinery. The full source code of CiteSeer is available at no cost for non-commercial use. CiteSeer, indexes 1,077,967 articles with 20,328,278 citations.

**Google Scholar** mimics the Science Citation Index (SCI) on the web. It is a free citation index based on selected “trusted scholarly sources” [112]. In Noruzi [190] the author lists advantages and disadvantages of this web service currently being in beta testing phase. Among the advantages is the fact that it is international, non-biased to journals nor to subjects, freely accessible and multidisciplinary. Through collaboration with libraries the search index is based on the full texts even of selected books. To retrieve the documents the user still needs a subscription to the each journal. Some disadvantages are that it is language biased, it can not handle all citation style inconsistencies, it does not disambiguate author names, many scholarly resources are still not covered and it has no subject indexing. Mayr and Walter [172] reported a coverage of 72-84% of articles from selected journals. Open access journals covered by the normal Google web search are not well covered. Nevertheless is has become a popular tool for searching scientific publications.

### 2.6.2 Full text corpora

In contrast to citation databases containing only meta data of publications a full text corpus contains all text sections of a publication plus all supplementary materials such as tables, figures and other data files provided with the article.

The usage of full text corpora for textmining has some clear advantages: The full text contains all assumptions and conclusions made by the author. The abstract text will omit details of the argumentation. On the contrary it can be assumed that facts stated in the abstract section are the most important conclusions. It is assumed that findings stated in the abstract of a peer reviewed scientific publication are based on profound methods and therefore reflect the current state of the scientific knowledge. However full text corpora
include the abstract text as well. The most limiting factor when using full text corpora is of course the number of available publications. While full text corpora contain thousands of articles citation databases contain millions of articles.

Currently one can see a trend that open-access publishing becomes more popular. Journals such as PLoS (Public Library of Science) and BMC (BioMed Central) offer free access to the full articles at no charge for the reader. The cost are covered by the publishing author/institution. This scheme enables journals to cover their editorial cost while offering free access to all research results, a desirable situation when research is funded by tax payers. The Directory of Open Access Journals (DOAJ) currently lists 3340 open access journals.

PubMed Central is the U.S. National Institutes of Health (NIH) free digital archive of biomedical and life sciences journal literature. It is online available at [pubmedcentral.nih.gov](http://pubmedcentral.nih.gov). Participation by publishers in PubMed Central (PMC) is voluntary, although participating journals must meet certain editorial standards. A journal must provide PubMed Central the full text of articles in an XML (eXtensible Markup Language) or SGML (Standard Generalized Markup Language) format that conforms to an acceptable journal article DTD (Document Type Definition). PMC does not accept articles in HTML format. A journal may delay release of its full text in PMC for some period of time after publication. For users who are interested in just the XML, an FTP service also has a single zip file containing the XML files for all the articles in the open access subset. The open access corpus contains 5,000 full text articles of 525 journals. The full DTD is online available at [dtd.nlm.nih.gov/publishing/](http://dtd.nlm.nih.gov/publishing/).

Biomed Central is the U.S. National Institutes of Health (NIH) free digital archive of biomedical and life sciences journal literature. It is online available at [biomedcentral.com](http://biomedcentral.com). All original research articles published by BioMed Central are made freely and permanently accessible online immediately upon publication. It is online available at [biomedcentral.com](http://biomedcentral.com). All research articles published by BioMed Central are archived without delay in PubMed Central. A single ZIP-compressed file containing all the full-text XML files is available via FTP. The open access corpus contains 8683 full text articles of 200 journals. A typical entry contains bibliometric meta data such as authors, affiliations, the document type, keywords and an abstract. The body part is divided into sections each containing a title and possibly subsections. The full DTD is online available at [biomedcentral.com/xml/](http://biomedcentral.com/xml/).

HighWire Press hosts a repository of high impact, peer-reviewed content, with 1149 journals and 4,750,743 full text articles from over 140 scholarly publishers. HighWire-hosted publishers have collectively made 1,875,091 articles free among them are 71 of the 200 most-frequently-cited journals. HighWire Press is online available at [highwire.stanford.edu](http://highwire.stanford.edu).

Text corpora merely contain articles and their supplementary material in PDF or XML format. The textual content is not more structured as to the markup of sections or paragraphs mostly in a proprietary format or XML grammar. In contrast to this annotated corpora add structural or semantic markups to the raw text of articles.

### 2.6.3 Annotation corpora

Linguists have created a number of annotated corpora to solve some problems of natural language understanding such as: text segmentation, coreferences, named entity recognition and word sense disambiguation. The discipline of statistical natural-language processing uses quantitative methods to solve some difficult problems in language understanding. Especially texts with long sentences with highly ambiguous terminology and complex structures are
difficult to be parsed and fully understood by NLP systems. Stochastical, probabilistical and statistical methods are used to overcome those problems which currently are believed not solvable by parsing with formal grammars.

Manually annotated corpora differ in various aspects. Besides corpus size, language coverage, free availability and domain subject some more technical aspects can be compared. A group of corpora was build for general research on natural language [169, 235, 170] and cover e.g. news language, poetry as well as technical speech. Other corpora focus on specific research domains and are biased toward a scientific language [193, 167, 252]. General corpora can be used to reveal domain specific terminology and linguistic structures in such domain specific corpora. Table 2.19 compares publication year, corpus size, subject domain and corpus composition of 15 freely available corpora. Some corpora are build using other corpora. The last column in the table lists which corpus has inclusions of other corpora.

Other differences between corpora are of more technical nature. A corpus can be encoded in a simple text file using tabulators or new line characters to separate words and annotations. In contrast to this approach XML-based corpora define an encoding scheme with a document-type definition (DTD). As an alternative to proprietary annotation schemes international standards for linguistic annotations are being developed [131].

Two widely used standards for document annotations are DocBook and the Text Encoding Initiative (TEI). TEI was originally conceived of as a means of representing previously existent documents, rather than creating them from scratch. Docbook was originally thought of as a publishing DTD, for converting between digital and print versions of newly authored documents. Table 2.8 shows and example for DocBook and table 2.9 for an TEI document. Both approaches do not offer specialized vocabulary and thus offer extension of their grammars. Scalable Vector Graphics (SVG), and MathML, for mathematical formulas, are examples for specialized document content. Thompson and McKelvie [266] compare inline [6, 193] and standoff [167, 252] annotations used in corpora. While inline annotations are inserted into the text standoff annotations are separated from the original text and refer to character positions the original text. With standoff annotations overlapping annotation hierarchies can exist in parallel. The original text may be distributed separately from the annotation e.g. with copyright conditions. A disadvantage for standoff annotations is the introduction of errors in case the original document is accidentally changed without maintaining the external annotations.

Annotated corpora differ in the level of structural annotations made by humans. Table 2.21 compares manual annotated corpora which structural annotations were made by curators.

Sentence segmentation is the task of finding the boundaries of a sentence in the text. It is an important initial step in structuring text as there are more intrasentinal linguistic relations than intersentinal relations [178]. The task is not entirely trivial as sentences may embed other sentences. Delimiting characters may be ambiguous, e.g. colon and semi-colon may or may not separate sentences. Punctuations in abbreviations may be misinterpreted as a sentence boundary.

Tokenization identifies the basic units of a sentence, e.g. word, punctuation symbols and brackets. In the biomedical domain tokenization is difficult as many labels of chemical contain brackets and punctuations which can be misinterpreted with sentence punctuations. Examples are:

- 5(S)-hydroxyperoxy-6E,8Z,11Z,14Z-icosatetraenoic acid
2.6. BIOMEDICAL TEXT CORPORA

Figure 2.8: An example of a DocBook document.

Figure 2.9: A fragment of a GENIA document encoded with TEI.
CHAPTER 2. BACKGROUND

- (Na\(^+\) + K\(^+\))\(\text{ATPase}\)
- 2,3,7, 8-tetrachlorodibenzo-p-dioxin
- 2-(4-acetoxyphenyl)-2-chloro N-methyl-ethylammonium
- 2-amino-6-methyldipyrido(1,2-a:3',2'-d)imidazole
- 2,2',4,5,5'-Cl\textsubscript{5}
- 1, 2-bis (o-aminophenoxy) ethane N, N, N', N'-tetraaceticacid tetra(acetomethoxy) ester

**Part-of-speech tagging** adds grammatical information to each token. In English language for example the word “lead” may be a noun but also a verb. Text corpora are manually annotated with grammatical forms to train machine learning algorithms to identify the correct grammatical form. Most words are assigned to be nouns, verbs, adjectives. Table 2.22 shows a complete list of tags defined for the Penn Treebank corpus. Rule-based taggers (Brill) use such corpora to iteratively learn linguistic rules to minimize errors on the manually annotated tags. Statistical taggers use such corpora to learn the probabilities for a POS tag depending on the previous words.

**Anaphora resolution** is the detection of instances of an expression referring to another expression. An example is “Ras and its GTPase activating proteins”.

**Acronyms** are abbreviations that are formed using the initial components in a phrase or name. An example is “a patient with the acquired immune deficiency syndrome (AIDS)”.

**Synonyms** are different words with identical or similar meaning. Examples in the biomedical domain are **programmed cell death** (synonym: **apoptosis**) and **Ganglion Cysts** (synonym: **Myxoid Cyst**).

**Chunks** are non-overlapping sequences of words forming a phrase. Noun phrases might consist of a noun and its adjective and can be linked by prepositions. Other types of phrases marked up in corpora are verbal phrases, predicate adjectival phrases and subordinate clause markers.

**Treebanking** annotates the syntactic structure of a sentence. An example from Penn Treebank WSJ is shown in figure 2.10. Noun, verb and prepositional phrases are marked up using a scheme defined in Marcus et al. [170].

Table 2.21 compares manually annotated corpora relevant for general and biomedical natural language processing. In column 3 the type of entities annotated for each corpus is listed. The most prominent in the biomedical domain are gene and protein names. Other concepts in the biomedical domain annotated frequently in corpora are species, protein structures, residues, substances and measurements. There exist biomedical ontologies containing each of that concepts. However, very few corpora take the approach of annotating ontologies at all.
2.6. BIOMEDICAL TEXT CORPORA

Figure 2.10: A sentence with treebanking information. Sentences (S), noun phrases (NP), verb phrases (VP)

GENIA Ohta et al. [193] define its own small ontology and has set up guidelines for curators how to annotate them. GENIA Corpus Version 3.0x consists of 2000 abstracts. The base abstracts are selected from the search results with keywords (MeSH terms) Human, Blood Cells, and Transcription Factors. Figure 2.11 shows the GENIA ontology. Only the leaf concepts of the ontology are being annotated by the curators. The corpus is encoded in XML format encoding sentence boundaries, term boundaries, term classifications, semi-structured coordinated clauses and recovered ellipsis in terms. All the abstracts and their titles have been marked-up with biologically meaningful terms, and these terms have been semantically annotated with descriptor from the ontology.

FetchProt Franzén and Oppenheimer [89] focus on a very small subset of the Gene Ontology and annotates GO concepts related to tyrosine kinase, namely TYROSINE KINASE ACTIVITY, TRANSMEMBRANE RECEPTOR PROTEIN TYROSINE KINASE ACTIVITY, NON-MEMBRANE SPANNING PROTEIN TYROSINE KINASE ACTIVITY and JANUS KINASE ACTIVITY. The corpus consists of free access documents, mostly concered with the description of lab experiments on tyrosine kinase activity. Some documents not concered with such experiments were included to test the FetchProt system for the ratio of false positives. The corpus contains also mentions of mutations of proteins related to tyrosine kinase activity. The text files have been produced by saving the PDF-files as text. The content of an article in plain text is interspersed with annotations (xml-style tags) surrounding specific semantic elements. Figure 2.12 shows a FetchProt annotation made for the PubMed document with the PMID 9582366. An entire paragraph was marked describing a NON-MEMBRANE SPANNING PROTEIN TYROSINE KINASE ACTIVITY.

None of the freely available manually annotated corpora provide a large enough test set of mentions for the Gene Ontology. GENIA and FetchProt annotate concepts related
Figure 2.11: The GENIA ontology (Version 3.01) is a concept tree. The GENIA corpus is only annotated with the leaf concepts printed bold in this figure.
We next examined whether Src or Fak catalyze the tyrosine phosphorylation of SHPS-1 in vitro. Incubation of a GST fusion protein containing the cytoplasmic domain of SHPS-1 (GST-SHPS-1-cyto) with Src kinase immunoprecipitated from Rat-1 cells resulted in the phosphorylation of both GST-SHPS-1-cyto (Fig. 5A) and Src (Fig. 5C).

Incubation of a GST fusion protein containing the cytoplasmic domain of SHPS-1 (GST-SHPS-1-cyto) with Src kinase immunoprecipitated from Rat-1 cells resulted in the phosphorylation of both GST-SHPS-1-cyto (Fig. 5A) and Src (Fig. 5C).

Figure 2.12: An XML fragment of a structural template for PMID: 9582366 annotated manually for FetchProt. The experiment describes a non-membrane spanning protein tyrosine kinase activity (GO:0004715).

to concepts in GO. A possible usage of the two corpora prerequisites the mapping of the annotated concepts two actual concepts of the current version in GO. FetchProt used at least one outdated term which was changed in GO in one of the more recent versions. The GENIA concepts are much more general concepts as most concepts known from GO. For a subset of the GENIA concepts a rough mapping is possible. Another issue is that GENIA does not apply the True Path Rule known from GO. The none leaf nodes in the GENIA ontology can be seen as hierarchical modifiers to the leaf concepts. The ancestors are not concepts in the same sense as the leaf nodes. In the GENIA corpus only the leafs are annotated by curators. The part of GO used in FetchProt is with 4 nodes rather small and can therefore not serve as a test corpus for ontology-based term identification. In contrast to the currently available text corpora there exist database curation project which make use of larger portions of ontologies.

2.6.4 Database curation projects

In order to progress toward to major goal of the exploration of the order and logic of genetic programs [35] enormous efforts are being made to collect information on functionality of biological units encoded in the genomes of organisms. For many organisms entire genomes have been sequenced [107, 54, 102, 76], see table 6.1. The next step is to understand the genetic program. For this the genome is structured into subunits. For the human genome for example the 3 billion base pairs are grouped into 23 chromosomes, the chromosomes are classified into gene encoding areas and non-encoding areas. The next task is to identify begin and end of subsequences encoding specific genes. It is known that genes are translated into proteins. In the normal case organisms poses only genomes encoding proteins which are somehow useful to the organism. That means each protein serves some useful function. The understanding of this functionality of each protein is of paramount importance to understand the genetic program. This task can only be solve by a hugh number of researches. A prominent database storing such knowledge is UniProt.

GOA The Gene Ontology Annotation (GOA) project is an effort of the European Bioinformatics Institute. Its goal is the functional annotation of proteins in the protein sequence database UniProt. That is the textual description of the function of a protein in an organism. The goal is also to provide experimental evidence described in research papers for each
annotation. One problem is the usage of different terminology among biologists to describe protein functionality. This problem has been addressed by the creation of the Gene Ontology resource [13], a dynamic controlled vocabulary that can be applied to all organisms even as knowledge of gene and protein roles in cells is accumulating and changing.

As curators of the GOA project assign not only functions defined in the Gene Ontology to proteins in UniProt but also provide textual evidence in research articles GOA can serve as a source for benchmark data for the identification of protein functions from text.

Reduced recall. A significant problem when using GOA as a benchmark corpus is that GOA curators use the full text of research articles to create the annotations. A term identification algorithm only presented with the abstract texts is therefore limited in terms of recall of correctly identified concepts from the molecular functions branch of the Gene Ontology.

Reduced precision. Another problem for the usage of GOA as a benchmark corpus is that GOA curators do annotate one protein at a time. It is not their goal to annotate an entire publication text, as is the case of text corpora, see previous section 2.6.3. That means literature abstracts of publications associated with GOA annotations contain mentions of GO concepts, but curators do not intend to mark each occurrence of a GO concept in the abstract. Therefore a benchmark measuring the precision of a term identification algorithm based on GOA annotations must be aware of this.

GOA currently integrates data from 23 annotation databases. Each is using the Gene Ontology terminology to annotate molecular function, biological processes and cellular components. Table 2.24 shows a typical GOA record.
<table>
<thead>
<tr>
<th>Corpus</th>
<th>Published</th>
<th>Size</th>
<th>Domain</th>
<th>Composition</th>
<th>Includes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brown</td>
<td>1964</td>
<td>500 texts (2000 words each)</td>
<td>press reportage, bibliography, technical texts, fiction</td>
<td>manually selected</td>
<td></td>
</tr>
<tr>
<td>Susanne</td>
<td>1992</td>
<td>64 texts from Brown corpus</td>
<td>press reportage, bibliography, technical texts, fiction</td>
<td>subset of Brown corpus</td>
<td></td>
</tr>
<tr>
<td>Penn Treebank</td>
<td>1993</td>
<td>4.9 million POS tagged tokens 2.9 million tokens with treebanking</td>
<td>general speech</td>
<td></td>
<td>WSJ, Brown</td>
</tr>
<tr>
<td>OHSUMED</td>
<td>1994</td>
<td>16140 pmids</td>
<td>Primary care</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MedStract</td>
<td>2001</td>
<td>320 abstracts</td>
<td>Medline</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Yapex</td>
<td>2002</td>
<td>200 abstracts</td>
<td>development and evaluation</td>
<td>PubMed Query: &quot;protein bind-</td>
<td>overlaps with GENIA</td>
</tr>
<tr>
<td>MEDCo</td>
<td>2004</td>
<td>228 abstracts</td>
<td>Medline</td>
<td>GENIA based</td>
<td></td>
</tr>
<tr>
<td>GENETAG-05</td>
<td>2004</td>
<td>20000 sentences</td>
<td>Medline</td>
<td>documents similar to documents with known genes</td>
<td></td>
</tr>
<tr>
<td>Caderige</td>
<td>2004</td>
<td>functional genomics of a model bacterium &quot;Bacillus subtilis&quot;</td>
<td></td>
<td>PubMed Query: &quot;Bacillus subtilis&quot; AND &quot;transcription&quot;</td>
<td></td>
</tr>
<tr>
<td>FetchProt</td>
<td>2005</td>
<td>190 articles</td>
<td>proteomics (tyrosine kinase)</td>
<td>190 full text scientific articles on tyrosine kinase</td>
<td></td>
</tr>
<tr>
<td>MedTag</td>
<td>2005</td>
<td>37000 sentences</td>
<td>molecular biology and clinical medicine</td>
<td>updated and unified GENETAG corpus, ABGene corpus and MedPost corpus</td>
<td>GENETAG, ABGene, MedPost</td>
</tr>
<tr>
<td>PennBioIE</td>
<td>2006</td>
<td>2257 abstracts</td>
<td>cancer genomics and drug developement</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2.19: Comparison of annotation corpora, part 1
## RESULTS

<table>
<thead>
<tr>
<th>Annotation Corpora</th>
<th>Objectives</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brown</td>
<td>NER, RE, AN</td>
<td>Text, POS, BI</td>
</tr>
<tr>
<td>FetchProt</td>
<td>NER, RE, AN</td>
<td>Text, POS, BI</td>
</tr>
<tr>
<td>PASTA Corpus</td>
<td>NER, RE, AN</td>
<td>Text, POS, BI</td>
</tr>
<tr>
<td>GENIA</td>
<td>NER, RE, AN</td>
<td>Text, POS, BI</td>
</tr>
<tr>
<td>GENETAG-05</td>
<td>NER, RE, AN</td>
<td>Text, POS, BI</td>
</tr>
<tr>
<td>Caderige</td>
<td>NER, RE, AN</td>
<td>Text, POS, BI</td>
</tr>
<tr>
<td>PennBioIE</td>
<td>NER, RE, AN</td>
<td>Text, POS, BI</td>
</tr>
<tr>
<td>OSHA-NER</td>
<td>NER, RE</td>
<td>Text, POS, BI</td>
</tr>
<tr>
<td>OSHA-RE</td>
<td>NER, RE</td>
<td>Text, POS, BI</td>
</tr>
<tr>
<td>Susanne</td>
<td>NER, RE, AN</td>
<td>Text, POS, BI</td>
</tr>
<tr>
<td>MedStract</td>
<td>NER, RE, AN</td>
<td>Text, POS, BI</td>
</tr>
<tr>
<td>Penn Treebank</td>
<td>NER, RE</td>
<td>Text, POS, BI</td>
</tr>
<tr>
<td>OHSUMED</td>
<td>NER, RE, AN</td>
<td>Text, POS, BI</td>
</tr>
</tbody>
</table>

**Table 2.21:** Comparison of annotation corpora, part 2

---

**CHAPTER 2. BACKGROUND**
Table 2.22: The Penn Treebank Part-of-Speech set.

<table>
<thead>
<tr>
<th>Field</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB</td>
<td>UniProtKB</td>
<td>Database from which annotated entry has been taken.</td>
</tr>
<tr>
<td>DB_Object_ID</td>
<td>A1L331</td>
<td>A unique identifier in the DB for the item being annotated.</td>
</tr>
<tr>
<td>DB_Object_Symbol</td>
<td>A1L331_MOUSE</td>
<td>A (unique and valid) symbol to which DB_Object_ID is matched.</td>
</tr>
<tr>
<td>GO ID</td>
<td>GO:0005829</td>
<td>The GO identifier for the term attributed to the DB_Object_ID.</td>
</tr>
<tr>
<td>Evidence</td>
<td>PMID:10219055</td>
<td>Reference cited to support the annotation. e.g. traceable author statement</td>
</tr>
<tr>
<td>Aspect</td>
<td>TAS</td>
<td>P (biological process), F (molecular function) or C (cellular component)</td>
</tr>
<tr>
<td>DB_Object_Name</td>
<td>Pam: Pam protein</td>
<td>Name of gene or gene product</td>
</tr>
<tr>
<td>Synonym</td>
<td>Gene symbol [or other text]</td>
<td></td>
</tr>
<tr>
<td>DB_Object_Type</td>
<td>protein</td>
<td>Type of entity being annotated</td>
</tr>
<tr>
<td>Taxon_ID</td>
<td>taxon:10090</td>
<td>Identifier for the species being annotated.</td>
</tr>
<tr>
<td>Date</td>
<td>0811-00-20</td>
<td>The date of last annotation update</td>
</tr>
<tr>
<td>Assigned_By</td>
<td>MGI</td>
<td>Attribute describing the source of the annotation.</td>
</tr>
</tbody>
</table>

Table 2.24: A typical GOA record.
2.7 Biomedical textmining methodologies

The biomedical literature as a whole is considered as a large unstructured knowledge database. Like biologist refer to the entire set of genes of an organism as “the genome” the biomedical textmining community refers to the entire literature as “the bibliome”. Like its biological counterpart the bibliome is highly unstructured and one needs a specialized machinery to interpret it. In the case of the genome those are molecular machines reading the genome and translating parts of it into functional units, the proteins. In the case of the bibliome those machines are traditionally the readers of research articles transforming information encoded in the articles into knowledge by accumulating and structuring facts. The bibliome, in contrast to the genome, is growing exponentially. It encodes wrong, contradictory and redundant information. While it is currently not possible to decode it entirely methods were developed to structure its content.

2.7.1 Lexical methods

Natural language documents can be structured into articles, sections, paragraphs, sentences, phrases and words. A word is the smallest linguistic information token in a text. Sometimes single characters form a token.

**Tokenization.** A naive algorithm splits the characters of a text at white spaces. This strategy will produce medium results and will introduce problems at later stages. Tokens are not always separated by white space characters. A period at the end of a sentence does not belong to the last token in a sentence. However, a period at the end of a abbreviation belongs to the token of the short form of the term. If the abbreviation is at the end of a sentence the period has two syntactical meanings. Biomedical text contains a huge number of abbreviations, with and without periods.

Measuring activities in tobacco control across the E.U. (PMID: 16722613)

The above example can be solved by first detecting sentence boundaries and later tokenize the sentences. Apostrophes may or may not belong to a token. In case of a possessive marker it must be separated from the word.

Network biology: understanding the cell’s functional organization. (PMID: 14735121)

In biochemistry apostrophes are frequently used in chemical identifiers.

Total synthesis of natural (+)-(2'S,3'R)-zoapatanol. (PMID: 15300507)

Catalytic sites for 3’ and 5’ incision of Escherichia coli nucleotide excision repair are both located in UvrC. (PMID: 10671556)

In both cases the apostrophes belong to the token forming the chemical identifier. The above example is also a case when hyphens do not delimit tokens. In contrast to the normal case were two or more words a concatenated by a hyphen.

The DNA-binding properties of these complexes were investigated. (PMID: 8774892)

Tokens may contain white space characters, brackets, periods and slashes as well as numbers and operation symbols. In general such graphemes mark delimiters for tokens.
The karyotypes, as assessed by banding, FISH, and STRs/STSs analyses, were 46, X,r(Y).ish r(Y)/(p11.3q11.222)(SRY+,DYZ3+) and 46, X,+r(Y)/45, X.ish r(Y)/(p 11.2q11.2)(Xp/Yp,SRY+,DYZ3+) respectively. (PMID: 17375527)

Sometimes sentences are structured by brackets. However, it can be seen from the previous example brackets can also be part of tokens. In PubMed one can find instances of recursive bracketing as well as false bracketing. This makes correct tokenization far from trivial.

Effects of sulfhydryl reagents (5,5'-dithiobis(2-nitrobenzoic acid) and N-ethylmaleimide) and potassium ferricyanide on the activities of branched-chain 2-oxoacid dehydrogenase complex and its kinase were studied. (PMID: 1772438)

The passage above is an example of recursive bracketing in a literature abstract.

'Multicentric giant cell tumor (GCTs)... with appropriate factor replacement.' (PMID: 17286774)

The abstract of the above article begins with two single quotes and ends with one corresponding double quote.

Grefenstette and Tapanainen [106] propose to use regular expressions for handling tokenization. Regular expressions however can not handle recursive bracketing. As many chemical names contain brackets the problem can not be ignored in the context of biomedical texts.

Sentence segmentation. A widely made assumption in biomedical Information Extraction is that sentences form a linguistically meaningful unit for relation extraction. Some approaches define a relation between entities if the co-occur in one sentence. Others take the syntactical structure and the semantic of a sentence into account. In any case it is most important to identify the boundaries of a sentence correctly to avoid false results at later stages of the analysis. An obvious simple approach to sentence segmentation is to split the text at periods followed by one or more white space characters and an upper case letter. Kiss and Strunk [144] report a 7.43% error rate on such a baseline algorithm.

CELLULAR COMMUNICATIONS INC. sold 1,550,000 common shares at $21.75 each yesterday, according to lead underwriter L.F. Rothschild & Co. (cited from WSJ 05/29/1987)

Several problems are introduced by this simple algorithm. Periods are also used to mark abbreviations, initials, ordinal numbers, and ellipses. In the literature several approaches are described to handle problems with sentence segmentation. Rule-based systems use fixed regular expressions and dictionaries [136, 106]. Supervised machine-learning approaches train a classifier on manually annotated text corpora [196, 100, 228, 226, 257]. Unsupervised machine-learning approaches dont need manually crafted rules or manually annotated corpora [177]. They can be quickly adopted to new domains. The authors in Kiss and Strunk [144] compare their system to the above mentioned systems and report very low error rates. Their algorithm is not depend on any additional resources besides the corpus it is supposed to segment into sentences.

Morphological analysis. Stemming, also named lemmatization, is the task of mapping tokens to a canonical form. Most words in texts can be used in an inflected form, not changing the part-of-speech, e.g. plural forms “cells vs. cell”, past-tense “regulated vs. regulate”, gerund form of verbs “inhibiting vs. inhibit”, or derived form, potentially changing the
part-of-speech, e.g. “environment vs. environmental”. The meaning of the words however is not changed in these cases, they belong to the same lexeme. It is desirable to “normalize” words to their simplest form, the lemma, without changing their meaning. This helps for comparing the content of documents on the lexical level. Two approaches can be identified for stemming: rule-based systems and systems based on lexicons.

The algorithm described in Porter [206] does not use any lexion. It applies a series of rules to each token in order to bring it to its base form. The algorithm is widely used in Information Retrieval systems. In Porter [208] the same author introduces a simple programming language, Snowball, for stemming algorithms. He also points out that many erroneous implementations of his original algorithm circulate.

In contrast to this programmatic approach there exist lexicons with all morphological variants and their grammatical features. With lexicons stemming becomes the task of mapping the variants to its stem by lookup. Such lexicons suffer from the fact that language is changing and therefore a lexicon may become outdated or incomplete. The dynamic nature of languages can be handle with a more recent approach. Goldsmith [103] describes a system which is only presented with a text corpus of any European language and uses unsupervised learning to find the morphological system of the corpus.

**Named Entity Recognition.** The recognition of textual mentions of concepts in natural text is a prerequisite to many subsequent tasks in Information Retrieval, Information Extraction and Knowledge Retrieval. Named Entity Recognition in biomedical text mining is difficult for several reasons. Primarily because there exist no common vocabulary in a quickly developing research area. While some efforts are being made to develop common terminologies [13, 253] the dynamic nature of language is clearly visible in the life sciences. The frequent use of abbreviations, synonyms and homonyms make the mapping of text mentions to defined concepts ambiguous. Another problem is the identification of textual boundaries of text mentions. Even human curators disagree on specifying the beginning and the end of named entities in texts.

Alternative interpretations of the above protein mention are: “CD6”, “human CD6” and “human CD6 T cell surface protein” all valid ranges for referring to a protein described in UniProt with the accession number P30203. Recently the biomedical domain raised interest in this topic previously studied in the newswire domain [108]. Early publications [91, 189] were followed by conferences [108, 202, 120, 118] and workshops [123, 148] specifically devoted to biomedical Named Entity Recognition. Leser and Hakenberg [159] give a survey on NER methodologies in the biomedical domain. Methodologies employed for NER include Dictionary-Based approaches [132, 194], Rule-Based approaches [91, 210], Alignment-Based approaches [150, 62, 79], Machine Learning approaches [189, 294, 258] and hybrid approaches [254, 212].

Named entity recognition is closely related to Term Recognition. The difference is that a recognized named entity corresponds to one defined semantic class such as proteins or genes. Usually authors mention a large and continuously growing number of instances of this semantic in the literature. Term Recognition aims at identifying mentions of a potentially large number of semantic classes with relatively few lexical variations used by authors in the text. The same challenges as for Named Entity Recognition apply to Term Recognition. However it might be necessary to fine tune the recognition of each descriptor in an ontology or taxonomy. Term Recognition is a semantic method discussed later in section 2.7.3.
2.7. BIOMEDICAL TEXTMINING METHODOLOGIES

Table 2.25: The part of speech tags for the sentence “Pockets of higher lead poisoning rates continue to be a problem in some geographic areas.”.

<table>
<thead>
<tr>
<th>Token</th>
<th>POS tag</th>
<th>lemma</th>
<th>category</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pockets</td>
<td>NNS</td>
<td>pocket</td>
<td>plural noun</td>
</tr>
<tr>
<td>of</td>
<td>IN</td>
<td>of</td>
<td>preposition</td>
</tr>
<tr>
<td>higher</td>
<td>JJR</td>
<td>high</td>
<td>comparative adjective</td>
</tr>
<tr>
<td>lead</td>
<td>NN</td>
<td>lead</td>
<td>noun</td>
</tr>
<tr>
<td>poisoning</td>
<td>NN</td>
<td>poisoning</td>
<td>noun</td>
</tr>
<tr>
<td>rates</td>
<td>NNS</td>
<td>rate</td>
<td>plural noun</td>
</tr>
<tr>
<td>continue</td>
<td>VVP</td>
<td>continue</td>
<td>verb</td>
</tr>
<tr>
<td>to</td>
<td>TO</td>
<td>to</td>
<td>the preposition “to”</td>
</tr>
<tr>
<td>be</td>
<td>VB</td>
<td>be</td>
<td>verb</td>
</tr>
<tr>
<td>a</td>
<td>DT</td>
<td>a</td>
<td>determiner</td>
</tr>
<tr>
<td>problem</td>
<td>NN</td>
<td>problem</td>
<td>noun</td>
</tr>
<tr>
<td>in</td>
<td>IN</td>
<td>in</td>
<td>preposition</td>
</tr>
<tr>
<td>some</td>
<td>DT</td>
<td>some</td>
<td>determiner</td>
</tr>
<tr>
<td>geographic</td>
<td>JJ</td>
<td>geographic</td>
<td>adjective</td>
</tr>
<tr>
<td>areas</td>
<td>NNS</td>
<td>area</td>
<td>plural noun</td>
</tr>
<tr>
<td>.</td>
<td>SENT</td>
<td>.</td>
<td>sentence delimiter</td>
</tr>
</tbody>
</table>

Words in texts can be syntactically labeled and grouped into phrases. Each word in a text can be mapped to a syntactic category such as verb, nouns or others. Together with syntactically related words it forms phrases and clauses. The following section gives an overview about methodologies for the syntactical analysis of texts.

2.7.2 Syntactical methods.

Syntactical information about words and sequences of words in a text helps to solve task of Information Extraction. The English noun “lead”, the metal, and the verb “lead”, being in charge of, may be distinguished by a syntactic analysis.

Part-of-speech tagging. The process of mapping textual tokens to syntactic categories is named part-of-speech tagging (POS tagging). Table 2.25 shows an output of a POS tagger for a sentence. The third column shows the lemma, the morphologically unchanged base form of the word. Tokens such as sentence periods, prepositions and the English preposition “to” in this example have specific categories. There exist two groups of POS tagging algorithms.

Rule-based systems apply transformations on the input text directed by lexical and contextual signals [33]. The transformation rules are learned using a corpus of manually tagged tokens. In each iteration step additional rules are tested whether they improve tagging performance or not by comparing the output of the tagger with the manually tagged corpus.

Statistical taggers use machine learning algorithms which train probabilistic models [32, 218, 133]. The assignment of a POS category in such a model is dependent on the current word and on the previously read words. The transition probabilities are learnt from large manually tagged corpora. Zhou et al. [294] retrain a Hidden Markov based POS tagger on the GENIA corpus and improve the performance of their Named Entity Recognizer significantly. The output of a POS tagger can be used in a phrase chunker.
**Phrase noun phrase**

**Phrase chunking.** Chunking is the process of grouping non-overlapping sequences of words into noun phrases, verb phrases or adjective phrases. Table 2.27 shows the previous example sentence chunked in phrases. Methods used for chunking include rule-based approaches [217], Hidden Markov models [180] and Support Vector Machines [151]. A recent survey on biomedical phrase chunkers is given in Wermter et al. [276]. The authors report that the performance for chunkers trained on a newswire corpus drops by 5% when tested on the GENIA corpus. As seen in other natural language tasks a retraining of the systems on a biomecial corpus is necessary.

**Clause parsing.** A clause is a word or a sequence of words consisting of a subject and a predicate. A sentence consists of one or more clauses. A clause may contain another clause. The following passage gives an example of a sentence with a clause contained in another one.

*The fact that normal cell lines were not affected by treatment with geodi-amolide H stimulates new studies toward therapeutic use for this peptide.*

(PMID: 18330887)

Full sentence parsing is a challenging task. It relies on the previous lexical and syntactical analyses. Nevertheless full sentence parsing was used in biomedical textmining [173, 284]. In recent benchmarks [74, 182] systems employing full sentence parsing were not superior over those relying on the elaborated analyses. This might be due to the fact that the available parsers nowadays are imperfect.

**2.7.3 Semantic methods.**

The analysis of the semantics of a word or a sequence of words such as chunks, phrases, sentences or whole texts is the attempt to infer the meaning of such words. The meaning of words may be restricted due to the context they appear in. For a semantic analysis additional background knowledge has to be used beside lexical and syntactical information. Such background knowledge may contain logical facts such as taxonomic information, e.g. a dog is an animal and animals are eukaryotes. Hierarchical "is-a" relations are called subsumption hierarchies. In contrast to taxonomies which constitute subsumption hierarchies only ontologies use a variety of relation types to describe concepts. An ontology might provide the information that eukaryotes have cells with membranes and prokaryotes have a nucleus. Using this knowledge a human can infer that an organism named dog must have cells with nuclei.

Consider the following sentences:
Regulation of meiotic progression by the meiosis-specific checkpoint kinase Mek1 in fission yeast. (PMID: 12458291)

Regulation of synaptic efficacy by coincidence of postsynaptic APs and EP-SPs. (PMID: 8985014)

Regulation of insulin receptor function by a small molecule insulin receptor activator. (PMID: 12213804)

With the appropriate knowledge about the concepts “meiotic progression”, “synaptic efficacy” and “insulin receptor function” we could for example infer that all three sentences mention a biological process. Using lexical and syntactic analysis one could identify the entities effecting the regulations. If those entities could again be mapped to a knowledge resource, such as an ontology, further statements could be made.

The correctness or generality of such statements needs to be handled. (1) The semantic analysis or previous processing steps may have introduced misinterpretation. (2) The knowledge resource may contain wrong facts or facts which can not generalized for every case. (3) The analysed text simple might state a wrong information. It is therefore advisable to label extracted facts hypotheses up until validation.

A recent survey [295] identified new frontiers in biomedical textmining including Text Summarization and Question Answering.

**Text Summarization.** The process of compiling as concise representation of the content of one or more documents is named Text Summarization. Targetted summarization focuses on a specific domain, e.g. collecting the best treatments for a disease from clinical documents. Indicative summaries aim at helping the reader to decided whether the summarized documents are of interest. The authors state that identifying the main topics of a document, e.g. MeSH terms, is related to summarization. Névéol et al. [185] provide methods for indexing the biomedical literature with MeSH headings. The authors assign multiple concepts to a documents to describe its content. Stoica and Hearst [260] assign GO concepts to genes by searching biomedical text for assigned GO concepts to orthologue genes, similar genes known from other species. Höglund et al. [127] use categorization of document abstracts for predicting subcellular localization. Such methods are closely related to the open problem 1 and will be discussde in the section 2.8.4 on state-of-the-art methodologies for Term Recognition.

**Question Answering.** Question answering systems analyse a full question sentence, formulate a query to some document source, retrieve relevant information, extract specific information from the relevant documents and summarize the answers for presentation. A recent survey on biomedical Question Answering was given in Zweigenbaum [296]. The described systems are relevant for the Open problem 2 of this thesis and will be discussed [in the appropriate section].

Most system for advanced task in biomedical textmining use semantic methods nowadays. Specifically do they use biological background knowledge in the form of databases, dictionaries, taxonomies or ontologies, see section 2.2 for a discussion on them. A common goal of such systems is to close the semantic gap between textual mentions of entities, concepts or relations in text and agreed formal representations in knowledge resources developed and used by the biomedical community. The more formal and logically founded the resource the more complex inferences can be made using them. On the other hand the less complex a knowledge representation is designed the more contributers can be attracted.
2.7.4 Software Resources

In order to implement a prototypical textmining applications various software packages can be used out of the box or as application programming interfaces.

**UIMA.** The Unstructured Information Management Architecture is a Java framework, now under Apache licence, which provides an architecture for Java and C++ components for from natural language dialog, information retrieval, topic-tracking, named-entity detection, document classification and machine translation [82].

**GATE** is a Java text mining toolkit developed at the University of Sheffield and released under the GNU Lesser General Public License. GATE provides a general offset-oriented development/deployment environment/framework and some rule-based tools to run within that framework.

**LingPipe** is a Java API for linguistic processing tasks that include: tokenization, sentence detection, part-of-speech tagging, phrase chunking, entity detection, within document coreference.

**JULIE** The JULIE Lab offers a comprehensive NLP tool suite for the application purposes of semantic search, information extraction and text mining. Most of their continuously expanding tool suite is based on machine learning methods and thus is domain- and language independent.

**OpenNLP** is a heterogeneous collection of projects distributed under a variety of open source licenses. The main projects being developed for OpenNLP itself include a general Java maximum entropy package released under the GNU Lesser General Public License.

**MALLET** is a Java natural language toolkit developed at the University of Massachusetts and released under the Common Public License. MALLET is most widely used for classification and sequence modeling. It also includes clustering. It provides maximum entropy training, including conditional random fields, general undirected graphical models, finite-state transducers, and some general numerical optimization classes.

Important properties of software packages and APIs are a good documentation for quick learning of programming patterns, open source code for efficient error fixing and debugging, tools for retraining of models to adopt new domains and an active community in order exchange experiences. Algorithms used in this software packages are described in the following section.

2.8 Terminology recognition in the life sciences

Biomedical concepts are defined in ontologies or taxonomies as discussed in section 2.2. Biomedical concepts are similar to Named Entities, which are instances of concepts such as real persons, companies, genes, proteins and others. However from the linguistic perspective, they differ in that Named Entities are mostly proper nouns, while biomedical concepts mostly contain common nouns. Proper nouns and common nouns differ in their syntactic properties. This section describes algorithms for the identification of Named Entities from text. The algorithms were applied for Entity Recognition in general English text as well as in the
biomedical domain. Chapter 3 discusses how the Alignment-based approach and Machine learning approaches can be applied for Term Recognition in the life sciences.

### 2.8.1 Problem definition for Terminology Recognition

Krauthammer and Nenadic [149] distinguish Term Recognition, Term Classification and Term Mapping. The authors define **Term Recognition** as *the process of marking single or several adjacent words that indicate the presence of domain concepts, ignoring the meaning of such words.* The authors further define **Term Classification** as *the process which assigns terms to broad biomedical classes, such as genes, proteins or mRNAs. The aim is to establish a broad notion of the nature of a biomedical concept.* This can be reflected by rather general concepts of an ontology. Furthermore the authors define **Term Mapping** as *the process of linking terms to well-defined concepts of referent data sources, such as controlled vocabularies or databases.*

The definition of Term Recognition is closely related to definitions for Named Entity Recognition given in the literature. “Identification of material names” [91], “The recognition of entity names (for people and organizations), place names, temporal expressions, and certain types of numerical expressions” [108] and “The task of identifying words and phrases in free text that belong to certain classes of interest” [243]. An ontology describing natural language processing methods would clearly label Term Recognition as a synonym for Named Entity Recognition.

One difference between Named Entity Recognition and Term Classification can be seen in the use of nouns. While named entities make more frequently use of proper nouns, e.g. “p53”, “BRAC1”, “Mr. Bean”, term or concepts mostly are composed of common nouns, e.g. “digestive process”, “human protein”, “mouse embryo”. Challenging for Named Entity Recognition are exception where named entities are formed by common words/nouns, e.g. there exist proteins with the names “for”, “an”, “by” “ken and barbie”, “lost in space”, “hu li tai sho”, “cheap date”, and “broken heart”. Challenging for Term Classification is the fact that terms may have different meanings in different context. The following two adjacent sentences in a publication refer to different concepts labeled with the development:

> Engineering of extensor tendon complex remains an unexplored area in tendon engineering research. In addition, less is known about the mechanism of mechanical loading in human tendon development and maturation. In the current study, an ex vivo approach was developed to investigate these issues. (PMID: 18423583)

Krauthammer and Nenadic [149] require the linking of terms to well-defined concepts during Term Mapping. An ontology is defined as a specification of a conceptualization [109]. Taking these definitions into consideration a strict definition can be suggested:

**Definition 1. Concept Recognition** is *the process of mapping lexical units in text to formally defined concepts of a commonly used ontology.*

The difference between Term Classification and Concept Recognition is that Concept Recognition requires the reference to concepts by unique identifiers having the same meaning in other texts as well. The referenced ontology must be a valid conceptualization independently of the scope of the text.

Concept Recognition, in contrast to Concept Annotation by the author of a text, is a difficult task. Only the author can unambiguously state the intended reference to an ontology concept. An annotator different from the original author will have to make assumptions about the authors intention based on his own experiences which might be different from
annotator to annotator. Even in this ideal case, when the authors establish the links to ontology concepts themselves, there might be a disagreement among the users of an ontology about the meaning of its concepts. Letting this case aside, Concept Annotation, can truly establish semantic links between documents which can be processed automatically. Concept Recognition aims at mimicking the Concept Annotation by the original author.

The following section describes techniques frequently used to solve the tasks of Named Entity Recognition and Term Classification. Section 2.8.4 discusses systems most relevant for the task of Concept Recognition.

2.8.2 Terminology recognition methods

Dictionary-based

Dictionary-based based approaches use a predefined collection of terms from terminological resources to identify occurrences in text. While this can be very efficiently done, e.g. using Finite State Automata [111], the straightforward use of a fixed dictionary reduces the precision and recall of a system (see section 2.9 for a discussion on evaluation measures). The amount of neologisms, new created terms, is large in the biomedical field. Therefore a dictionary will be incomplete most times. Unknown entities will result in a high ratio of missed terms. Hirschman et al. [122] report of 16 to 69% missed terms in their experiments. Another problem of the simple use of dictionaries is the ambiguity of terms. Hirschman et al. [122] report only 2 to 7% correctly identified terms in their experiments with a simple dictionary approach.

The number of registered term labels in a dictionary is fixed while the number of term variations is continuously growing. An approach to target the potentially infinite number of term variants are rule-based methods.

Rule-based

McDonald [174] first used contextual information for his “proper name recognition and classification facility”. The system processed text in three stages. First it identifies candidates using lexical hints and a dictionary. Second, it classifies the candidates using surrounding words as its context. Third, the identified terms can later be reidentified when mentioned as abbreviations.

Ananiadou [7] suggest a method which uses layers of rules which reflect morphological modifications. Fukuda et al. [91] describe a method which extracts target material names by performing connection and extension processing around a core term. They achieve good results with this rule based approach by first finding core-terms with five rules. Afterwards, the core-terms are connected, using other rules and a part of speech tagger.

Hakenberg et al. [111] expand their dictionary by generating variations of gene and protein names after splitting them at visual gaps. A visual gap is for example in the gene name “BRAC1” between the “C” and the “1” because of the change from letters to numbers. The expansion rules allow for variations like “BRAC-1” or “BRAC 1”. At a later stage the arabic numbers are allowed to be interchanged with roman numbers resulting in the acceptance of “BRAC I”.

Rule-based approaches need tedious manual tuning for each domain. The resulting systems are therefore domain-specific and need considerable manual adoption if applied to new domains.
2.8. TERMINOLOGY RECOGNITION IN THE LIFE SCIENCES

Alignment-based

The problem of low recall due to unknown variations of term labels was targeted by Tsuruoka and Tsujii [267, 268]. The authors allowed for edit operations in the terms such as substitution, deletion, insertion of characters and digits. Krauthammer et al. [150] employed a sequence alignment approach to the problem of term to text alignment. They translate the text and the terms into nucleotide codes and use the BLAST algorithm to compute the alignment.

The current version of GoPubMed uses an word alignment algorithm [67]. The text and the terms are decomposed into token stems. A local sequence alignment algorithm [248] is used to map term tokens to text tokens. Penalties values for gaps, deletions and insertions were experimentally calculated. The word alignment is geared by the notion of Information Value for a word. The Information Value for a word is based on the frequency of the occurrence of words in the ontology. The algorithm was tested on 100 manually curated MEDLINE abstracts and achieved good results (89.5% precision and 81.4% recall). The algorithm has a quadratic runtime, the nature of all approaches based on dynamic programming. Processing of a MEDLINE abstract takes about 10ms upon fresh annotation. Considering the worst case of 10,000 new articles for a query result the annotation pipeline alone uses 100 seconds. Heuristics and a caching mechanism ensures an average response of the current system within less than 10 seconds. The system does not disambiguate the meaning of words. Therefore false annotations of short ambiguous ontology concepts occur in the current GoPubMed.

Machine-learning methods can be used to target the tasks of Word Sense Disambiguation [204].

Machine-learning

Each of the limitations of the previously described approaches was targeted by machine-learning methods. For example Collier et al. [53] used Hidden Markov Models and specific orthographic features for the extracting of genes and protein names. Features used include word contains only lower case letters, word contains caps, e.g. “kappaB”, letters and digits mixed, e.g. “p53” and word contains greek letters. Yamamoto et al. [285] use morphological features, e.g. prefixes, lexical features, e.g. part-of-speech tags and stems, and syntactical features, e.g. noun phrases, as well as a dictionary of protein names. The authors report that the use of a dictionary is crucial in protein name tagging. Gaizauskas et al. [92] use machine-learned rules and heuristics for Names Entity Recognition. Hakenberg et al. [111] trained a Support Vector Machine for the disambiguation of gene names.

This section describes machine-learning techniques frequently used for biomedical textmining. All machine-learning systems have in common that a training component is involved, mostly they rely on manually produced training data and a correct tokenization is essential for their performance.

Hidden Markov Models. Hidden Markov Models were first used for part-of-speech tagging by Church [51]. In Bikel et al. [24] the authors suggest to use HMMs for Named Entity Recognition and achieve similar performance compared to rule-based approaches without the need to manually create rules. The idea is to learn the probabilistic model of an automata which produces labels while consuming the text tokens.

The advantage is that HMMs work on sequences and that is exactly how texts are represented. However there is a drawback of using Hidden Markov Models on texts. HMMs can only consume atomic observations. One token after each other is consumed by the
model but no additional hints can such as POS tags, previously identified entities or other meta-data can be utilized. Other machine-learning approaches can use complex feature vectors.

**Support Vector Machines.** Support Vector Machines classify objects represented as feature vectors [271]. The idea is to compute the optimal classifier for a set of training objects. The training objects are represented as vectors. The optimal classifier is a hyperplane between the vectors of the different classes. In the two-dimensional case the hyperplane can be line between the vectors (points in the two-dimensional case). Figure 2.13a shows the simple two dimensional case. The hyperplane is determined by the vectors closest to the area between the two classes, also called Support Vectors. This means not all training vectors are needed for the definition of the optimal classifier. Therefore the dimension of the feature vectors can be very large and sparse. Eventually only the supporting vectors determine the classifier.

Figure 2.13b show a case were no linear classifier can be found. In this case the vector space of the feature vectors is transformed into a higher dimension for which a linear classifier can be found. The computation of the classifier in the higher dimension is done implicitly. This is possible because of the properties of the Kernel function, see Burges [40] for a good tutorial on Support Vector Machines.

![Figure 2.13: (a) linear case, the hyperplane is a line between the 2D-vectors (b) non-linear case, a Kernel function is needed](image)

The computation can be expensive for large training sets and the classification of a standard SVM is always binary. The following methods allow for the classification into more than two classes.

**Maximum Entropy (ME).** In Berger et al. [20] the authors suggest a method for statistical modeling based on maximum entropy. The idea of the ME approache is to compute the probability of each known class for a given feature vector by making the minimal assuptions about the data. The authors introduce the idea with an example: Suppose from the training data we see that words from an input stream are classified into five classes (A,B,C,D and E). If we do not know the frequency for any class assignment the model has to assume that each class is assigned with a probability $p(A) = p(B) = p(C) = p(D) = p(E) = 20\%$. This model makes the most uniform assumption about the observed data. Suppose we know that in 30\% of the cases the input vector is classified into class A or B the constraints for our model are the following:
2.8. TERMINOLOGY RECOGNITION IN THE LIFE SCIENCES

\[ p(A) + p(B) + p(C) + p(D) + p(E) = 100\% \]

\[ p(A) + p(B) = 30\% \]

One of the possible solutions for these equations is: \( p(A) = 15\% \) \( p(B) = 15\% \) \( p(C) = 70/3\% \) \( p(D) = 70/3\% \) \( p(E) = 70/3\% \). Is the most uniform model and makes the least assumptions about the data. Suppose we observe further that in 50\% of the cases the correct class is A or C the constraints for our model are the following:

\[ p(A) + p(B) + p(C) + p(D) + p(E) = 100\% \]

\[ p(A) + p(B) = 30\% \]

\[ p(A) + p(C) = 50\% \]

The model maximizing the entropy here is: \( p(A) = 20\% \) \( p(B) = 10\% \) \( p(C) = 30\% \) \( p(D) = 20\% \) \( p(E) = 20\% \).

The authors describe the goal intuitively: model all that is known and assume nothing about that which is unknown. In other words, given a collection of facts, choose a model consistent with all the facts, but otherwise as uniform as possible.

The example illustrates that the output of a ME model are probabilities for multiple outcomes. ME models can handle large feature vectors and large amounts of training data. The final model is computed iteratively using the Viterbi algorithm [273].

Maximum Entropy Models were used for Part-of-speech tagging [218], Sentence segmentation [226], Named Entity Recognition [220], Noun Phrase Chunking [20] and others.

2.8.3 Word Sense Disambiguation

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 [240]. WSD is essentially 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. Assigning senses to word occurrences can comprise two assumptions. According to the “one sense per discourse” assumption, within a discourse (e.g. a document) a word is only used in one sense [93]. The “one sense per collocation” assumption supports that words have a tendency to exhibit only one sense in a given collocation (neighbouring words) [287].

During the last years, word sense disambiguation has become a hot topic in the biomedical domain. The challenge here is the rapid growth of the biomedical 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.

Two main decision points for WSD in the biomedical domain are (1) the granularity to which WSD should be performed and (2) the selection of an appropriate corpus for training and evaluation.

First, concerning granularity, some disambiguation tasks are easier than others; “bank” the building and the “BANK” gene will appear in completely different context, whereas the “BANK” gene, protein or mRNA are even likely to appear in the same article abstract, making often the disambiguation task difficult even for the common reader. “Transport by air” or “patient transport” will be easier to distinguish from the GO sense of transport, but “transport of virus cultures” will appear in a closer molecular biology context. Distinguishing between “transport”, “RNA transport”, “tRNA transport” or “ion transport” can
become less difficult by using the hierarchical information in the ontology, e.g. exploiting subsumption relations between ontology terms. Some terms are also easier to disambiguate in the same task, depending on: the number of their different senses, the way they appear in text (e.g. some can be easily distinguished with the help of regular expressions) and the number of tokens they comprise (one-token terms are usually more ambiguous as they are more likely to correspond to common English).

Second, concerning the biomedical corpora, those are few, mainly due to the time-consuming and labor-intensive process of manual or semi-automatic annotation. Examples of such datasets are the NLM WSD test collection [275], Medstract\(^9\) for acronyms and the BioCreAtIvE set\(^10\). However, depending on the task, researchers need to collect own gold standard datasets.

WSD algorithms can be distinguished as \textit{supervised}, \textit{unsupervised}, or using \textit{established knowledge} [240, 73]. Table 6.5.3 in the appendix compares available approaches. In the biomedical domain researchers have focused on supervised methods [115, 163, 97, 195] and using established knowledge [237, 128, 111, 78] 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 [111] 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 [97]. The above approaches use algorithms such as cosine similarity [237], SVM [97, 195], Bayes, decision trees, induced rules [115], and background knowledge sources such as the Unified Medical Language System (UMLS)[30], Medical Subject Headings (MeSH)[184], and the Gene Ontology (GO)[14]. Two approaches use meta-data, such as authors [78] and Journal Descriptor Indexing [128]. Most of the \textit{unsupervised} approaches so far were evaluated outside the biomedical domain [239, 241, 199, 211, 288, 70, 176], with the exception of [278], who used relations between terms given by the UMLS for unsupervised WSD of medical documents and achieved 74% precision and 49% recall. Another approach by [70] is based on a graph model representing words and relationships (co-occurrences) between them and uses WordNet [80] for assigning labels.

\subsection{2.8.4 State-of-the-art in Concept Recognition}

The systems which participated in the BioCreAtIvE workshop [27] are relevant for this work. In the first subtask 2.1, a system is given a document, an associated protein and a GO code, and is asked to return a segment of text from the document that supports the annotation of the text with the GO code (the evidence text). In the second subtask (2.2), a system is given a document and an associated protein, and is asked to return all GO codes that the pair should be annotated with, along with the associated evidence text for each GO code.

\textbf{FiGO.} In Couto et al. [57] the authors present a method to identify GO terms from unstructured text. This task is highly relevant to the open problem number 1 discussed in this thesis. The systems aim was to identify Gene Ontology terms cooccurring with protein names in sentences.

The main idea of the system named FiGO is the use of the evidence content of term names. In a preprocessing step the system creates a statistic of each word used in the ontology, basically the frequency of usage in ontology terms. The term \texttt{PUNT BINDING} adds

\footnotesize
\(^9\)www.medstract.org
\(^10\)www.mitre.org/public/biocreative/
one to the frequency of the two words 'punt' and 'binding'. The word evidence content is defined as:

\[ \text{WordEC}(w) = -\log\left(\frac{\text{Freq}(w)}{\text{MaxFreq}}\right) \]

where \( \text{MaxFreq} \) is defined as the highest observed word frequency in the ontology. The name evidence content is defined as:

\[ \text{NameEC}(n) = \sum_{w \in \text{Words}(n)} \text{WordEC}(w) \]

A sentence is associated with a concept if one of the concepts synonyms shares enough words with the sentence. A threshold can be set to manipulate the sensitivity of the algorithm. Unfortunately no information about the specific tokenization method, the stopword list and the sentence segmentation algorithm is provided. The authors do not mention any morphological methods employed by FiGO.

The authors tested their system in the first BioCreAtIvE workshop. Their system performed second best in task 2.1 with a precision of 28.72% and in task 2.2 about 8.91%. In comparison with the other systems this was a good result. The authors argue reasonably that their system would achieve about 30% more precision if the task had not a second part, the identification of protein names, which is a difficult task by itself.

The authors list several problems in their approach which they think decreased the performance:

- FiGO predicted obsolete terms, terms which have be marked for deletion from the ontology. In another context this might even be a desirable behaviour of the system. Obsolete terms might be still relevant for other ontologies.

- FiGO did not filter the GO terms that could not be annotated with Human proteins (e.g. germination). The workshop task restricted the annotation to human proteins. The workshop used manual curation data from the Human GOA database which itself was restricted to human genes and proteins. This seems more a problem of gene name mapping than of Concept Recognition. Hakenberg et al. [111] present a system to map human gene and protein names to their database identifiers. A prefiltering of the sentences for mentions of human gene/protein names would clearly improve FiGOs performance without changing the system.

- FiGO predicted GO terms out of context. A pre-selection of relevant ontology terms would resolve this problem.

- FiGO selected sentences from irrelevant sections (e.g. “Material and Methods”). Again a pre-processing step segmenting the text correctly would improve FiGOs performance without changing the system at all.

- FiGO does not search for term mentions across sentence boundaries. It is not clear if the overall performance of the system would benefit from segmenting the text differently. A high number of false positive candidates could decrease the precision of the system dramatically without further filtering.

- FiGO does not take into account how often a term was used in the texts. A high frequency might indicate a correct term. On the other hand it might also indicate a rather general term.
FiGO does not take the word order into account. It is not clear whether the system would benefit from taking the word order into account. A sense disambiguation step to filter out false candidates seems more preferably.

FiGO represents a promising approach to Concept Recognition. The system achieved 28.72% precision in the BioCreAtIvE task 2.1. The recall numbers were among the highest. The evaluation included the task of identifying protein names which reduced the precision in the combined task. Therefore another evaluation was executed:

GOAnnotator [58] is an application of the FiGO algorithm. It additionally uses a concept similarity measure to discard candidates which are not similar enough to a pre-selected GO term, named uncurated annotations. The pre-selected term was derived from other automatically derived annotation of annotation databases, e.g. UniProt. The validation of automatically annotated proteins with GO terms is a tedious work. Out of 1953 uncurated proteins for 66 proteins GOAnnotator presented evidence text to human curators. The 66 proteins were selected because GOAnnotator reported the highest confidence for its annotation and the identified terms were most similar to the pre-selected terms. Given the hints of the uncurated annotation the system had a very good precision of 93%. The authors argue that syntactical techniques, the modification of the segmentation and the identification of noun phrases could improve the performance.

MeKE. Another GO term extraction method uses a hybrid approach of phrase pattern matching and Bayes sentence classification [50]. It is based on Chiang and Yu [49] and uses a simple tokenization at whitespaces, punctuations and brackets. The tokenizer turns “alpha(1,3)- fucosyltransferase” as well as “alpha-1,3-fucosyltransferase” into “alpha 1 3 fucosyltransferase”. A simple stemming of plural forms in contrast to a full morphological analysis is executed, e.g. “protein transport” and “protein transporter” are equal.

The system works in stages. First sentences are split and positions of candidates of protein names and GO terms stored in a relational database. Protein names and GO terms were stored as a dictionary and processed with the same tokenizer and stemming algorithm. GO term variants were added to the dictionary by mining other terminological resources for terms with a minimal edit distance to the original terms, e.g. “inner mitochondrial membrane” (source LocusLink, now EntrezGene) becomes a variant for “mitochondrial inner membrane” (source GO).

Now sentences with co-occurring GO terms and proteins are processed by a POS tagger and phrases are identified using a finite state automaton. Previously identified syntactic pattern, using sentence alignment [49], are then used to filter the sentences, e.g.

```
{ {NP {{. . P}}} } <- Place holder for Protein
{ VP {{'plays' . .}}} }
{ NP {{'role' . .}}} }
{ . {{ IN .}}} }
{ { { . . G}}} } <- Place holder for GO term
```

A sentence matching such a specific pattern is selected immediately. If no pattern matches the sentence is classified using a Naïve Bayes classifier described in Chiang and Yu [49]. If the classifier accepts the sentence as a protein-function describing sentence the sentence is selected as a candidate. Candidates selected by the sentence classifier have a lower priority than the ones selected by the pattern matching. Finally the same steps are executed for the GO term variants. The candidates are ranked using again the sentence classifier which was basically trained to estimate the probability that a sentence describes a gene function.
2.8. TERMINOLOGY RECOGNITION IN THE LIFE SCIENCES

The system was most precise in the BioCreAtIvE task 2.1 with 80% correct annotations for proteins with GO terms and 34.15% in task 2.2. However the methods recall was very low overall. The authors argue that the use of learned patterns account for the high precision. Patterns in their approach are short sequences of consecutive words, e.g. prefixes such as “Amino acid”, “Role of”, “Show that”.

**Proximity words network.** Verspoor *et al.* [272] found that GO terms have frequently short labels and are therefore not informative enough. They suggest a method to compute a network of other words frequently appearing in the same paragraph of a GO term. Those words were taken as additional hints for the mention of the GO term itself. The system score 25.73% precision in the BioCreAtIvE task 2.1 and 5.52% in task 2.2.

**Sliding windows.** Some systems use a technique named sliding window to restrict the range of the text under consideration for each extraction candidate. Ehrl er *et al.* [75] used a sliding window of 5 words across each sentence and a vector space model to identify GO term. Official results reported 25.57% precision for the task 2.1 and 12.3% in task 2.2. Krallinger *et al.* [147] use sentences as their basic unit of sliding windows and achieve 28.86% precision on the task 2.1.

**Label-based SVM.** Rice *et al.* [227] trained a Support Vector Machine on 1858 term-protein-document triplets for 638 distinct GO terms. (2.82 per Term) As document features they used labels identified by the C-value method which is also used for terminology generation from large document corpora [87]. The idea behind this is that labels generated with this method represent a corpus well and linguistic variants can be identified during this generation process. Grouping those lexical variants of the same label creates normalized features of documents which can be learn by a machine learning method. The precision of the system with 12.53% precision was low and also the recall was lower than other system. A reason for this result is the small amount of training data and the fact that 43% of the GO terms were not at all represented in the training data. In task 2.2 the system had a precision of 3.48%.

**Statistical learning.** Ray and Craven [219] followed a statistical approach. They extensively used external databases to train their Naïve Bayes classifier on bags of words. Namely they used the SGD, FlyBase, WormBase and TAIR databases which are all contributers to the GOA database [41]. In contrast to Couto *et al.* [57] they estimated the informative nature of a term word not on the ontology terminology but on the training data. For each GO term they counted unigrams, bigrams and trigrams of words in the abstracts provided by GOA annotations. They run a \( \chi^2 \) test and ranked the strings according to their indication of the term. They found for example that the word ‘symporter’ is not significantly indicative for the presence of the term SODIUM SYMPORTER ACTIVITY. The current version of GO contains 94 concept names with the word ‘symporter’ and 47 concepts with the word ‘sodium’. Therefore the approach of Couto *et al.* [57], FiGO, would rate the word ‘sodium’ more informative than the word ‘symporter’ as well but in contrast to this method based on the ontology terminology.

The authors used the ontology relations to collect more training data for general terms. GOA annotations associated with a terms descendant concepts were also used for training the ancestor terms. The accumulated training examples were used to train Naïve Bayes classifiers for each GO term. In the BioCreAtIvE task 2.1 the system had 20.10% precision and 21.31% precision in task 2.2.
BioCreAtIvE Evaluation. The evaluation carried out by Blaschke et al. [27] revealed that the systems made mostly non-overlapping predictions. Except the system of Chiang and Yu [50] which made mostly predictions overlapping with predictions of other systems. This suggests that all approaches found a partial solution to the problem. The MeKE system seems to be most general to the problem solution. Although it generated a low number of prediction. The recall performances of the systems are not directly comparable as the manual curation was carried out after the submissions. The curators only judged the correctness of the submissions but did not curate the full corpus. So the maximum number of correct predictions is unknown for the corpus.

Interestingly the GOA database curators use their own large scale electronic GO annotation tool which is not textmining based. They report a precision of 91 to 100%. They conclude that current textmining methods can not support their work [43].

Latent Semantic Analysis. A recently published article [270] describes a system which uses latent semantic analysis (LSA) for browsing PubMed articles using the Gene Ontology. The system transforms documents and queries into vectors of stemmed words. The high dimensional vector is then projected into a lower dimension via singular value decomposition. The reduced vectors are then ranked using a the cosine similarity measure. To assess the performance of their system the authors select three GO terms and manually compare 16 articles for their association with GO terms assigned by Doms [67] and their system. They conclude that latent semantic analysis can associate documents with terms not mentioned in the text.

LSA & Patterns. A similar approach was combined with the idea of sliding windows as mentioned earlier. In Ruch [234] the author describes a system combining a module aiming at high recall, a vector space model, based on word stems and cosine similarity, and a module aiming at high precision, a pattern based approach similar to the one described in Ehrler et al. [75]. The result of the first module is a ranked list of candidates. Candidates also identified by the second module get a boost and are ranked higher depending on their length. The author finds that GO term are more difficult to identify than MeSH term. Synonyms are more helpful for GO than for MeSH. The vector space module is complementary to the pattern module.

MetaMap. MetaMap [12] is a program that maps biomedical text to concepts in the UMLS Metathesaurus. The program relies heavily on dictionaries. Namely the SPECIALIST lexicon and the UMLS Metathesaurus. The data files provided for the program are manually maintained and contain several hundred thousand entries. The SPECIALIST lexicon lists acronyms, abbreviations, synonyms, derivational variants and meaningful combinations of these. The Metathesaurus contains more than 1.3 million strings and the SPECIALIST lexicon provides 360,688 entries with 625,041 inflected forms.

The basic idea of MetaMap is to identify noun phrases in the text and marking the head noun. Then for each word in the phrases known variants are looked up in the SPECIALIST lexicon. Metathesaurus labels containing those variants become candidates. The candidates are ranked by mapping the labels to the noun phrase from the text. Four measures are combined: centrality (involvement of the head noun), variation (an average of inverse variant distance from its original), coverage and number of pieces. The best of the candidates are organized into a final mapping in such a way as to best cover the text.

The output is used by human curators to annotate MEDLINE abstracts. New articles in PubMed are manually annotated with a delay of one or two month. MetaMap is written
2.9 Evaluation Methodologies

It is important to know which quality can be expected from a textmining system. To measure the quality usually manually annotated corpora are used. The so called 'gold standard' provides a large enough collection of problems and their manually created solution, e.g. a text in which all mentions of an ontology concepts were made by an expert. The quality usually is measured by the percentage correctly identified entities, recall, and the percentage of entities which were identified correctly, precision.

2.9.1 Definitions of quality measures.

To assess the quality of a system several statistics can be collected. Suppose entity $a$ belongs to class $A$ and $b$ does not belong to class $A$.

**True positives.** A true positive event is observed when the system classifies $a$ correctly as to belonging to class $A$.

**True negatives.** A true negative event is observed when the system classifies $a$ correctly as not belonging to class $A$.

**False positives.** A false positive event is observed when the system classifies $b$ as to belonging to class $A$.

**False negatives.** A false negative event is observed when the system classifies $a$ as not to belonging to class $A$.

\[
\begin{array}{|c|c|c|}
\hline
 & \text{relevant entities} & \text{non relevant entities} \\
\hline
\text{identified} & \textbf{True Positives (TP)} & \text{False Positives (FP)} \\
\text{not identified} & \text{False Negatives (FN)} & \textbf{True Negatives (TN)} \\
\hline
\end{array}
\]

Table 2.28: Contingency matrix for the outcomes of a term identification system. This shows all possible outcomes of a system which has the task of identifying relevant entities in a text.

All outcomes of a term identification system can be assigned to one of the four possibilities in the contingency matrix in table 2.28. Any classification algorithm aims at maximizing the true positives and the true negatives.

**Recall** (or sensitivity) is the measure of the ability of an algorithm to extract all relevant entities.

\[
\text{recall} = \frac{\text{number of relevant entities extracted}}{\text{total number of relevant entities in the abstract}} = \frac{TP}{TP + FN}
\]

A high recall indicates that most of the relevant entities were found. In contrast, a low recall indicates that the majority of the relevant entities were missed by the algorithm. Recall 100% means that every relevant entity was found. In theory, it is easy to achieve
good recall values: All entities for every document simply have to be returned. Recall by itself is not sufficient to characterize the effectiveness of an algorithm.

**Precision** measures the ability of an algorithm to retrieve only relevant entities.

\[ \text{precision} = \frac{\text{number of relevant entities extracted}}{\text{total number of entities extracted}} = \frac{TP}{TP + FP} \]

Precision is a measure how well an algorithm performs in avoiding to return non-relevant entities. A precision of 100% is reach when every entity extracted is relevant. There is no trivial way to achieve a 100% precision other than returning no term at all for every document.

**F-measure** combines recall and precision in a single efficiency measure.

\[ F = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \]

It is the harmonic mean of precision and recall.

In Baldi *et al.* [17] the authors point out that precision and recall are not only properties of the evaluated system but also depend on the frequency of positive examples in the evaluation corpus. Consider a corpus that contains 1000 positive examples and 1000 negative examples. A system assigns 90% of the positive examples and 10% of the negative examples as relevant. The system has a precision of 90% and a recall of 90% (TP=900, FP=100, TN=900, FN=100). Evaluating the system on a corpus with only 100 positive examples and 1900 negative examples the same system has a precision of only 32.1% and a recall of 90% (TP=90, FP=190, TN=1710, FN=10). Evaluating the system on a corpus with only 1900 positive examples and 100 negative examples the same system has a precision of 90% and a recall of 99.4% (TP=1710, FP=10, TN=90, FN=190). As a conclusion of this two systems can only be compared when they are evaluated on the same corpus.

To compare the efficiency of the algorithms, a recall-precision graph can be used, as described in the Text REtrieval Conference (TREC) quality benchmark [119].

Precision and recall are set-based measures. They evaluate the performance of an algorithm on an unordered set of retrieved items, e.g., terms in this case. However, they do not account for the quality of ranking the extracted terms. The better the algorithm is, the more relevant terms are among the top items in the hit list. The relevance of a ranking can be measured by computing the precision at different cutoff values, precision at \( n \). In a list, where the first three terms are relevant and the fourth and fifth are non-relevant, one gets a precision of 100% for the first three and a precision of 60% for all five terms.

Figure 2.14 shows an interpolated precision graph for a list of 20 terms extracted from a document. The first, second, third and fifteenth terms in the hit list are relevant terms. The graph is plotted at 11 standard recall levels \( r = (0.0, 0.1, ..., 1.0) \). The exact recall points are 0.25, 0.5, 0.75 and 1.0. That means that after retrieving 50% of the total number of relevant terms, two of four terms in this case, the precision is still 100%. At recall level 10 (90% of the relevant terms retrieved), the precision is 27%.

**Recall-precision Graph** is a way of displaying the precision at \( n \), as one goes through the list of extracted terms. It plots average interpolated precision numbers against percentage recall for all processed documents. Recall-precision graphs usually have a concave shape. Trying to increase the recall typically gives more and more false positives in the list,
2.9. EVALUATION METHODOLOGIES

Figure 2.14: Interpolated term extraction precisions for one article. The dotted line marks the interpolated values.

The precision decreases disproportionally. The area under the recall-precision graph is the average precision.

\[ \text{precision}(\text{recall} = r) = \max(\text{precision}(\text{recall} \geq r)) \]

**Recall-precision averages table** is computed by summing the interpolated precisions at specific recall levels taken from the interpolated precision graphs for all processed documents. The values of the recall-precision table are later used to draw the recall-precision graph.

**Average precision and average recall** do not measure the average of the precision and recall values at standard levels, they rather evaluate the averages of the precision and recall values, obtained after each relevant term is calculated.

2.9.2 Difficulties of corpus-based evaluations.

To evaluate a literature-mining method, its output is either compared to a gold standard or is manually inspected by an expert. In Leser and Hakenberg [159] the authors name facts which make the evaluation based on manually annotated corpora difficult.

**Few available Gold Standards.** The number of accessible test corpora is limited. Section 2.6.3 describes the most important accessible corpora available today. Table 2.21 shows that the available corpora mainly focus on protein and gene entities in text. Only the FetchProt corpus provides marked up annotations of Gene Ontology terms. Unfortunately the number of different terms and the amount of manual annotations do not allow for evaluating a Term Recognition algorithm. The GENIA corpus provides enough annotations for an evaluation. However the 33 ontology terms do not cover for an evaluation of an ontology with thousands of terms. The terms of the GENIA ontology can only be mapped to the Gene Ontology or MeSH on a very high level.
CHAPTER 2. BACKGROUND

In Doms [67] an evaluation set of 100 documents marked up with GO terms was used but it was not created by domain experts. The freely accessible GOA database provides a large amount of manually curated associations between PubMed identifiers and GO terms which characterize gene and proteins. However the textual evidences are from the full texts which are often not freely accessible. In Doms [67] the evaluation was executed on the freely available abstract texts. The recall performance could therefore not be measured realistically. As many mentions of GO terms do not appear in the abstract. Also the precision performance seemed reduced on this dataset because not only GO term characterizing proteins were annotated by the system. The non-protein describing GO terms were counted as false positives despite the fact they were correctly identified.

The BioCreAtIvE workshop provided the participants with associations of GO terms with text passages from PubMed articles. It is the best available Gold Standard for GO term markups in full texts of PubMed articles. A drawback is that the markups are not complete and reflect only annotations for selected proteins. Relevant mentions of other GO terms were not evaluated by the human curators.

Low annotator agreements. Leser and Hakenberg [159] point out that the markup of entities in text depends greatly on the curators. Curators might miss correct annotations because no biologist knows thousands of proteins including their synonyms and abbreviations. Ideally the curation process is executed by several experts on the same texts. In this case an inter-annotator agreement can be measured which marks a natural upper bound on the quality achievable by automatic systems. Inter-annotator agreement is reported between 75 to 90% for gene and protein names [92, 3]. Clear curation guidelines improve the curation consistency [168].

Identifying GO terms in full text articles seems to produce a high disagreement between curators. Results reported in Camon et al. [43] indicate that there is 39% chance of curators exactly interpreting the text and selecting the same GO term, a 43% chance that they will extract a term from a different lineage, and a 19% chance that they will annotate a term from the same GO lineage. Ignoring differing lineages this amount to a curator agreement of 58% only. This is due to the fact that curators are taught to annotate according to their individual level of confidence. Nevertheless 72% of the time curators recalled all possible valid GO terms from the articles.

Taking the fact into account that human expert curators disagree considerably depending on the task there is a high risk that any system reporting a higher performance represents an overfitting to the particular gold standard.

Problem severity depends on entity type. A varying severity is observed between different entity classes, e.g. the best f-measure for protein and gene names recognition in the second BioCreAtIvE workshop was 85%. Recognition performance on ontology concepts is frequently lower, 25 to 80% [57, 50, 272].

The severity of Ontology Term Recognition is high because each concept in the ontology has its own entity class. Another problem is that most concepts defined in ontologies, e.g. the OBO Foundry, are not covered by any training corpus. Hersh et al. [117] report much lower performance rates for the GO term subtask (50-60%) than for the other subtasks.

Unsharp markup boundaries. Another difficulty is the ambiguous beginning and ending characters of entities. Alex et al. [3] report the most disagreements about the markup ranges. Depending on the guidelines organism names may or may not be part of a protein
2.9. EVALUATION METHODOLOGIES

name, e.g. human growth hormone or human IGF-II protein. It also depends on the guidelines how conjunctions are tagged in the corpus, e.g. CSN subunit 4,5,6 or CSN subunit 4,5,6. See also section 2.6.3 on inline vs. standoff annotations. Standoff annotations can express conjunctions better than inline annotations. To make systems comparable the annotations can be normalized to the sentence level. The problem of markup boundaries is less severe for ontology-based literature search as the automatic annotations are normalized to sentence or abstract level.

Bias toward specific tasks. The systems evaluated in the BioCreAtIvE workshop assumed that database curators favor a high precision the support their task. On the other hand Alex et al. [3] state that some curators prefer a high recall. Also when analyzing large amounts of text to identify protein interaction networks a high recall might be favorable when human evaluate the networks. Variations of names are tolerated by humans naturally. When comparing systems the specific task the system was designed for has to be taken into account.

In the case of ontology-based literature search a high recall is favorable because filtering articles is done using general terms. An intolerant recognition strategy might fail to associate a document at all in the ontology. The document becomes invisible in this case.

2.9.3 Back-To-Back Tests.

Back-To-Back tests are a common software quality measure. The idea is to assume the output of one system as the standard or baseline and the other system is evaluated against this. In the case of Ontology Term Recognition both systems must have access to the same textual informations and background knowledge, e.g. full texts and same ontology version. MeSH headings and Gene Ontology terms are predicted by the systems described in section 2.8.4. Vanterru et al. [270] executed a back-to-back test with the GoPubMed system on three queries to the online system.

2.9.4 BioCreAtIvE

The most relevant large scale comparison to the problem of Ontology Term Recognition are the task 2.1 and 2.2 of the BioCreAtIvE workshop held in 2004. The first task was concerned with the identification of gene mentions in text and linking protein database entries to abstracts. The second task was related to the extraction of human gene product annotations with GO terms (i.e. text passages supporting those annotations). The second BioCreAtIvE workshop held in 2007 did not include a task comparable to the tasks 2.1 and 2.2 of the year 2004.

Task 2.1 Identification of annotation relevant text passages The goal was to evaluate the approaches for the extraction of text passages which contain statements that relate GO terms to gene products. The participants were provided with a test set consisting of triplets of protein identifiers (Swiss-Prot accession number), GO term and the articles’ filenames. They returned text fragments giving evidence for the predicted GO term and its association to the gene product.

Task 2.2 Assignment of GO terms to gene products This task simulated the human annotation procedure. Participants had to return GO terms and the text evidences supporting the protein-GO term association. The number of GO terms expected was given.
Recall of the system could not be compared as the evaluation was carried out after submission of the predictions. Only the submitted prediction were evaluated by human curators. The maximal number of correct GO evidences was never evaluated.

The results of both tasks can be used to evaluate the Term Recognition algorithm described in the following chapter. Assuming the different systems produced mostly non-overlapping predictions and were thus complementary as reported in Blaschke et al. [27] one can assume nearly completeness of the annotations on the provided texts. Therefore an estimation of the recall can be done. Both tasks were connected to the problem of gene/protein name identification. After the workshop the evaluated prediction were made available for future evaluations.

2.9.5 GOA database.

All data used in the BioCreAtIvE workshop stems from the GOA database. A larger scale evaluation can be executed on the freely accessible curations marked with TAS (traceable author statement). This however is restricted by the lack of full texts for the referenced articles. The BioCreAtIvE subset provides the relevant text passages. Similar dataset provided by the MGI database was used in other evaluations as well.

2.9.6 TREC Genomics Track

The TREC Genomics Track 2006 [118] was an annual activity of the information retrieval community aiming to evaluate systems and users from 2003 to 2007. In the years 2003 and 2004 a relevant subtask was to assign GO terminology to literature abstracts to MEDLINE abstracts. The data used came from the MGI database and did not include full texts. The MGI data is now part of the GOA database. For the evaluation of biomedical search engines a new single task was developed that focused on retrieval of passages (from part to sentence to paragraph in length) with linkage to the source document. Topics are expressed as questions and the systems were measured on how well they retrieve relevant information at the passage, aspect, and document level. The participating systems returned passages linked to source documents. Judges rated the returned passages and grouped them by aspect.

2.9.7 Author curations.

For the evaluation of the Concept Recognition of GO terms in literature abstracts the BioCreAtIvE corpus is a good dataset. Some difficulties with corpus-based evaluations as described above apply: (1) The number of annotations provided is still low with 202 articles and 286 protein-GO term pairs. (2) The provided gold standard misses GO term mentions not related to protein or gene mentions. (3) The full texts are given in SGML format which is considerably more difficult to parse. Especially because not well formed fragments were provided. This might affect the markup boundaries and correct tokenization. (4) The original annotations were biased toward precision by database curators. Not all GO terms are represented in the evaluation base. (5) Finally the evaluation was carried out by three curators and the reported agreement was not very high.

An ideal evaluation corpus for the task of Ontology Term Recognition in literature abstracts would consist of curation made by each individual author of an article. A benchmark based on such data would reflect many individual curation styles. Resulting in a test corpus best representing the authors intention. The purpose of the Ontology Concept Recognition is to associate a whole abstract with a concept. Therefore it would be valid to ask the authors to base their decisions on the whole abstract instead of individual statements. This
clearly creates difficulties in case there are different lexical mentions with different meaning. In this case the author needs to decide whether or not to associate the whole text with the concept. Authors could be invited to curate their own abstracts if the terms found in them were never before evaluated. This could maximize the coverage of the ontology curation. An inter-annotator agreement evaluation could easily be implemented. An evaluation based on this ideal gold standard measures the system performance most realistically.

2.10 Outline

This work is arranged in three main chapters. Following the background section, chapter 3 describes a Concept Recognition framework. Given a corpus of scientific abstracts from the PubMed literature database all mentions of Gene Ontology concepts are automatically detected. The framework allows to import the textual labels of an ontology and to generate recognition rules for each label. Terminological characteristics such as word frequency statistics or syntactic categories of label words are used as well as relational information stored with the ontology. Maximum Entropy models to distinguish the meaning of ambiguous concept labels can be trained. The importance of this task became evident with international competitions such as the BioCreAtIvE workshop and the TREC Genomics Track.

A literature corpus marked-up with semantic links to ontology concepts is a prerequisite for ontology-based literature search. Chapter 4 describes methods to reason over millions of documents using ontologies containing thousands of concepts. It is shown how the semantic links and the structural information in the background knowledge of ontologies and taxonomies can be used to answer biomedical questions. Algorithms are developed which identify informative aspects covered by a subset of documents and link them to formally defined concepts. To demonstrate the usefulness of the new search paradigm questions raised by biologists and listed in the TREC Genomics Track 2006 were answered with the system. The first ontology-based, large scale, online available, up-to-date bibliometric analysis for topics in molecular biology is described.

The methods developed for answering real biomedical questions with an ontology and a literature corpus are the basis for the biomedical search engine on the web - GoPubMed. Chapter 5 describes the requirement analysis, the software requirement specification and the architectural design of the first ontology-based search engine which is currently used by thousands of users worldwide including research institutes such as Stanford School of Medicine, Unilever Research & Development and Federal Institute for Risk Assessment in Germany.
Chapter 3

Algorithms for Concept Recognition

The goal of this work is to design an efficient method to annotate biomedical literature abstracts with textual concepts of biomedical ontologies. This chapter describes a Concept Recognition framework developed for this work. It allows to incorporate various ideas used in promising approaches seen in the literature. The contribution made can be grouped into four stages:

(1) Pre-processing modules for ontology terminology were developed. The result are recognition patterns which are used as rules to recognize concept candidates.

- A scripting environment for the development and test of Concept Recognition pipelines was implemented. This facilitates the development of the textmining pipeline.

- Basic textmining modules were developed for the pipeline including improved stemming strategies, modules adding lexical variants to the rules and Gene Ontology specific heuristics.

- Heuristics were developed to compute the information value of concept label words. This reduced the number of false positives.

(2) A recognition engine using the produced patterns was developed. It enables the simultaneous detection of several ten-thousands of concept on one machine.

- The recognition engine which loads patterns of ontology concepts and detects candidate concepts and links them to text ranges in the input text.

- The quadratic complexity of the word alignments required was addressed with a caching solution.

- A pre-selection of candidates based on word importance was implemented.

(3) Ambiguous meanings of concepts were filtered using maximum entropy models. This improved the precision of the pipeline.

- Three approaches of word sense disambiguation for ontology concepts were compared

- Very high recall (97.5%) and precision (91%) values were achieved when applying the context model approach to predict document of interest to the human curators of the MGI database.
• The most ambiguous concepts tested could be disambiguated with 100-200 training examples

• The recognition performance was improved by 25.7% achieving 79.9% induced precision and 72.7% induced recall.

• The most discriminating feature for the disambiguation task were identified: the title words, the journal name and textual phrases

(4) The performance of the pipeline was measured. The results can be used to compare other approaches and future modules. The resulting curation data can be used for the training of the maximum entropy models.

• A recently published benchmark dataset was used to compare the recognition performance to approaches in the literature.

• A new benchmark comprised of 689 PubMed abstracts and 18,356 curations, personally curated by the original authors, was created.

• A new performance measure was developed to reflect ontology-based annotation characteristics.

The contributions made in section 3.2.3 are joined work and were submitted for publication [5]. The Meta-Data method was implemented for this work using the Maxent\(^1\) package.

\(^1\)maxent.sourceforge.net
3.1 Introduction

It has been a long studied problem to analyze natural language texts and to identify lexical
tokens referring to a real world object or a conception. Depending on the intended abstrac-
tion level systems aim at linking text with physically objects, such as a living person, or
some generalization, e.g. a type of cells. A generalization typically retains only information
which is relevant for the specific purpose leaving out some properties such as the individual
organism a cell was studied in.

The task of Named Entity Recognition was previously studied in the newswire domain,
see section 2.7 on textmining methodologies. One focus was on living persons, existing
companies and business events mentioned in the newswire domain. Recently ontologies
became popular to encode domain knowledge in a computable form, see section 2.2 on
terminologies and ontologies in the life sciences. Types of entities can now be described in
relation to other concepts, e.g. the Gene Ontology describes biological functions of protein
families. Recently it has become a field of research to identify ontology concepts in free text,
see section 2.8 on terminology recognition in the life sciences. Textmining competitions
such as the BioCreAtIvE workshop and the TREC Genomics track aimed at evaluating
biomedical textmining systems, see section 2.9 on evaluation methodologies.

The goal of this work is to design an efficient method to annotate biomedical literature
abstracts with textual concepts of biomedical ontologies. The MetaMap algorithm does
this for the MeSH terminology and is used to generate suggestions for the human curator
of PubMed. This work focuses on Gene Ontology concepts as the BioCreAtIvE challenge
showed it is an open problem. Out of a pool of several thousands ontological concepts the
system has to detect candidates in a way that tolerates potentially new syntactical variants
of known concepts aiming at a high recall. Concepts with ambiguous senses need to be
distinguished from mentions with different meanings aiming at a high precision.

A number of promising approaches exist. Section 2.8.4 discusses the state of the art
systems in biomedical concept recognition. Systems with high precision had an extremely low
recall. One result of the BioCreAtIvE I workshop was that the systems were complementary
to each other. This suggests that a combination of strategies may outperform each single
approach. All systems in this workshop focused on the annotation of gene products with
Gene Ontology terms supported by the literature. A crucial point was the assumption of
a mention of the gene or gene product in a short text. For the described objective this
assumption can not be made. Literature abstracts refer to concepts of the Gene Ontology
without mentioning any gene or gene product. The lack of a corpus curated with Gene
Ontology concepts disregarding gene or protein mentions complicates the evaluation. The
systems did not aim at providing a general framework to load multiple ontologies and execute
the annotation using their concepts. Unfortunately no information on the execution times
of the systems were provided. It is an objective of this work to provide a system to annotate
literature abstracts on the fly while searching a citation database.

This chapter describes an improved method for Concept Recognition of ontology terms.
An earlier work [67] showed that an approach using sequence alignment and the information
value of ontology words is promising but fails on ambiguous concepts. Maximum Entropy
machine learning approaches are widely used for disambiguation tasks. However there is
not enough training data available for ten thousands on ontology concepts. The approach
taken is hybrid. In a first stage concept candidates are identified with a rule-based method
eliminating the quadratic costs of the sequence alignment approach. A confidence (or match)
value is computed based on information value of words. In case of an ambiguous concept a
trained model is applied to revise the candidate.

Problem: Training data available is limited

Goal: Hybrid approach
3.2 Methods

One result of the BioCreAtIvE I workshop was that the currently available state of the art methods for Concept Recognition are complementary to each other. This suggests that a combination of successful properties of the systems can be combined to achieve a better performance. The properties of the systems described in section 2.8.4 can be grouped into three categories: (1) pre-processing of the terminology, (2) processing of the input texts and (3) processing of the concept candidates. The following sections describe how the ideas were combined into a single framework for the recognition of concepts of a given ontology. The framework can be divided into four parts. At first the terminology of an ontology is pre-processed. The result are recognition patterns for each label of an ontology concept. Those patterns are loaded into an recognition engine which processes structured text and links potential candidates to text fragments and to contexts in the document, e.g. a candidate appears in the title or a sentences of the abstract. In the next stage the candidates are disambiguated using trained models which take binary features as input. A binary feature may be the cooccurrence of a word stem in the same sentence or the associated title or the name of a journal in the affiliation string. The result are semantic markups of ontology concepts in the title and abstracts of PubMed citations. Finally the recognition performance of the pipeline needs to be evaluated and training data for ambiguous concepts need to be collected.

### Concept Recognition Pipeline

<table>
<thead>
<tr>
<th>Terminology Processing</th>
<th>Input Text Processing</th>
<th>Sense Disambiguation</th>
<th>Performance Feedback</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tokenization</td>
<td>Structuring</td>
<td>Word Sense Disambiguation</td>
<td>Benchmarks</td>
</tr>
<tr>
<td>Morphology</td>
<td>sentence segmentation</td>
<td>best-friends approach</td>
<td>bioreative corpus</td>
</tr>
<tr>
<td>stemming</td>
<td>word tokenization</td>
<td>closest friends approach</td>
<td>author curations</td>
</tr>
<tr>
<td>lexical variants</td>
<td>content scopes</td>
<td>context models</td>
<td>Performance</td>
</tr>
<tr>
<td>Syntax</td>
<td>Concept Candidates</td>
<td>Training Data</td>
<td>Measures</td>
</tr>
<tr>
<td>part-of-speech</td>
<td>rule-based candidates</td>
<td>high quality</td>
<td>precision</td>
</tr>
<tr>
<td>sub-clauses</td>
<td>text ranges</td>
<td>medium quality</td>
<td>recall</td>
</tr>
<tr>
<td>Corpus</td>
<td>candidate selection</td>
<td>low quality</td>
<td>f-value</td>
</tr>
<tr>
<td>Characteristics</td>
<td>candidate ranking</td>
<td>Document Features</td>
<td>induced measures</td>
</tr>
<tr>
<td>word information</td>
<td></td>
<td>context scopes</td>
<td></td>
</tr>
<tr>
<td>domain specific rules</td>
<td></td>
<td>meta-information</td>
<td></td>
</tr>
</tbody>
</table>

### 3.2.1 Processing of the terminology

Authors do not have the concept labels in mind when they write abstracts. The goal of the Gene Ontology project is to create a common vocabulary to describe gene and gene products. In recent years the project became popular. An experiment on a full text index using Lucene\(^2\) showed that PubMed abstracts mentioned 28.3% of the Gene Ontology concept labels literally. Therefore the GO concept labels are a rich source of information for textmining tasks. It can be assumed that authors use them in lexically or syntactically

\(^2\)lucene.apache.org
3.2. METHODS

Figure 3.1: The table shows the distribution of the word counts per GO concepts. Most GO labels have 2-4 words. Very short labels such are rare and tend to be ambiguous. Very long labels encoding additional information in a subclause are a peculiarity of GO.

modified variants. The Gene Ontology has a systematic structure which was recently studied, see section 2.2.2 for a discussion of GO’s architecture. In order to identify mentions of GO concepts in free text the labels need to be preprocessed.

Concept labels.

Some approaches make use of the lexical and syntactical information provided with the labels. A promising approach decomposes the multi-word labels of concepts into words and computes how important each of the words is for the individual concept [57, 67], see section 2.8.4 on state of the art approaches for concept recognition.

The labels of a given ontology were segmented into lexical units, such as word tokens, whitespace characters, opening and closing brackets and commas were identified. Figure 3.1 shows the distribution of the number of words per concept label in GO. Most GO labels have 2-4 words.

Information values.

After tokenization and stemming, GO’s vocabulary consists of 7,841 words. The majority of the GO words found occur only once in the whole ontology. On the other hand 51 of the GO words occur at least 100 times in the ontology. Table 3.1 shows the words with the highest frequencies.

In figure 3.2, a log scale is used to display the word frequency distribution. In GO, 4,402 words, more than 56%, occur only once, and 7,136, more than 90%, do not occur more than 10 times.

For the recognition of concept labels from text, words with a very high frequency do not give much information as they are part of many labels in the ontology. However, extracting a word with a low frequency gives a much better hint about a mentioned concept. Highly frequent words give a low information value, low frequent words give a high information value.

The information value $I$ for a word with the frequency $f$ in the ontology was defined in
Table 3.1: Table of most frequent words in the Gene Ontology. “Activity” is the most often used word. GO appends the word “activity” to gene products to clarify that it refers to the attribute of the gene product and not to the product itself.

Figure 3.2: This plot shows the distribution of GO words in the Gene Ontology. The distribution follows Zipf’s law. A few words occur very frequent while many terms occur rarely. Words with a very high frequency do not give much information as they are part of many labels in the ontology. Highly frequent words have a low information value, low frequent words have a high information value.
3.2. METHODS

Doms [67] as follows:

\[ I(f) = (1 - \mu) \cdot e^{\left(\frac{1 - f}{\mu}\right)} + \mu \quad \text{with} \quad \alpha = \frac{1 - \eta}{\ln \left(\frac{\mu - \lambda}{\eta - 1}\right)} \]

where \(0 \leq \mu \leq 1\) is the lowest information value for any word and \(0 \leq \lambda \leq 1\) is the information value for a word with mean frequency \(\eta\).

The mean word frequency \(\eta\) can be computed from the given ontology. Optimal values for the parameters \(\mu\) and \(\lambda\) were previously determined in Doms [67].

<table>
<thead>
<tr>
<th>Revision</th>
<th>Heuristic</th>
<th>Count</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GlobalWordValue</td>
<td>statistical</td>
<td>57173</td>
<td>computes the global information value of all words as describes in this section</td>
</tr>
<tr>
<td>SubTermRevision</td>
<td>lexical</td>
<td>15648</td>
<td>increase information value of frequent word if otherwise the pattern is ambiguous</td>
</tr>
<tr>
<td>ArticleRevision</td>
<td>syntax</td>
<td>284</td>
<td>set information value of articles to a minimum</td>
</tr>
<tr>
<td>PrepositionRevision</td>
<td>syntax</td>
<td>21850</td>
<td>set information value of prepositions to a minimum</td>
</tr>
</tbody>
</table>

Table 3.3: Label Revisions modifying word information values

The word information value is applied to each instance of the word in terminology. This allows to estimate how important a word in a label is for the differentiation from other concept labels. The word activity is least informative to distinguish concept labels. The difficulty with this global information value is that some concepts are represented by labels which only differ in such a lower information word. The Gene Ontology contains the following concept labels: snRNA binding, snRNA transport and snRNA transcription. The word “snRNA” has a very high information value. The words binding, transport an transcription have low information values. Using only the global word information value it is not possible to select one concept label in case the authors mentions snRNA only and not one of the processes. In other cases the word binding is less important to distinguish meaning, e.g. type 1 fibroblast growth factor receptor binding is unambiguous also if binding is not mentioned explicitly.

**Problem:** word importance is not the same for each concept label

**Solution:** Label Revisions selectively modify word information values

Label Revisions. Label Revisions were developed as a means of applying specific heuristics to selected labels, e.g. modifying the word information value depending of the sibling concepts, introducing lexical variants for specific tokens such as numbers, filtering inflected variants based on syntactic categories. In contrast to other approaches this allows to modify the recognition heuristic for each label separately.

Other approaches use heuristics globally applied to the whole ontology’s vocabulary. Label revisions allow the adjustments of heuristic based on label specific characteristics. For example the computed global word information values can be modified for selected words. In the previous approach a list of stopwords was maintained. The information value of those words was set to zero for all occurrences. With Label Revisions this heuristic was replaced by two revisions the ArticleRevision and the PrepositionRevision. The two heuristics set the information value of a word in a label to a minimum of 0.5% if its syntactic category is an article or a preposition. Table 3.3 lists other heuristics used to modify information
values based on ontology properties, syntactic categories, lexical features and features based relations between concepts. The **SubTermRevision** modifies word information values of 15,648 label pattern which were otherwise ambiguous.

---

**Problem:**
wrong stemming produces false positives and too many candidates

**Goal:**
speed-up stemming and increase precision

---

Figure 3.3: Output of the ElivagarConsole program to pre-process the Gene Ontology vocabulary. Line 1 prompts the successful initialization of some taggers. Line 2 prompts the registering of the revisions. Java’s reflections mechanism is used to detect revision plugins. Line 9 executes an external script to load the 57173 labels of the Gene Ontology and run some revisions on them. In line 13 the GO specific trailers like “sensu ...” are removed and stored for later use. Line 14 executes the computation of global word information values. Line 16 executes revisions filtering invalid inflections of words based to the previously assign syntactic categories. Line 25 executes revisions adding lexical variants to word recognized as Roman and Arabic numbers. Line 28 marks mandatory word in each label. Line 30 saves the revised labels into a file.

**Morphological variants.**

Rule-based stemmers do not take the syntactic category into account when reducing a word to its lemma. This can result in wrong stem variants. A full syntactic analysis of the input text is time consuming. The idea is to analyse the concept label words with a part-of-speech tagger and remove invalid stemming variants.

For each lexical unit the syntactic category was assigned using the part-of-speech tagger. The implementation allows to select the tagger by wrapping the existing implementation. For the Gene Ontology the MedPost tagger [250] produced the best results as it was trained on biomedical texts.

For each lexical unit all known morphological variants used in the entire PubMed corpus were computed using the Porter Stemmer implementation of Snowball. The stemming algo-
3.2. METHODS

<table>
<thead>
<tr>
<th>Word in PubMed</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>acidically</td>
<td>non valid english word</td>
</tr>
<tr>
<td>acidity</td>
<td>refers to the pH level</td>
</tr>
<tr>
<td>acidities</td>
<td>Plural form of acidity</td>
</tr>
<tr>
<td>acidates</td>
<td>“amino acidates” is a rare synonym for “amino acids”</td>
</tr>
<tr>
<td>acidics</td>
<td>unknown lexical variant</td>
</tr>
<tr>
<td>acidism</td>
<td>unknown definition</td>
</tr>
<tr>
<td>acides</td>
<td>unknown inflection</td>
</tr>
<tr>
<td>acidic</td>
<td>adjective</td>
</tr>
<tr>
<td>acid</td>
<td>noun</td>
</tr>
<tr>
<td>acids</td>
<td>noun, plural</td>
</tr>
<tr>
<td>Acid</td>
<td>noun</td>
</tr>
<tr>
<td>Acids</td>
<td>noun, plural</td>
</tr>
</tbody>
</table>

Table 3.5: Words found in PubMed which are stemmed to “acid” by the PorterStemmer.

The algorithm is not aware of the syntactical category of a label word. WordNet was used in order to filter out invalid stemming variants based on the previously computed syntactical category. A stemming variant is invalid if its morphological root does not match the syntactic category of the word token. For example the term PROTEIN AMINO ACID OXIDATION contains the noun “acid”. As the PorterStemmer does not use syntactic information all words listed in table 3.5 are stemmed to the same lemma “acid”.

StemFilters. One reason to filter out improper inflections is to speed-up the recognition process while keeping the recall high. WordNet organizes lexical entities in sets of synonyms to describe a concepts. A synset contains synonyms and spelling variants, such as British or American English. A number of filters were implemented which retain only inflected forms of words found in a synset listed in WordNet. For example the PorterStemmer stems “upping”, “UPs”, “Ups”, “ups”, “UP”, “Up”, “uP” to the preposition “up”. Inflected words other than “up”, “UP” and “Up” are unlikely to map to the preposition but generate candidates which have to be filtered out later. Such lexical variants not listed by the synsets known by WordNet are removed from the list of recognized inflections. In the case of PROTEIN AMINO ACID OXIDATION only the inflected forms “acidic”, “acid”, “acids”, “Acid” and “Acids” are retained. The filters aim at reducing the number of false positive candidates by ignoring improper inflections. On the other hand label token may have lexical variants not listed by dictionaries. The following section describes revisions for adding valid lexical variants explicitly.

Lexical variants.

The Gene Ontology contains over 34,476 synonyms. However not all lexical variants are listed. The idea was to expand the vocabulary of words automatically for selected categories of words. For example numbers can be written in Roman or Arabic variants.

Two revisions, RomanNumbersRevision and ArabicNumbersRevision, were implemented which add the complementing variants. In the Gene Ontology terminology 762 labels contain a Roman number and 448 labels contain an Arabic number. Some label revisions can be applied to any ontology while others have been specifically developed for the Gene Ontology.

Solution: filter invalid stemming variants

Problem: some lexical variants are missing in the synonyms

Solution: write rules to expand vocabulary automatically
to deal with its peculiarities.

**Syntax analysis.**

The Gene Ontology has 2354 labels containing a subclause at the end of the main label. For example:

- oxidoreductase activity, acting on the CH-CH group of donors, quinone or related compound as acceptor
- oocyte axis determination, positioning of nucleus
- phosphogluco, water dikinase activity
- alcohol dehydrogenase activity, iron-dependent
- upregulation of long-day photoperiodism, flowering
- 3'-splice site cleavage, exon ligation
- viral spread within host, cell to cell

The revision **TrailerLabel** removes the subclauses from the label and stores it for a later post-filtering step after candidate computation. The words after the comma are removed and stored with the labels for later post-processing. Concept candidates are identified only using the words before the comma. The current implementation does not yet use the subclause for disambiguation. Future extensions may disambiguate concept labels based on similarity of the subclause and the concept’s context in the text.

Another pattern used in Gene Ontology labels are restrictions to specific organisms. This conflicts with the official definition of the Gene Ontology project which states: The Gene Ontology project provides a controlled vocabulary to describe gene and gene product attributes in any organism. The Gene Ontology lists 1032 concepts restricting their application scope to certain organisms. For example:

- cuticular attachment to epithelium (sensu Nematoda)
- structural constituent of pupal cuticle (sensu Insecta)
- proteasome regulatory particle (sensu Eukaryota)
- cytosolic large ribosomal subunit (sensu Bacteria)
- lymph gland hemocyte differentiation (sensu Arthropoda)

The revision **SensuOrganism** removes the phrase in brackets from the main label and stores the string for a later post-filtering.

**Goal:** develop and test the revisions

**Revision Console.** For the development of the Label Revisions a development and testing environment using a scripting language was developed. Basic functionalities of the scripting environment are:

- Loading of OBO ontologies
- Plugin Framework for Label Revisions
- Scripting language to describe revision pipes
3.2. METHODS

- Training of disambiguation models based on curation databases
- Validation of disambiguation models
- Testing of Concept Recognition using the revised ontology
- Creation of benchmarks from curation databases
- Execution of benchmarks
- Saving of ontology revisions

Solution: scripting environment

The environment can be used to load an ontology, work with revisions and execute benchmarks on test corpora. A revision is applied to a set of concept labels. Each revision defines a predicate to decide whether the revision applies to concept label or not. For example, the `SensuOrganism` revision accepts only labels ending on the “(sensu .*)” pattern. After determining all accepted labels the revision applies the revision rule. The `SensuOrganism` removes the part in brackets from the label and stores it with the label for later reuse. The console environment identifies revisions using Java’s reflections mechanism and manages their lifecycles. The revisions have two functions: (1) selecting of concept labels with a predicate and (2) modify each selected concept label. Possible modifications are: splitting the label into alternatives, removing the label, adding or deleting word variants, assigning context models. Some revisions, for example the revisions based on the WordNet dictionary, require large resources. Once a revision was initialized it can be reused.

The console environment can be used to experiment with revisions on ontologies. After applying a new revision benchmarks can be executed and the performance before and after can be compared. It is as well possible to simply the recognition performance on single phrases and documents from a database. The following section describes the implementation of the candidate identification which takes a revised ontology model as input and processes an input text.

3.2.2 Candidate identification

The next step is to identify non-continuous text passages which represent potential mentions of ontology concept labels. The difficulty is that the mentions may contain insertions of words into the labels or some words may be omitted by the author because from the context the meaning becomes clear to the readers. Another goal was to make the recognition phase independent from the size of the background terminology. The candidates should be produced in one parse.

The recognition algorithm takes single sentences as input. The assumption made here is that a sentence naturally states one or more facts. Statements comprising several sentences tend to be more complex and it is more difficult to analyse them as additional complex analyses become necessary such as anaphora resolution. Some systems [75, 147] use a technique called “sliding window” in which only a limited sequence of words in a text is considered for the identification of a concept label candidate. The sliding window technique does not require the analysis of sentence beginning and endings but misses labels spawning over more words than the size of the sliding window.

Candidate Generation

Some approaches analyse the input text and assign syntactic categories to the text words. Chiang and Yu [50] use this technique to detect phrase patterns mentioning Gene Ontology
terms. The intention is to assign noun phrase tags to continuous sequences of words. In Chiang and Yu [50] a concept candidate must be entirely contained in a single noun phrase. This approach can recognize concepts mentioned in long phrases while it misses concepts mentioned in separated phrases. A method to detect concept words across non-continuous phrases is treebanking. Some systems use full sentence parsing for recognition tasks. In the BioCreAtIvE workshop such systems did not perform better than approaches based on simple sentence segmentation [27]. It is argued that one reason might be that the available parsers are imperfect. They produce a high precision while missing many correct results, resulting in a low recall.

Goal: identify all candidates in one parse

For this work a rule-based sentence segmentation algorithm was implemented which is based on a tokenization of the text into lexical units, see listing 6.1 in the appendix. The resulting sentences were searched for mentions of known word variants. Each sentence was processed using the algorithm described in listing 3.4. One problem was that too many candidates were produced initially. Most of them were filtered out in the next step which retains only those having enough evidence in the text. However a large amount of candidates were produced because of words with a low information value such as the words “of” or “from”. An optimization generates only candidates if at least one important word was found. The TermCoreRevision computes the important words as follows: all least informative words with a combined information value of less than 30% are flagged, the other words considered important for the candidate identification.

```java
1     findCandidates(Text sentence) {
2         foreach token:sentence {
3             if( token is an important word in a candidate ) {
4                 store candidate to list
5                 store range of token with candidates in list
6                 if( token word is known in ontology) {
7                     store token in separate list
8                 }
9             }
10         }
11     foreach candidate in list {
12         add missing tokens from separate list
13     }
14     return completed candidates
15 }
```

Figure 3.4: This algorithm produces only concept label candidates which have a textual token for the required words. For example the term lytic virus budding from plasma membrane is not selected as a candidate if only inflected forms of the words “from” and “membrane” are detected. The two words combine an information value of less than 16% of the whole label.

Problem: several candidates for one label in a sentence, word alignment is expensive

Pre-selection of candidates. Sentences can contain several potential mentions of a the same concept. In the previous implementation [67] this was computed using a local sequence alignment algorithm. To reduced the computational costs solutions are cached. For example: given a concept label with the words $w_1$ and $w_2$. A long sentence may contain the ten words: $w^0w^1w^2w^3w^4w^5w^6w^7w^8w^9$. There are four potential candidates of the label in this sentence: $w^0w^6$, $w^1w^2$, $w^4w^6$ and $w^4w^9$. The problem are written as:

$$
\begin{pmatrix}
2 & 4 & 6 & 9 \\
1 & 1 & 2 & 2
\end{pmatrix}
$$
3.2. METHODS

Ignoring the tokens not part of the combinatorial task the problem can be normalized to:

\[
\begin{pmatrix}
0 & 1 & 2 & 3 \\
1 & 1 & 2 & 2
\end{pmatrix}
\]

The solutions to the problem are:

\[
\begin{pmatrix}
0 & 2 \\
1 & 2
\end{pmatrix},
\begin{pmatrix}
0 & 3 \\
1 & 2
\end{pmatrix},
\begin{pmatrix}
1 & 2 \\
1 & 2
\end{pmatrix},
\begin{pmatrix}
1 & 3 \\
1 & 2
\end{pmatrix}
\]

The solutions are normalized and stored in a cache. The cached solutions are translated back into the text as follows:

\[
\begin{pmatrix}
2 & 6 \\
1 & 2
\end{pmatrix},
\begin{pmatrix}
2 & 9 \\
1 & 2
\end{pmatrix},
\begin{pmatrix}
4 & 6 \\
1 & 2
\end{pmatrix},
\begin{pmatrix}
4 & 9 \\
1 & 2
\end{pmatrix}
\]

Caching frequently reused solution allows to reduce the computational costs at the expense of memory resources. The following section describes how the concept candidates are weighted according to the information value of the identified words.

Candidate Ranking

False positives need to be filtered from the concept candidates. The idea is to remove candidates at this stage which do not carry enough information for a positive identification. For example the sentence: A highly specific binding signal was measured. does not carry enough information to decide which binding activity is meant.

The idea is to rank the candidates according to their information value and set a threshold of minimum information value. Concept candidates are ranked according to their confidence value. The confidence value \( m \) of a concept label candidate \( c \) is defined as:

\[
m(c) = \sum_{i=1}^{n} w_i \cdot g_c(w_i)
\]

where \( n \) is the number of words in the concept label and \( g \) is a weighting function. A simple definition of \( g \) is:

\[
g_c(w) = \begin{cases} 
1 & \text{candidate } c \text{ contains } w \\
0 & \text{otherwise}
\end{cases}
\]

to cope with the ambiguity of words another definition of \( g \) was used:

\[
g_c(w) = \begin{cases} 
d(c, w) & \text{candidate } c \text{ contains } w \\
0 & \text{otherwise}
\end{cases}
\]

where \( d \) is the confidence that \( w \) has the same meaning in the context of the candidate \( c \) as defined by the concept. Words used in natural languages are ambiguous. Their meaning depend on the context they are used in. The following section discusses experiments carried out with the goal to determine an efficient method to deal with ambiguous concepts.

3.2.3 Disambiguation Methods

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 [240]. WSD is essentially 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 can comprise two assumptions: a. within a discourse, e.g. a document, a word is only used in one sense [93] and b. words have a tendency to exhibit only one sense in a given collocation - neighboring words [287].

During the last years, word sense disambiguation has become a hot topic in the biomedical domain. The challenge here is the rapid growth of the biomedical 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.

Two main decision points for WSD in the biomedical domain are the granularity to which WSD should be performed and the selection of an appropriate corpus for training and evaluation. Concerning granularity, some tasks are easier than others (e.g. distinguishing between “BANK” as a building vs the “BANK” gene is easier than “BANK” gene vs the protein). Concerning the biomedical corpora, those are few, mainly due to the time-consuming and labor-intensive process of manual or semi-automatic annotation. Examples of such datasets are the NLM WSD test collection [275], Medstract for acronyms [213] and the BioCreAtIvE set for mouse, fruitfly, and yeast. However, depending on the task, researchers need to collect their own gold standard datasets.

Algorithms for Word Sense Disambiguation

WSD algorithms can be distinguished as supervised, unsupervised, or using established knowledge [240, 73], see also table 6.5.3 in the appendix. In the biomedical domain researchers have focused on supervised methods [115, 163, 97, 195] and using established knowledge [237, 128, 111, 78] 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 [111] 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 [97]. The above approaches use cosine similarity [237], SVM [97, 195], Bayes, decision trees, induced rules [115], and background knowledge sources such as the Unified Medical Language System (UMLS) [30], Medical Subject Headings (MeSH) [184], and the Gene Ontology (GO) [14]. Two approaches use meta-data, such as authors [78] and Journal Descriptor Indexing [128]. Most of the unsupervised approaches so far were evaluated outside the biomedical domain [239, 241, 199, 211, 288, 70, 176], with the exception of Widdows et al. [278], who used relations between terms given by the UMLS for unsupervised WSD of medical documents and achieved 74% precision and 49% recall. Another approach by Dorow and Widdows [70] is based on a graph model representing words and relationships (co-occurrences) between them and uses WordNet [80] 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 [216, 261, 225, 160, 164, 15, 238, 61]. The work described in this section fills this gap by systematically comparing three approaches using ontologies with inference and semantic similarity and the use of meta-data to solve the problem of WSD for ontological terms. The goal is to establish how the use of ontologies and meta-data can improve results.

In previous work Andreopoulos et al. [9] proposed and evaluated two approaches to the WSD problem, namely term co-occurrences in PubMed abstracts and document clustering.
3.2. METHODS

A methodology was proposed for finding whether an article in an automatically annotated database is likely to be true or false with respect to the biological meaning and constructed a co-occurrence graph of GO terms based on Gene Ontology Annotations (GOA) [42]. This approach extends this approach (called ‘Term Cooc’) to disambiguate GO and MeSH terms in a way that it incorporates relationships of terms in GO and MeSH (by propagating the terms’ log-odds scores toward the top of the hierarchies, ‘Inferred Cooc’). It also combines the graph-based decision function with a support vector machine, arranged in a co-training scheme, to learn and improve models without any labeled data. The ‘Term Cooc’ approach is similar to Dorow’s approach [70], with the difference that it constructs the co-occurrence graph based on GOA and MeSH, which are manually annotated datasets. Therefore, the graphs contain only relations (edges) between terms (nodes) that are semantically meaningful in the context of an article. Dorow’s graph contains all the nouns that co-occur with one another, but in the case of the biological context, only the local subgraph of Dorow’s graph is of interest (i.e. ‘development’ only in the biomedical sense). Another difference is that the approach uses established knowledge in GO and MeSH to draw the nodes and in the different configurations of the method it uses a support vector machine and/or incorporate the term relationships in GO and MeSH.

Two more methods for disambiguation are introduced, differing from the co-occurrences approach in terms of automation and background knowledge required. The ‘Closest Sense’ approach computes similarities between the senses of the ambiguous term, the senses of its neighbors (co-occurring terms) and the type of relations that could occur between them. ‘Closest Sense’ (CS) uses the UMLS semantic network as background knowledge, like Widows et al. [278], who rely on the context of the ambiguous term in order to compute a score for each sense candidate. This score consists of the number of terms in the document which are related, in the UMLS, with the different senses of the ambiguous term. In comparison to the CS method, this approach is different in two main points: (i) it does not take advantage of the hierarchies of concepts and relations in the UMLS and (ii) it ignores terms which co-occur with the ambiguous term in the same context but do not have a direct link with it in the UMLS. The ‘Meta-Data’ method uses maximum entropy for modeling the behavior of occurrence of contextual terms and phrases in text together with a potentially ambiguous term. The features selected are n-tuples of word stems and meta-data such as the journal and document title. The method requires a set of labeled documents for each term to be disambiguated. Following the three strategies for WSD are evaluated and compared, starting from the unsupervised/automated to the least automated one. The comparison includes each method’s requirements and limitations in terms of training data and automation, the behavior of the methods during the use of different taxonomies (GO/MeSH/UMLS) as well as comparison against a classical stem co-occurrence approach.

Classification of approaches. Three Word Sense Disambiguation methods were designed, implemented, and evaluated. The following section refer to them as: Closest Sense (CS), Term Cooc (TC), and Meta-Data (MD). These differ as follows:

Background knowledge: Closest Sense (CS) uses the UMLS semantic network; it represents an abstract as a list of UMLS terms occurring in the abstract.

Term Cooc (TC) uses co-occurrences of terms in GO and MeSH, built from a curated dataset; it represents a document abstract as a list of GO and MeSH terms occurring in the abstract. The Meta-data method (MD) uses meta-data about the journal and title; it represents a document abstract as n-tuples of word stems and meta-data.

Classification: CS uses shortest semantic distance of co-occurrences to sense. TC uses SVMs and co-occurrences from a training dataset for finding boundary between senses. MD
CHAPTER 3. ALGORITHMS FOR CONCEPT RECOGNITION

Figure 3.5: The figure shows the path between the UMLS terms Body Part Organ or Organ Component and Amino Acid Peptide or Protein. The edges describe relations between entities (in our case, the subtype-aware-signature and its sub-properties) and nodes consist of classes and relations of the ontology. Body Part Organ or Organ Component is a subsumption of Fully Formed Anatomical Structure, which belongs to the signature of the relation produces. This relation has as range Organic Chemical which is a super-class of Amino Acid Peptide or Protein. The length of this path is 4.

uses the maximum entropy to model the behavior of the co-occurrence of contextual words and meta-data with the ambiguous term.

Solution:
senses are close if concepts are connected with short paths

Closest Sense method (CS). This WSD approach was initially used to address the ambiguity problems in the MeatAnnot system [142]. The main idea of the approach is the following: given a set of different senses of the ambiguous term, the co-occurring terms in the same text and the hierarchy where they belong (including the different types of relations), decide which sense is true based on the (shortest) distance to the senses of the co-occurring terms.

A new semantic distance is defined, the subtype-aware signature distance, which merges signature and hierarchies graphs. The main idea is to find a path between two concepts by using the ontology structure (subClassOf relations between terms, subPropertyOf relations between properties) and the signature of relations (domain and range). The subtype-aware signature comprises of relations in the hierarchy (subClassOf, subPropertyOf) additional to the common signature (domain and range of a property). It is aware of the properties of a term (signature), the position of the term in the hierarchy (subClassOf relations) and the hierarchy of the properties (subPropertyOf relations).

Figure 3.5 provides an example of the subtype-aware signature distance calculation between two terms in the UMLS semantic network. Body Part Organ or Organ Component is a subClassOf Fully Formed Anatomical Structure, which belongs to the signature of the relation produces. This relation has as range Organic Chemical which is a superClassOf Amino Acid Peptide or Protein.

For the example in Figure 3.6, the CS method determines the meaning of ‘thrush’ by examining what appears in the same sentence and/or paragraph (e.g. ‘mouth diseases’ or ‘oral ulcer’) and then computing a similarity based on semantic distances to ‘songbird’ and ‘oral candidiasis’ in the ontology; the highest similarity determines the result. Intuitively, with semantic distances, two senses are close if there exists a possibility to use them in a concise annotation graph. For instance, the concepts EMBRYONIC MITOTIC CELL
3.2. METHODS

Figure 3.6: *Thrush* is an ambiguous term, as its senses include *songbird* or *oral candidiasis*. This figure shows the possibilities for disambiguating ‘thrush’. Solid edges are *is a* relationships.

**CyCle and Eclosion** could be considered as close in an ontology with the following semantic relations: `EMBRYONIC MITOTIC CELL CYCLE → EMBRYONIC DEVELOPMENT → DEVELOPMENTAL PROCESS → HATCHING → EClosion`. This leads to a semantic distance $d_S(embryonic mitotic cell cycle, eclosion) = 4$.

**Algorithm.** The ‘Closest Sense’ algorithm takes as input: (i) the ambiguous term $\tau$, (ii) the vector $V_\tau$ of different senses of $\tau$, (iii) the vector $V_{C_\tau}$ of senses found in the context (sentence and/or paragraph containing the ambiguous term $\tau$), and (iv) the UMLS semantic network.

First, the disambiguator builds a vector $V_{C_\tau}$ of senses describing the context of the ambiguous term $\tau$. This vector comprises the senses of terms that are neighbors of $\tau$. Then, it computes the similarity between each sense in vectors $V_\tau$ and $V_{C_\tau}$.

The resulting similarity is the average of similarities between senses in the two vectors. Finally, the sense in $V_\tau$ that has the highest average similarity to $V_{C_\tau}$ is proposed as the best for $\tau$.

**Subsumption distance and signature distance.** A distance for CS is a metric space. Modifying the distance can be done by modifying the metric, the space or both. When an ontology is used to provide the space of a metric space, there are many ways to choose and tune the part of the ontology that will be used for the distance; e.g., using only the oriented graph of the subClassOf, using abstract classes to augment depth, or using different sub-parts of the ontology in different ways to adapt to local specificities of a model. Therefore, there is strong coupling between the design rationale used for the ontology and the characteristics...
of the space that will be used to define the distance. In addition, there is a tension between the ontology as a specification of a conceptualization and the ontology as a compilation of a categorization used as a metric space. It is possible that these two goals may not be reconcilable in one ontology. Alternatives to the classical distance for CS are: i. The subsumption distance is the length of the shortest path between two senses in the hierarchy of senses, where the length of an individual subsumption link gets exponentially smaller with the depth of the senses it links in the hierarchy. ii. The signature distance is the length of the shortest path between two classes through the graph formed by the property types with their range links and domain links.

**Solution:**

Senses are close if concepts are used often together

**Term Cooc method (TC).** The Term Cooc method relies on the highest co-occurring (with the ambiguous term) term among all the terms in the document. It selects the highest co-occurring term with the ambiguous term for defining the given sense as true or false. In order to formalize the notion of term co-occurrences (GO or MeSH), pairs of GO/MeSH terms that appear in the same abstract were considered and all such pairs of terms were represented in a manually annotated GOA or MeSH co-occurrence graph (see training with co-occurrence graphs subsection below). Each node in the co-occurrence graph represents a GO or MeSH manual annotation. An edge between nodes α and β represents a real number, the log-odds score, representing the frequency $\log \frac{\text{odds}(\alpha, \beta)}{\text{odds}(\beta, \alpha)}$ of the terms’ α and β co-occurrence over all articles, weighted by their total number of occurrences.

For the example in Figure 3.6, the TC method determines the meaning of ‘thrush’ by examining what appears in the same abstract (e.g. ‘swallows’) and then considering all known co-occurrences between ontology terms in a training corpus; the value of the highest co-occurrence determines the result, e.g. ‘swallows’ would have relatively high co-occurrence with ‘thrush’ songbird.

**Algorithm.** First, a simple threshold was used considering how close to an ambiguous term the highest co-occurring term (of the ones in the article) is; if below a user-defined threshold $\theta$, the ambiguous term is negative, else it is positive with respect to the term. Second, Support Vector Machines (SVMs) were trained on all tokens of a text [145].

The method first runs a Binary SVM against a set of articles ordered by maximum co-occurrence with the ambiguous term. The highest and lowest 10% of articles in the set are labeled as positive and negative; then the SVM is trained on lower 10% (article with least co-occurring term with the ambiguous term) and upper 10% (article with highest co-occurring term with the ambiguous term). After the initial convergence is achieved, the error (wrongly classified vectors) will be low, likely near 0. The algorithm next improves this result by iteratively re-classifying the remaining articles that have less extreme co-occurrences with the ambiguous term, one-by-one, followed by re-training the SVM on the newly relabeled data set. This continues until no more articles are left. The steps are:

1. Set $S = \text{Order articles based on their highest co-occurring GO/MeSH annotation (from the co-occurrence graph) with the ambiguous term.}$

2. $T = \text{lowest and highest 10\% of } S; \text{ label } T \text{ as negative and positive; train SVM with } T; \text{ remove } T \text{ from } S.$

3. For $s \in S$: move $s$ to $T; \text{ classify } s; \text{ re-train SVM with } T.$
3.2. METHODS

Training with co-occurrence graphs. These graphs are used in the TC method for training. For WSD of GO terms, the co-occurrence graph derived from the Gene Ontology Annotations (GOA) [42] and for the MeSH terms from the Medical Subject Headings (MeSH) [184]. GOA represents articles manually annotated with GO terms and consists of ~34,000 articles. There exist ~16,400,000 documents to which experts have assigned MeSH terms. Co-occurring terms were found in the GOA and MeSH annotated articles and built a co-occurrence graph representing how frequently pairs of GO or MeSH terms co-occur. Nodes represent annotations and edges represent the frequency of co-occurrence of two annotations in the same article, normalized based on each GOA/MeSH annotation’s individual occurrence frequency in the specific corpus.

Training with inferred co-occurrences. The co-occurrences were extended in a hierarchical fashion to ensure that given a GOA-derived co-occurrence between a pair of terms, $GOAcooc(\alpha, \beta)$, the ancestors of $\alpha$ and $\beta$ in the ontology are updated with the co-occurrence such that only the maximum co-occurrence is kept. This is important given the few annotations in GOA and the is-a relationships between GO terms, since ancestors inherit the co-occurrences of their children.

With the inferred co-occurrences, given an ambiguous term $\alpha$, the co-occurrence of $\alpha$ with a term $\beta$ will not be lower than $\alpha$’s co-occurrence with any of $\beta$’s descendants.

Meta-Data method (MD). As the third method for WSD a maximum entropy approach was used [20, 203]. Maximum entropy models have been successfully used in tasks like part of speech tagging, sentence detection, prepositional phrase attachment, and named entity recognition.

For the example in Figure 3.6, the MD method determines the meaning of ‘thrush’ by using meta-data for the document and then deciding based on what was previously learned about this meta-data from training examples. The occurrence of contextual words and phrases in a text together with a potentially ambiguous term can be seen as a random process. Maximum entropy modeling aims at modeling the behavior of this random process. Provided a large amount of training examples, the algorithm automatically extracts a set of relationships inherent in the examples, and then combines these rules into a model of the data that is both accurate and compact.

Algorithm. The training and test data in our case are sentences containing the potentially ambiguous term flagged with the sense. Each training example, one sentence each, is represented as a set of features. An implementation of the Porter stemmer was used [207] and as features n-tuples of word stems and meta information of the document was selected, such as the journal and title words and the publication period (10 years ranges).

The implementation\(^3\) takes a series of events to train a model. Each event is a configuration of binary relations associated with a label. The resulting model is applied to an unknown configuration of binary relations. The result is the predicted probability for the previously trained outcomes. MeSH terms already assigned to the articles are excluded, for the performance evaluation to be independent of them.

Given the abstract of a scientific article and the ambiguous term, the steps followed are:

1. extract binary features (n-tuples of word stems from different scopes - title, sentence, entire abstract -, publication period, journal title)
2. get scalar product of feature vector and model (vector based on training)

\(^3\)maxent.sourceforge.net
3. the result are the probabilities for predefined outcomes (in this case True or False)

4. if above a threshold 0.5, the term is True, else False.

As an illustrating example of the features extracted, articles mentioning ‘signal transduction’, ‘kinase’, ‘embryo’, ‘neuron’ or ‘stage’ are more likely to refer to ‘multicellular organismal development’ than to another sense, such as development of an algorithm or a disease in an organism. Articles mentioning ‘anxiety’, ‘behavior’, ‘memory’, ‘social’ and ‘fear’ are more likely to refer to ‘psychological inhibition’, instead of ‘enzyme inhibition’.

Experimental setup

Classification task and limitations. The disambiguation performed here is mainly a classification task; it comprises the decision whether an annotation is positive or negative with respect to the GO/MeSH sense. Instead of assigning one of the numerous different senses to a term, a positive or negative label is assigned to it, when it corresponds to the GO/MeSH sense or not, respectively. Acronym ambiguity was not handled separately. However, in cases where an acronym belongs to an ontology term label (e.g. FA for GO term ‘Fanconi Anaemia’ vs ‘Fatty Acids’, AMP as of MeSH term ‘Adenosine Monophosphate’ vs ‘Antimicrobial Peptides’, etc.), this is disambiguated in the same way as all ontology term labels.

As mentioned in the introduction, some disambiguation tasks are easier than others; “BANK” the building and the “BANK” gene will appear in completely different context, whereas the “BANK” gene, protein or mRNA are even likely to appear in the same article abstract, making the disambiguation task often difficult even for a domain expert. ‘Transport by air’ or ‘patient transport’ will be easier to distinguish from the GO sense of transport, but ‘transport of virus cultures’ will appear in a closer molecular biology context. Distinguishing between ‘transport’, ‘RNA transport’, ‘tRNA transport’ or ‘ion transport’ can become less difficult by using the hierarchical information in the ontology (e.g. exploiting subClassOf/subPropertyOf relations between ontology terms). Some terms are also easier to disambiguate in the same task, depending on the number of their different senses (see table 3.6) and the distance between them, the way they appear in text (e.g. some can be easily distinguished with the help of regular expressions) and the number of tokens they comprise (one-token terms are usually more ambiguous as they are more likely to correspond to common English).

The ambiguous terms examined are the GO terms: development, spindle, nucleus and transport and the MeSH terms thrush, lead and inhibition. Most of the different senses of the terms examined (see table 3.6) belonged as well to the biomedical domain, making the disambiguation task more difficult (e.g. development of a cell culture, development of a cytopathic effect, maturation–GO development). The limited number of terms examined is due to the labor-intensive process of manual collection of proper benchmark datasets. As mentioned in the introduction, there exist few annotated biomedical corpora for evaluation and depending on the task, researchers need to collect their own gold standard datasets. Datasets were collected for a list of ambiguous terms based on the amount of true/false data available and the frequency of occurrence in PubMed (2600 manually curated documents of high/medium curation quality for 7 selected GO and MeSH terms). The aim was to keep the ratio of true/false abstracts close to 1, giving a 50% chance to each appearance of the term to be true or false with respect to the GO/MeSH sense (although the ratios in Medline will be different per term). First the UMLS WSD collection [275] and later a list of common False Positive terms (based on curations in GoPubMed) was examined. From the UMLS WSD collection terms that were GO/MeSH terms and the senses provided were distant to
3.2. METHODS

<table>
<thead>
<tr>
<th>Term</th>
<th>Senses</th>
</tr>
</thead>
<tbody>
<tr>
<td>Development</td>
<td>biological process of maturation (GO); development of a syndrome/disease/treatment; cataract development; colony development; development of a method; staff/economic development; software/algorith development</td>
</tr>
<tr>
<td>Spindle</td>
<td>mitotic spindle (GO); sleep spindles; muscle spindle; spindle-shaped cells</td>
</tr>
<tr>
<td>Nucleus</td>
<td>cell nucleus (GO); body structure (UMLS, subthalamic/cochlear/caudate nucleus); aromatic nucleus</td>
</tr>
<tr>
<td>Transport</td>
<td>directed movement of substances into/out of/within/between cells (GO); patient transport (UMLS); transport by air; transport of virus cultures; maternal transport</td>
</tr>
<tr>
<td>Thrush</td>
<td>Oral Candidiasis (MeSH); songbird (e.g. thrush nightingale)</td>
</tr>
<tr>
<td>Lead</td>
<td>heavy metal (MeSH); lead measurement (UMLS); to result in inhibition</td>
</tr>
<tr>
<td>Inhibition</td>
<td>psychological/behavioral inhibition (MeSH); metabolic inhibition (UMLS); % inhibition (SNOMED)</td>
</tr>
</tbody>
</table>

Table 3.6: Ambiguous terms and their senses in the WSD datasets collected.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Development</td>
<td>False True</td>
<td>False True</td>
<td>False True</td>
</tr>
<tr>
<td>Spindle</td>
<td>98 111</td>
<td>271 56</td>
<td>2296 715</td>
</tr>
<tr>
<td>Nucleus</td>
<td>50 48</td>
<td>70 48</td>
<td>519 599</td>
</tr>
<tr>
<td>Transport</td>
<td>99 100</td>
<td>25 61</td>
<td>131 1336</td>
</tr>
<tr>
<td>Thrush</td>
<td>102 91</td>
<td>102 56</td>
<td>1043 699</td>
</tr>
<tr>
<td>Lead</td>
<td>17 83</td>
<td>45 7</td>
<td>35 1131</td>
</tr>
<tr>
<td>Inhibition</td>
<td>71 27</td>
<td>202 22</td>
<td>1564 735</td>
</tr>
</tbody>
</table>

Table 3.7: The above datasets contain manually collected PubMed articles by one expert (high quality / low quantity), manually curated articles by a group of non-experts (medium quality / medium quantity) and semi-automatically collected articles (low quality / high quantity).

For each other were selected, i.e. in the case of ‘lead’, the two senses with short semantic distance (compound; laboratory procedure of lead measurement) were considered as one, as they both are about the compound. A semantically more distant sense is that of the verb to lead/result in. Regarding the false/positive ratio limitation/criterion, for some terms this was not satisfied, not allowing the inclusion into the evaluation dataset. For example, for ‘transport’ the UMLS WSD collection contained 93 abstracts classified as sense1 (True for GO sense) and only 7 as other (curators in this collection had 3 options: sense1, sense2 or other, here sense1 as the biological transport and sense2 as patient transport). Therefore false examples containing other senses for a balanced corpus were manually collect.

Datasets. Three different benchmark datasets (see table 3.7) were collected to evaluate the performance of the three methods. They differ in quality and quantity, depending on their collection process (manual by one curator, directed manual by several curators, mainly automatic). The common reference dataset between the three methods is the manually annotated by a domain expert one:

Problem: high quality training data is rare
High quality, low quantity corpus: this corpus consists of ~100 true and 100 false example documents (abstracts) per ambiguous term. For the ambiguous GO terms examined and the MeSH term thrush both true and false examples were collected manually. True examples are abstracts that discuss, for instance, development, in the sense specified by GO. False examples also contained the ambiguous term, but in other senses, closer or not (see table 3.6). For the ambiguous MeSH terms lead and inhibition (psychology), the test set originated from the UMLS WSD corpus [275]. These two were the only terms depicting MeSH terms. All other terms in the UMLS WSD (such as growth, repair and reduction) were only found in GO or MeSH as substrings and would thus not be contained in either co-occurrence graph as single nodes.

Medium quality, medium quantity corpus: this corpus consists of documents for which the annotation has been manually confirmed by a group of expert and non-expert curators. Colleagues were asked to confirm or reject the automatic annotations (for GO and MeSH terms) provided by GoPubMed for a collection of article abstracts. This collection has been mainly automatically created, as described next (low quality, high quantity corpus). For each of the automatic annotations, the curators could select among three options: a. true and important for the context of the publication, b. of minor importance/relevance and c. false annotation. The curation tool is available via GoPubMed [64].

Low quality, high quantity corpus: this corpus was created mainly automatically. A similarity-based clustering of abstracts with literal occurrence of the ambiguous terms was implemented. Each abstract was matched to its nearest abstract, conceptualized as a directed edge from the former to the latter. Then every connected component was considered as a cluster. From an initial manual evaluation of the clustering results, clusters of size > 60 were consistent enough, meaning that articles in such clusters were referring to one sense of the ambiguous term in 72-95% of the cases. Each cluster’s abstracts were input into a system developed in-house (also used in Alexopoulou et al. [4]) that generated a list of terms describing each cluster based on term frequency inverse document frequency (TFIDF). The top 20 terms of the list were later evaluated by an expert which labeled the clustered articles as true or false for the respective GO/MeSH term. The above facilitated and accelerated the dataset collection process without any significant loss in data quality (compared to the gain of data quantity for benchmarking).

Experiment For evaluation and comparison purpose, each method’s performance was tested (in terms of precision, recall and specificity) on the high quality / low quantity dataset (see CS1-2, TC1-4 and MD1-3 in table 3.9). A classical stem co-occurrence analysis was also applied as a baseline on the same dataset; this consisted of basic maximum entropy modeling on stems without any use of meta-data or hierarchical information (see bME in table 3.9).

Each method’s performance was tested separately with different test datasets. For the ‘Term Cooc’ method (TC), the performance of co-occurrences of GO/MeSH terms and inferred co-occurrences of GO/MeSH terms (each one of the variants combined -or not- with Support Vector Machines) was tested in the three benchmark datasets described earlier, in order to evaluate the method in larger (but of lower quality) datasets, since it has been shown that sample size, sense distribution and degree of difficulty impact on the classification task [282]. Input to this method were the automatic annotations per article provided by GoPubMed (GO/MeSH terms and MeSH hand annotation) and the respective co-occurrence graph. As a side experiment, the TC method was tested for the disambiguation of MeSH terms without including the MeSH hand annotations in the automatic annotations provided by GoPubMed, to estimate how the quality of the input influences the quality of the results.
3.2. METHODS

Table 3.9: CS1 column contains the results (% $f$-measure) for the Closest Sense (CS) approach with the use of the classic distance (only subsumption). CS2 column contains the results for the CS approach with the use of the optimized signature together with the subsumption distance. TC1 and TC2 contain the results of the Term Cooc (TC) approach, when the co-occurrences or the inferred co-occurrences are used, respectively. TC3 contains the results for the TC approach with co-occurrences and support vector machines, and TC4 when inferred co-occurrences and SVMs are used. bME column contains the results for the baseline method (classical Maximum Entropy modeling of stems without meta-data or hierarchical information), trained and tested on the high quality corpus in a 5-fold cross validation. MD1 is for the Meta-Data approach, trained and tested on the high quality corpus in an initial experiment (MD1) as training and testing dataset, in a 5-fold cross validation. Then the medium quality and low quality datasets were separately used as training sets, with testing of the method on the high quality dataset (MD2 and MD3, respectively).

Experimental Results

The performance of the three disambiguation approaches (CS, TC, MD) and the baseline (bME) was tested on a common high quality / low quantity dataset. The overall results of this comparison are shown in table 3.9. All methods perform well between 73-96% average $f$-measure. In particular, the Meta-Data (MD1-3) approach is the best one: when trained on high quality data (MD1), it achieves 96% $f$-measure. When the meta-data are not used (baseline method, bME) the accuracy falls to 90%. The Term Cooc (TC1-4) method follows with 81% and the Closest Sense (CS) approach with 77% (80% for the optimized signature together with the subsumption distance, in CS2). All methods present low $f$-measure for ‘development’ and ‘lead’ (79% and 60% in average). The best results (in average for all methods) are obtained for GO terms ‘transport’, ‘nucleus’ and ‘spindle’ (88%, 87% and 85% respectively).

As far as the Closest Sense approach is concerned, there is a clear improvement in the results (from CS1 to CS2) with the use of the optimized signature together with the subsumption distance. For the Term Cooc approach, when the inferred co-occurrences are
taken into account (the scores are propagated to the parents of the terms, from TC1 to TC2) in the case of the GO terms the results remain the same, whereas in the case of the MeSH terms the results are worse, mostly in terms of recall (see supplementary material). For GO terms, the results are best when inferred co-occurrences are combined with SVMs (TC4, 79-98% $f$-measure), whereas for MeSH terms, the best $f$-measure (79-93%) is achieved when co-occurrences with SVMs are used, without the inferred co-occurrences (TC3). This difference can be explained by the different structure of the two hierarchies. GO can be described as “tall and thin” (few children per node, many levels, with maximum number of levels 19), whereas MeSH is “short and fat” (many children per node, not many levels, with a maximum of 9 for the version of 2007). Additionally, the relations between terms in MeSH are not exact is-a relations, but rather is-related_to. Therefore, propagating the term co-occurrences in MeSH does not improve the results, since it does not necessarily mean that annotating with term $MeSH_X$ also means all of $X$’s ancestors. On the contrary, in GO this is more likely to hold. The Meta-Data method gives - as expected - the best results. When the method is trained and tested on the same high quality test (with a 5-fold cross validation, see MD1), it results in an average $f$-measure of 96%. When trained on the medium quality (MD2) and low quality (MD3) corpora and tested against the high quality corpus, the $f$-measure decreases into 81% and 70%, respectively, which are nonetheless high, compared to the quality of the training sets. The high performance of the Meta-Data approach is mainly due to the use of meta-data as the title of the abstract and the journal. For example, for the terms ‘inhibition’ and ‘spindle’ it achieves 100% $f$-measure and for ‘nucleus’ 99%. The true sense of inhibition for MeSH is psychological inhibition, which is easier to disambiguate, since it will mostly appear in psychology/psychiatry journals. The same applies for ‘spindle’, which will mostly occur in cell biology and cell division/cycle journals.

Each method’s performance was tested separately with different test datasets. The ‘Closest Sense’ method was also tested on the NLM UMLS WSD Collection [275] to compare four versions of semantic distance computation in order to disambiguate term mapping to the UMLS semantic network. The experiment showed that the use of the ontology definition can improve significantly the precision. Over the 22 ambiguous terms examined, the overall average precision was 83%.

For the ‘Term Cooc’ method, the performance of the different variants (co-occurrences +/- inferred co-occurrences +/- SVMs) was tested in the three benchmark datasets described earlier (see Datasets), in order to evaluate the method in larger (but of lower quality) datasets. Testing the method from the highest toward the lowest quality (but higher quantity), the $f$-measure decreases only by 3-10%, indicating a consistent behavior of the method. As a side experiment, the ‘Term Cooc’ method was tested for the disambiguation of MeSH terms without including the MeSH hand annotations in the automatic annotations provided by GoPubMed, to estimate how the quality of the input influences the quality of the results. As expected, the results decreased dramatically (46%), indicating that the MeSH hand annotations provided per article are important for the disambiguation.

Discussion of methods

Overall, the MD method gave the highest $f$-measures among all methods. The results became worse for the medium and high quantity datasets, since these were of lower quality in terms of correctness. The MD approach’s consistency with giving the highest results is due to integrating meta-data, such as journal and title, which are representative of the true meaning of an ambiguous term. The MD approach needs plenty of labeled data for training.

When comparing the results of the TC and CS methods to the baseline method (bME) that performs only maximum entropy modeling of stems (without use of meta-data), bME
still gives better results, but this is due to the available training data of high quality. The disadvantage of MD and bME compared to TC and CS is the need of high quality training data.

The MD approach is less scalable in terms of storage demands as the number of articles increases, while the CS and TC approaches have constant storage demands (ontology and a co-occurrence graph).

In the TC method the SVMs increase the results up to 98%. The TC method requires an ontology and co-occurrence graphs. The origin of this graph should be a manually curated data source, in our case GOA and MeSH. The quality of the graph will heavily depend on its origin and quality of the data.

The inferred co-occurrences improve the results for GO, while for MeSH they get worse.

This is due to the different structures of the two semantic hierarchies; the ancestors of an applicable GO term are more likely to also be applicable to the same article, because of GO's structure that is “tall and thin”. But MeSH's structure is “short and fat” and is not always a thesaurus; not all of a node’s ancestors are also applicable.

Moreover, in the TC method the inferred co-occurrences only improve the result if combined with the SVM. This is because the inferred co-occurrences make the extreme co-occurrences with the ambiguous term, which the SVM uses for training, more representative of an ambiguous term’s true meaning.

The CS approach needs only a semantic hierarchy in the form of an ontology, and in this sense is the most automated of the three methods. Moreover, CS gives good results, where the only problematic term is ‘lead’. However, CS is sensitive to the design of the ontology or subdomain of UMLS used, which reflects the view of the designers. As shown by the accuracy of Humphrey et al. [128], Liu et al. [163], UMLS may not be the best choice to be used as background knowledge as the different parts of the hierarchy are modeled differently (MeSH, GO, SNOMED, etc.), resulting in different granularity. Different groups of people design ontologies differently; the various subdomains of an ontology will reflect the designers' views respecting depth, number of nodes, and structure. Therefore, the subdomains of the ontology influence the performance of the CS method, and the design rationale of the ontology may be ultimately responsible for performance differences on various terms. For example, ‘nucleus’ is a subtree root in both GO and SNOMED (anatomical structures); in GO there are 2000 descendants of nucleus, while in SNOMED 10.

Conclusions. Based on the results, meta-data and training data of high quality seem to be the key point for the increase of the accuracy. When such training data are not available - as happens in most of the cases - co-occurrence of ontology/taxonomy/thesaurus terms can provide the way to the right decision. Moreover, the hierarchy of the terms and the subdomain, when consistently modeled, can depict the correct sense of an ambiguous term.

The MD method produced the best results by including meta-data in the WSD decision, but it requires high quality training data. The most interesting thing about the TC and CS methods is that they are semi-automated, given a co-occurrence graph or ontology; then the training does not require manual intervention. TC requires well modeled ontologies such as GO, and deteriorates as the structure becomes less rigorous as in MeSH. CS requires large and consistently modeled ontologies, which are two opposing requirements. Thus, for TC and CS the structure of the ontology and subdomain affect the distance metric used and WSD quality. Future work will include identifying ambiguous terms for a certain corpus automatically. For this purpose, it is planned to employ WordNet, clustering, Part of Speech and noun phrase statistics, and expert input.

For TC and CS, it is assumed that the other terms in the context are correct and
independent of one another; in fact, they could also be ambiguous and therefore false. For CS it is planned to optimize the distance computation and propose other distances, taking into account existing annotation bases and ontology structure.

### 3.2.4 Context Models for Concept Disambiguation

The previous section shows that a maximum entropy classifier trained on high quality training data performs best on the task of sense disambiguation. This section shows how Context Models are used to disambiguate concept candidates.

#### Walkthrough of recognition pipeline

The Concept Recognition configured with PubMed as document source and the Gene Ontology as the source of defined concepts can be executed as follows.

**Document Retrieval.** The documents to be annotated are PubMed abstracts. The article with the PMID 17963238 is fetched because the user searched for "Tim10". Along with this article 44 other articles are fetched as well. The following is executed for each article.

**Terminology Processing.** The concepts to be identified are from the Gene Ontology. One concept defined in the Gene Ontology is DNA binding. The terminology of GO is preprocessed as described in section 3.2. The label of DNA bindings tokenized into two words: 'DNA' and 'binding'. The word 'DNA' is an abbreviation and has no stemming variants. The word binding has the stemming variants 'bind', 'binds' and 'binding'. Both word are very frequent word in the Gene Ontology. That is, 203 other concepts contain the word 'DNA' in the label and 540 other concepts contain the word 'binding' in the label. The mentioning of one of the word does indicate but does not ensure that the text refers to the concept DNA binding.

**Input Text Processing.** The document attributes of the citation 17963238 are fetched. The citation has the title Zinc binding of Tim10: Evidence for existence of an unstructured binding intermediate for a zinc finger protein. The journal it is published in is named Proteins. The article was published in 2007. The abstract and the title are segmented into sentences and words. Four content scopes are defined: word, sentence, paragraph and citation. The scope "citation" includes the journal name and the publication date.

The title and the sentences of the abstract are scanned for concept candidates. One candidate is the concept DNA binding as the sentence: Next, the stopped-flow fluorescence technique was used to investigate the kinetic process of the binding reaction. mentions a stemming variant of the word 'binding' in the label of the concept.

**Sense Disambiguation.** The curation database contains 251 positive and 163 negative training examples. This allows to distinguish the meaning of 'binding' in 82.9% of the cases correctly, see table 3.13. The trained models learned which features of a citation indicate the meaning DNA binding and which not.

Table 3.10 shows examples of features which can be extracted from the citation. For example in the scope “title” the stemmed word phrases 'zinc finger', 'protein' and 'unstructured bind' can be identified. The content of the scope “title” is the same for all words in the abstract. In contrast to this the scope “sentence” is different for words in different sentences in the abstract. In this case the word 'binding' is contained in a sentence with stemmed word...
3.2. METHODS

<table>
<thead>
<tr>
<th>Binary feature</th>
<th>Scope</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>zinc_finger</td>
<td>title</td>
<td>word phrase</td>
</tr>
<tr>
<td>protein</td>
<td>title</td>
<td>word phrase</td>
</tr>
<tr>
<td>unstructur_bind</td>
<td>title</td>
<td>word phrase</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>word phrase</td>
</tr>
<tr>
<td>stoppedflow_fluoresc_technique</td>
<td>sentence</td>
<td>word phrase</td>
</tr>
<tr>
<td>investig</td>
<td>sentence</td>
<td>word phrase</td>
</tr>
<tr>
<td>kinet_process</td>
<td>sentence</td>
<td>word phrase</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>word phrase</td>
</tr>
<tr>
<td>degree_of_fold</td>
<td>paragraph</td>
<td>word phrase</td>
</tr>
<tr>
<td>zinc_concentr</td>
<td>paragraph</td>
<td>word phrase</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>word phrase</td>
</tr>
<tr>
<td>Proteins</td>
<td>journal</td>
<td>title</td>
</tr>
<tr>
<td>2000-2010</td>
<td>year</td>
<td>number</td>
</tr>
</tbody>
</table>

Table 3.10: Examples of binary features extracted from the PubMed abstract below. The word phrase features were stemmed using the Porter Stemmer algorithm. The journal title is used as it is. The years are accumulated to decades, so the example falls in to the decade 2000-2010.

**TITLE:** Zinc binding of Tim10: Evidence for existence of an unstructured binding intermediate for a zinc finger protein.

**ABSTRACT:** ... Comparison between the results of CD and fluorescence studies showed that the zinc-binding reaction is not a simple one-step process. It involves formation of a binding intermediate that is structurally as unfolded as the apoTim10; subsequently, a degree of folding is induced at increased zinc concentrations in the final complex. Next, the stopped-flow fluorescence technique was used to investigate the kinetic process of the binding reaction. Data analysis shows that the reaction has a single kinetic phase at a low free Zn(2+) concentration (~1 nM), and a double kinetic phase at a high free Zn(2+) concentration...

**Journal:** Proteins 2007. (c) 2007 Wiley-Liss, Inc. (PMID: 17963238)

The cooccurrence of such word phrases is used as binary features in case the model was trained on the scope “sentence”. Other scopes found in the example include “paragraph”, “journal”, “year” and “citation”. Note that the scope “citation” includes all other scopes.

The outcome of the disambiguation depends on how the model was trained and which threshold was set. The confidence value is computed as described in section 3.2.2. In case the confidence value is higher than the threshold the text range of the word “binding” is marked up with a link to the concept DNA BINDING. This the given example it is correct to link the word binding to the meaning DNA BINDING in the Gene Ontology because zinc fingers are motifs in DNA- and RNA-binding proteins whose amino acids are folded into a single structural unit around a zinc atom.

Using Context Models to predict relevant articles

The idea here was to use Context Models to predict whether PubMed citations are relevant for curators of the Mouse Genome Informatics database. To further evaluate the performance of Context Models and to evaluate which feature are learned a context model was trained on data provided by the MGI database. The algorithm had only access to abstracts, titles, journal name, authors, etc., but not to full text. Later the learned features were examined.
Essentially, the algorithm learned from positive and negative examples the characteristics of relevant articles. Obviously, these involved a lot of mouse terminology, but there were not so obvious hints for good and bad articles, too. A list of 5000 abstracts of manually curated full text articles was used to train a model to predict whether the article is relevant for MGI’s research on mouse. The resulting models are large: They comprise some 80,000 tokens, which are judged for their relevance or irrelevance.

**Experiment 1:** From the ca. 120,000 PMIDs listed in the MGI corpus four 5000 ids across all years were randomly selected as a positive data set. As a negative data set, 5000 out of the 2,000,000 most recent PMIDs were randomly selected. The positive and negative datasets were both randomly split into training (85%) and test set (15%). So, there were: 4250 positive training examples and 4250 negative training examples and 750 positive and 750 negative test examples.

The learning algorithm learned from the training data set (8500 articles) and was evaluated on the test set (1500). Training and test sets were completely independent, they do not overlap.

The algorithm achieved a success rate of 94.1% (precision 91%, recall 97.5%). These are average values, since the above generation of test/training data was repeated multiple times.

The algorithm learned for 80,000 tokens how relevant they are. Some feature gave strong negative hints:

- tokens such as “health”, “manage”, “work”, “questionnaire”, “physician”, “sample”, “bioethical”, “speech”, “risk factor”
- years such as 1900-1910, 1920-1930

Other features gave strong positive hints:

- tokens comprising all kinds of variant and combination of mouse, but also knockout, +/-, disrupt gene, human nomenclatur, b cell, control type, erythroid
- journals such as Gene, Genet Res, Neurn, Eur J Immunol, Cytogenet Cell Genet, Genomics

As mentioned, there were 80,000 tokens used in the model overall. This made an accurate fingerprint of what MGI’s curators annotate and what they do not which uses terminology and meta information such as journal names.

**Experiment 2:** Further it was of high interest how sensitive the trained model is to identify abstracts which mention mouse, mice, or murine in the abstract, but which are not in the list of papers selected by curator previously? This was a difficult task, as these negative examples were much closer to the positive ones.

The classification of papers with the learned model gave a confidence value from 0% to 100% and in the above analysis, an article was labeled positive if it had a confidence score >50%. The question was, how do these values look for the negative examples with mouse in the abstract? The results show that 1260 predictions out of 4657 were below 50%, i.e. they were correctly labeled as negative examples.

One the other side, the 5000 positive articles from your PMID list had an average confidence score of 87% and only 250 out 5000, i.e. 5%, got erroneously misclassified as negative.

---

3.2. METHODS

**Experiment 3:** Finally the question was how many of the 120,251 PMIDs previously manual curated by MGI’s curators, that do not have “mouse”, “mice” or “murine” in the abstract or title were correctly identified nonetheless?

As a result 8807 out of 120251 articles did not mention “mouse”, “mice” or “murine” in the abstract or title and 8233 of them had a confidence >=50% and were thus correctly identified, i.e. 94% of the articles without the mouse keywords were identified nonetheless.

**Summary** The Maximum Entropy models take binary features of unigrams, bigrams and trigrams of words in the sentences and in the title. Additionally the journal title was used as a feature. Table 3.10 visualizes how the context of the word “binding” is computed in an PubMed abstract. Note that the journal title and the title words are treated separately from phrases in the abstract text. This allows title words to become more important than words from the abstract text if the training data indicates this. Journal title and publication year may help to distinguish vocabulary differently used in various research areas and over time. Manual MeSH annotations were not used as features although it is obvious they would support the disambiguation task greatly. The aim was to keep the models independent of manual pre-processing of the documents.

From experiments it can be concluded that (1) a context model is very good to find articles that are of interest for MGI’s research (i.e. very high recall of 97%). (2) When evaluated on random articles it has also very high precision (91%). (3) For very difficult cases, where mouse is in the abstract, but the article is nonetheless not relevant, the precision is not as good. (4) The context model also identified relevant articles (94% of all previously manually curated articles) that do not contain the keywords “mouse”, “mice” or “murine”. (5) Since the classification gives a confidence score it is possible to retrieve articles, sort them by confidence and thus ensure that most important ones come first.

To summarize the advantages of the Context Models:

- Context Models outperform approaches relying on the terminology’s structure
- Context Models make use of textual contexts of concept, e.g. words not defined in the ontology, as well as meta-information, e.g. journal title or publication years, in a single model
- Context Models need only few high quality training examples
- Context Models can be computed very efficiently
- Context Models are robust against varying granularity of ontologies
- Context Models can be used to rank citations by interest for a group of researchers

The disadvantages of the Context Models are:

- Each ambiguous concept needs separate training
- The training data must be of same/similar scope, e.g. abstracts vs. full texts
- The quality of training data must be high
- Context Models require additional memory in the range of several 100KBytes
3.3 Evaluation of Concept Recognition algorithms

In order to evaluate the performance of the Concept Recognition pipeline two benchmarks were used: (1) the BioCreAtIvE Gold Standard and (2) a benchmark based on curations made by authors of PubMed abstracts. The Concept Recognition pipeline uses a rule-based approach with information value of ontology words and employs trained context models for ambiguous words if available.

3.3.1 BioCreAtIvE Gold Standard

The BioCreAtIvE Gold Standard is provided as XML-like files containing 19,939 records of a protein mention in text along with some evidence of a GO concept. One problem was the non-well formed XML syntax. In fact there was no XML schema provided at all and some tags contained broken SGML. Many fixes were applied to make the files well-formed and all HTML entities were declared using a DTD. Unicode characters encoding Greek letters were mapped to their Latin correspondence, e.g. the unicode letter for \( \beta \) was mapped to the word “beta”.

Problem: the BioCreAtIvE I corpus is not well-formed

The benchmark contains records produced by the competing systems in the BioCreAtIvE workshop and were checked manually by two experts during the evaluation phase. To measure the performance of the GO Concept Recognition pipeline implemented for this work only records with precise GO evidences were used. As a result 1860 unique text passages mentioning on average 1.35 concepts confirmed by two experts were used as the benchmark set.

Figure 3.7 shows that the recognition performance at various threshold levels. The best performance is measured at a threshold for the confidence of 0.84. The measurements show that the recognition pipeline recovers 489 (26.3% recall) unique annotations from the dataset. The two best systems in the BioCreAtIvE workshop identified 303 and 301 protein-GO associations with precisions of 28.9% and 28.7% precision respectively. The precision of the implemented system was only 16.5%.

Problem: BioCreAtIvE corpus is geared toward protein annotation

The lower precision can be explained with a biased corpus toward a specific task, see also section 2.9.2 on difficulties of corpus-based evaluations. The systems in the BioCreAtIvE workshop were geared toward the identification of GO concepts describing a specific protein. In contrast to this the implemented pipeline identifies all GO concepts disregarding protein mentions in the text. Some general GO concepts which were obviously correctly identified by the pipeline such as CELL, LIGAND and MEMBRANE were not marked by the workshop’s curators. Therefore such general concepts account for many false positives and decrease the precision of the system. Table 3.12 shows the concepts most often flagged as a false positive matches in this measurement. The third column shows the text passages selected as evidence by the systems. The table shows that the curators were not focused on marking all GO concepts in the passages.

The ultimate application of the Concept Recognition algorithm is the automatic classification of literature abstracts for the ontology-based literature search, as discussed in the next chapter. A text is relevant for an ontology concept if it mentions a concept, a synonym or a descendant concept. It is not trivial to decide which concepts were originally meant by the author. The following section discusses a method to measure the performance of the implemented system more realistically with a new benchmark created with the support of original authors of PubMed articles. Additionally a modified performance measure was developed which reflects the task to classify abstracts according to mentions of concepts, synonyms or descendants.
### 3.3. Evaluation of Concept Recognition Algorithms

<table>
<thead>
<tr>
<th>Matches</th>
<th>GO concept</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>381</td>
<td>CELL</td>
<td>As postulated for fetal cells, BSDL lipolytic activity could be essential for the metabolism of lipids...</td>
</tr>
<tr>
<td>194</td>
<td>LIGAND</td>
<td>The B7 family members CD80 (B7-1) and CD86 (B7-2) are two principal ligands for CD28 and for CTLA-4 (CD152)...</td>
</tr>
<tr>
<td>113</td>
<td>MEMBRANE</td>
<td>Indeed, receptors for this peptide and related analogues were first suggested by binding of 125I-labeled rat NMU in rat uterus membranes...</td>
</tr>
<tr>
<td>95</td>
<td>RNA SYNTHESIS</td>
<td>Finally, the ability of the ELLs to regulate the rate of messenger RNA synthesis,...</td>
</tr>
<tr>
<td>94</td>
<td>REVERSE TRANSCRIPTION</td>
<td>Sequencing of fragments, obtained by mRNA reverse transcription and a...</td>
</tr>
<tr>
<td>90</td>
<td>TRANSCRIPTION</td>
<td>The shorter transcripts encoded a cytosolic form of the enzyme...</td>
</tr>
<tr>
<td>88</td>
<td>KINASE, putative ENDOPEPTIDASE...</td>
<td>The C-terminal region of ERK5 appears to regulate negatively its kinase activity...</td>
</tr>
<tr>
<td>82</td>
<td>BIOLOGICAL REGULATION</td>
<td>Superoxide and its derivatives are increasingly implicated in the regulation of physiological functions from...</td>
</tr>
<tr>
<td>65</td>
<td>GROWTH</td>
<td>ARF inhibits cell growth by interacting with MDM2,...</td>
</tr>
<tr>
<td>64</td>
<td>BIOSYNTHETIC PROCESS</td>
<td>This enzyme catalyzes the key step in vitamin A biosynthesis,...</td>
</tr>
</tbody>
</table>

Table 3.12: GO concepts most often marked as false positive matches in the comparison with the BioCreAtIvE benchset. The curators of this corpus focused only on concepts explicitly relevant for the named protein in the texts. Other general concepts were not marked as GO concepts. This results in a low measured precision on this corpus.

![recognition performance](image)

Figure 3.7: The figure shows the measured performance on the BioCreAtIvE corpus for different threshold parameter values. The optimal threshold parameter for Context Model algorithm is between 0.80 and 0.84.
CHAPTER 3. ALGORITHMS FOR CONCEPT RECOGNITION

3.3.2 Author Curations Benchmark

The authors of PubMed articles were asked to judge the quality of automatically annotated abstracts in the GoPubMed search engine. The result is a new benchmark comprised of 689 unique PubMed abstracts flagged with 18,356 relevant concepts.

The Curation Tool can be accessed online via the free biomedical search engine GoPubMed at www.gopubmed.org. Authors were provided with a link displaying the automatically annotated concepts next to the abstract text. Each item was rated by the authors as follows: (1) highly relevant for this article, (2) less relevant for this article and (3) not correctly identified as a mention in this article. Missing concepts were added via a search in MeSH and the Gene Ontology. Figure 3.8 shows a screenshot of the online tool integrated in the biomedical search engine GoPubMed.

For the successful training of context models a minimum of 100 training examples of high quality is necessary for each term. Table 3.9 shows decreasing performance for less than 100 training examples when disambiguating the term development. Table 3.13 shows the training data for GO concepts available after two month public accessibility of the curation tool. Concepts with less than 100 positive and negative examples have a lower disambiguation performance. The number of concepts that can be successfully trained is still low. One reason is that the Curation Tool did not emphasis on the curation of ambiguous concepts specifically. The 18,356 author’s hints distribute over 1,729 MeSH and GO concepts.

Local Term Alignment algorithm. In order to compare the recognition performance with the previous implementation used in GoPubMed [64] the Local Term Alignment algorithm, described in Doms [67], was evaluated on the authors’ curation data. Figure 3.11 shows the measured performance for threshold parameters between 0.60 and 1.00. The optimal value for the threshold parameter is 0.74. The algorithm achieves 35.8% precision and 85.7% recall on the authors’ curation data set. Often the authors did not mark general concepts to be relevant if a more specific concept was found to be relevant. An additional performance value was introduced to measure correctly recognized super-concepts.
3.3. EVALUATION OF CONCEPT RECOGNITION ALGORITHMS

<table>
<thead>
<tr>
<th>GO concept</th>
<th>positive</th>
<th>negative</th>
<th>disambig.</th>
</tr>
</thead>
<tbody>
<tr>
<td>MULTICELLULAR ORGANISMAL DEVELOPMENT</td>
<td>103</td>
<td>1374</td>
<td>98,1%</td>
</tr>
<tr>
<td>NUCLEUS</td>
<td>912</td>
<td>125</td>
<td>99,7%</td>
</tr>
<tr>
<td>DNA BINDING</td>
<td>251</td>
<td>163</td>
<td>82,9%</td>
</tr>
<tr>
<td>TRANSPORT</td>
<td>170</td>
<td>236</td>
<td>85,2%</td>
</tr>
<tr>
<td>SPINDLE</td>
<td>106</td>
<td>142</td>
<td>89,1%</td>
</tr>
<tr>
<td>SENSITIZATION</td>
<td>32</td>
<td>71</td>
<td>58,3%</td>
</tr>
<tr>
<td>SIGNALING</td>
<td>46</td>
<td>18</td>
<td>81,8%</td>
</tr>
<tr>
<td>MEMBRANE</td>
<td>48</td>
<td>10</td>
<td>52,5%</td>
</tr>
<tr>
<td>LEARNING</td>
<td>34</td>
<td>2</td>
<td>n.a.</td>
</tr>
<tr>
<td>PATHOGENESIS</td>
<td>28</td>
<td>1</td>
<td>n.a.</td>
</tr>
<tr>
<td>NURSING BEHAVIOR</td>
<td>2</td>
<td>27</td>
<td>n.a.</td>
</tr>
<tr>
<td>DEGRADATION</td>
<td>21</td>
<td>6</td>
<td>n.a.</td>
</tr>
<tr>
<td>SIGNAL TRANSDUCTION</td>
<td>20</td>
<td>7</td>
<td>n.a.</td>
</tr>
<tr>
<td>INTRACELLULAR</td>
<td>21</td>
<td>3</td>
<td>n.a.</td>
</tr>
<tr>
<td>EXTRACELLULAR</td>
<td>21</td>
<td>1</td>
<td>n.a.</td>
</tr>
<tr>
<td>INVASIVE GROWTH</td>
<td>7</td>
<td>13</td>
<td>n.a.</td>
</tr>
<tr>
<td>APOPTOSIS</td>
<td>18</td>
<td>1</td>
<td>n.a.</td>
</tr>
<tr>
<td>LOCALIZATION</td>
<td>15</td>
<td>3</td>
<td>n.a.</td>
</tr>
<tr>
<td>OOCYTE CONSTRUCTION</td>
<td>1</td>
<td>16</td>
<td>n.a.</td>
</tr>
<tr>
<td>EQUATOR SPECIFICATION</td>
<td>1</td>
<td>13</td>
<td>n.a.</td>
</tr>
</tbody>
</table>

Table 3.13: List of 20 most often curated GO concepts in a time period of two month. The last column shows the disambiguation performance measured using a 5-fold cross validation for Context Models if available. Many of the concepts provide not yet enough training data for disambiguation. Some disambiguation hints are very uneven providing almost only negative or positive hints. The Maximum Entropy models require a reasonable balanced set of training examples. A strategy to improve the system’s recognition performance is to focus the curation efforts of concepts with prevailing negative hints, e.g. the concept label NURSING BEHAVIOR. Such as miss-balance indicates systematic miss-classification of the concept. Concept labels with prevailing positive hints tend to be less ambiguous. For example the concept label APOPTOSIS.

Figure 3.9: The disambiguation performance for the concept DEVELOPMENT decreases with less training examples reaching a plateau at 100 examples.
CHAPTER 3. ALGORITHMS FOR CONCEPT RECOGNITION

Figure 3.10: Schematic view of induced concept sets: The outer triangle denotes the whole ontology. Top corner is the root concept $r$. Concept annotations $a_1$ and $a_2$ induce concepts included in all paths up to the root concept, visualized by the corresponding rectangles. The overlapping parts of manual curations induced by $c$ and annotations denote the set of induced true positive matches ($iTP$). Annotations not covered by induced manual curation denote the set of false positive matches ($iFP$). Unannotated manual curation denote the set of induced false negatives ($iFN$).

**Induced Performance Measures.** The ultimate application for the Concept Recognition algorithm discussed here is ontology-based search as discussed in the next chapter. For this purpose one goal is to maximize the number of correctly classified documents for any concept in the ontology. The user explores documents using the ontology hierarchy by filtering documents for concept mentions. A document is relevant for a selected concept if it mentions the selected concept, a synonym of it or one of its descendant concepts. For example, a document mentioning *Tuberculosis* is relevant for a user filtering for *Bacterial Infections* as well because *Tuberculosis* is a descendant concept of *Bacterial Infections*. The induced sets of concepts of a document are measured as follows:

$$iTP = A \cap K$$
$$iFP = A \setminus K$$
$$iFN = K \setminus A$$

where $A$ is the induced set of annotated concepts in the document by the algorithm and $K$ is the induced set of concepts marked as relevant for the document by the authors. An induced set of concepts $I$ is defined as follows:

$$I(C) = \bigcup_{x \in C} ancestors(x)$$

where $C$ is a set of concepts and $ancestors$ is a function returning all concepts part of any path to root concept of an ontology. The schema in figure 3.10 visualizes the relations between the true positive matches, false positive matches and false negative matches for a document.

The induced performance measures are defined as follows:

$$iRecall = \frac{|iTP|}{|iTP \cup iFN|}$$

$$iPrecision = \frac{|iTP|}{|iTP \cup iFP|}$$
3.3. EVALUATION OF CONCEPT RECOGNITION ALGORITHMS

\[
\text{induced } f - \text{measure} = \frac{2 \cdot i\text{Recall} \cdot i\text{Precision}}{i\text{Recall} + i\text{Precision}}
\]

The authors’ curations were used to evaluate the Local Term Alignment algorithm with the induced performance measures. Figure 3.11 shows the results of this benchmark measuring 51.9% induced precision and 87.7% induced recall for a threshold value of 0.74 with an maximal induced \(f\)-measure of 65.2%.

The results show that the measured performance is lower than previously measured in Doms [67] (81.4% recall and 89.5% precision). This is an expected outcome as the authors of PubMed abstracts tend to be critical about vague annotations. Some authors even tend to emphasize the highly relevant concepts by marking all other concepts as not relevant for the article. The precision is therefore lower than previously measured. The recall value is actually slightly higher than previously measured.

The newly introduced induced performance measure reflects the impression of a user of an ontology-based search engine better than the standard measures. The user of such a system explores articles beginning from the root level. It is therefore of less impact on the users impression if a Concept Recognition algorithm identifies a concept which is very close to the perfect match as long as the two concepts share a close ancestor. For any concept above this common ancestor the classification is correct. The user will notice the error only when he/she explores the most specific concepts.

The induced performance measure favors algorithms recognizing specific concepts deep in the ontology correctly. An error on a deep level in the ontology generates more false positive matches as the set of induced concepts is larger. On the other hand it tolerates missing general concepts if more specific concepts are descendants of them.

The above results were taken as a baseline to evaluate the performance of the newly implemented Concept Recognition using rules and context models.

Evaluating the Context Model approach. In order to compare the performance of the new Concept Recognition algorithm the same benchmark data was used and the same evaluations were done. Figure 3.12 shows the results obtained for the different threshold levels. A threshold parameter of 0.80 resulted in an optimal \(f\)-measure of 73.5% with a precision of 72.8% and a recall of 74.2% and an induced \(f\)-measure of 76.1% (79.9% induced precision and 72.7% induced recall).

The Context Model approach outperforms the baseline approach by 1.45 times when comparing the standard \(f\)-measures. Applying the induced performance measure the performance is 1.16 times better. The performance increase is mainly due to the increased precision of the system (35.8% vs. 72.8%). The recall values are slightly lower than measured for the baseline approach. One reason for this is that the new approach favors precision which usually decreases recall. However the overall performance gained was significant.

Evaluation of Concept Revisions. To assess which measures contributed to the improvement of the recognition performance at first a very simple pipeline was configured: The GO labels were loaded and no statistics about word frequencies were used to compute the word information value. Each word was weighted equally important. All stems recognized by the Porter algorithm were accepted. Figure 3.13 shows the lowest performance for this configuration with an overall \(f\)-measure of 47.0%. The next configuration computed global word information values for all words in GO as discussed in section 3.2.1. Additionally the “Trailers” revision which truncates all subclauses from the labels was applied. This configuration resembles the idea of the Local Term Alignment method best. The set up achieved comparable performance to the baseline method (Local Term Alignment) on
Figure 3.11: Recognition performance of the Local Term Alignment algorithm on the author curations benchmark. The best performance is measured at a threshold level of 0.74. The induced precision is generally higher than the standard precision. The reason is that missing ancestors of manual curations are not counted as false positives matches. Also the induced recall is higher than the standard recall. The reason is general concepts with a low confidence are less likely counted as false negative matches if specific descendant labels with higher confidence are mentioned.

Figure 3.12: The figure shows the performance of the Concept Recognition pipeline on author curated PubMed abstracts. The best performance is achieved at a confidence threshold level of 0.80. The induced precision always higher than the standard precision. Induced recall does not much differ from the standard measure. The recall decreases dramatically for threshold level above 0.96.
3.4. SUMMARY AND DISCUSSION

The authors’ curation benchmark with 49.1% f-measure (precision: 36.5%, recall: 75.0%), the baseline approach (LTA) achieved 50.4% (35.8% precision and 85.7% recall as discussed earlier in this section).

The next configuration executed a syntactic analysis of the label words an added part-of-speech information for each word. A number of stem revision was then applied to the concept labels all relying on the POS tags. Wrong word variants for determiners were filtered by the “Determiner” revision and the “Prepositions” revision filtered false words mapped to prepositions. The word value of determiners and prepositions was set to a minimum. The goal was to replace stopword lists mostly containing such words. Unknown or unusual inflections which were not known by the WordNet dictionary were filtered for the word categories verbs, nouns and adjectives. This configuration achieved 67.8% f-measure (precision: 59.6%, recall 78.6%).

The best performance was achieved when applying word disambiguation models using contextual meta-information. The f-measure was 73.5% with a precision of 72.8% and a recall of 74.2%. While the precision was improved the recall slightly dropped. One reason is that the trained models are imperfect. The overall performance was nevertheless better.

![Figure 3.13](image.png)

Figure 3.13: The figure shows the average precision at eleven standard recall levels for the concept recognition using various revision of the concept labels. The lowest performance was achieved when weighting all label words equally important. Using the global word information value compares with the baseline algorithm the Local Term Alignment. The stem revisions remove inflection variants often leading to false positive candidates. Finally the the context models help improving precision at the expense of recall.

3.4 Summary and Discussion

The objective was to design an efficient method to annotate biomedical literature abstracts with concepts of a biomedical ontology.
CHAPTER 3. ALGORITHMS FOR CONCEPT RECOGNITION

The many participants of the BioCreative workshop Task 2 in 2004 show the significance of this task. The best systems had a high precision but extremely low recall measures. One outcome was that the ideas are complementary. The developed system is a hybrid approach which uses recognition patterns for unambiguous concepts and a machine learning approach for sense disambiguation.

A Concept Recognition framework was developed. The framework consists of separate processing modules named taggers which can be joined into textmining pipelines. To facilitate development, test and benchmarking of the modules and pipelines a scripting environment was developed. Pre-processing modules include taggers such as a bracketing aware tokenizer, a rule-based sentence splitter, a wrapper for different part-of-speech taggers, a statistic module computing the importance of words in an ontology and two GO specific heuristics.

The result are recognition patterns for ontology concept labels which can be loaded into a recognition engine. The engine uses all patterns to produce concept candidates. A caching strategy was developed to re-use solutions gained from previously solved word alignment problems. Corpus statistics and concept specific heuristics are used to reduce the number of false candidates early and to rank the candidates.

While most concept labels in the biomedical domain are unambiguous a number of concepts have overlapping meanings in other domains. Three word sense disambiguation methods were compared. The Closest Friends method uses short distances to other concept labels detected in the same text to distinguish the sense of ambiguous concept labels. The Best Friends method needs training data a uses statistics about cooccurrence of concept labels in texts to infer the right meaning. The third method performs best and needs training data as well. Maximum Entropy models were trained taken binary features such as groups of word stems in the same sentence or abstract, title words or the journal name. High quality training data of about 100 positive and 100 negative allow produce disambiguation performance well above 90%.

The performance of the Concept Recognition pipeline was compared with approaches submitted to the BioCreative competition. The same benchmark data was used. The implemented system recovers more GO concepts from the text passages than the best system in the workshop. The precision was slightly lower. The reason is the biased corpus toward the protein annotation task.

Because the BioCreative corpus was geared toward the task of protein annotation and was executed on full text articles the absolute numbers of the evaluation do not reflect the recognition performance on PubMed abstracts as provided by the NCBI search engine. With the help of authors of PubMed abstracts a new benchmark was created. An online web tool implemented by Transinsight GmbH and integrated into the free biomedical search engine GoPubMed helped to collect the curation data for 689 abstract with 18,356 concept curations. An evaluation on the authors benchmark showed that the recognition performance was improved by 25.7% achieving 79.9% induced precision and 72.7% induced recall compared to the previous approach which is similar to Couto et al. [57].

The curation data was used to train Maximum Entropy models using binary features such as journal title, title phrases and abstract text phrases. Employing a 5-fold cross validation models of ambiguous concepts were trained and used to disambiguate concept candidates during the recognition phase. A 13.2% increase in precision with a minimal loss in recall of 4.4% was achieved. More training data is expected to increase the precision even more. The low number of trainable models is due to the short time the online curation tool is available and the fact that the curation effort was not focused on the difficult concepts. There are currently only 209 concepts in MeSH and GO which have more than 4 reported false annotations. This suggests that with a reasonable amount of resources the concepts
3.4. SUMMARY AND DISCUSSION

most often seen as false annotations can be eliminated. If the efforts can be focused on those concepts and the same rate of participation of authors can be assumed the collection of the curation data takes two more month.

The recognition time per abstract was about 150ms. Which allows an annotation process on the fly. The outcome of this work is a crucial part in an ontology-based literature search system. The quality and completeness of the automatic ontology concept annotations determine the quality of an ontology-based search. The idea of ontology-based search is described in the following chapter.

Outlook. In Lambrix et al. [155] the authors use SVM-based classifiers to assign multiple ontology concepts to one PubMed abstract. The goal is to align two different ontologies by defining a similarity measure between concepts based on relationships between the documents in which they are used. An interesting study could use the Concept Recognition described in this chapter for ontology alignment.
Chapter 4

Ontology-based Literature Search

The goal was to develop reasoning algorithms for mining information from text based on semantic markups in literature abstracts. In contrast to classical searches the aim was to provide a table of content outlining the whole result set and providing links to subsets of documents grouped by topics. The goal was also to link bibliometric analyses to ontology-based literature search to discover trends, prominent authors and research institutions as well as important journals for single research topics. The contribution made in this chapter can be grouped into four stages:

1. Establish semantic links: This chapter shows how to use the algorithms developed in the previous chapter and the background knowledge of ontologies to answer questions.
   - It was shown that ontological background knowledge can be used for query expansion.
   - A data structure specifically optimized for minimal memory usage with simple API for text processing was designed.
   - A text processing component framework was designed which assembles taggers into pipelines. The input and output are structured texts.
   - Components named DocumentAnnotators were implemented and used to create more than 412 million semantic markups in PubMed abstracts.
   - Ontology Induction was formally defined as a method make explicit links implicit knowledge in PubMed abstracts.
   - Ontology Induction is a potentially resource intensive task. A heuristic was developed to reduce the computational costs.

2. Recover implicit knowledge: The induced ontology contains a large amount of useful information. This chapter shows how reasoning over background knowledge can be used make implicit knowledge in PubMed abstracts explicit.
   - Two criteria were developed to identify informative paths and nodes in the induced ontology.
   - Experiments showed that a simple tf-idf measure is most useful to rank informative concepts.
For the ranking of informative paths the hierarchy of the background knowledge is useful.

It was shown that it is useful to rank documents of the same aspect according to the proximity of query keywords and concept mentions.

(3) Question Answering: This chapter shows that ontology-based search with GoPubMed is able to answer real biomedical questions with a minimum of user interactions in a web browser.

It is shown that biomedical questions of the TREC Genomics Track can be answered with the induced ontology and the criteria for informative paths and concepts.

A prototype of a fully automated question answering system was implemented. It is shown that simple question types can be answered with this approach using ontology induction.

28 questions given by the TREC Genomics Track 2006 were answered with the free biomedical search engine GoPubMed. The answers were compared with answers manually curated by domain experts given by other question answering systems.

(4) Bibliometric Analysis: This chapter presents the first ontology-based, large scale, online available, up-to-date bibliometric analysis for topics in molecular biology represented by GO concepts.

It is shown how ontology-based literature mining can be used to identify trends in the literature for ontology concepts.

It is also shown that the same analysis is useful for keyword queries to the GoPubMed search engine.

It is shown that the method is in line with existing, but often out-dated, analyses.
4.1 Introduction

When users search they have questions in mind. Answering questions in a domain requires the knowledge of the terminology of that domain. Classical approaches to search do not make use of background knowledge during search. Section 2.2 introduces terminologies and ontologies as background knowledge created in the life sciences. Section 2.6.1 introduces literature databases in the life sciences.

This chapter describes a new approach that uses ontological background knowledge when mining a literature corpus to answer biomedical questions. It is shown that the background knowledge can be used to find more relevant documents and to organize the results in order to focus on important aspects. The ultimate goal is to answer biomedical questions with a minimum of user interactions.

The process of sending a query to a system which responds with a collection of relevant documents for the query is named Information Retrieval, see section 2.4.1 for a comparison of systems for Information Retrieval. The goal is to maximize the precision of the results, by minimizing the number of irrelevant documents presented to the user, and to maximize the recall, by minimizing the number of missed relevant documents, see section 2.9 for a discussion of evaluation methods.

After retrieving relevant documents the user is interested to collect specific information from the documents. Ultimately the user wants to answer a question and thereby acquire knowledge. It can be assumed that the understanding of data and information is a prerequisite for acquiring new knowledge. Section 2.4.2 discusses the relation between data, information, knowledge and understanding. The idea is to organize search results using the structured background knowledge of ontologies, such that its intention is to be useful.

The background knowledge can be used to improve recall during document retrieval. One way of maximizing the recall is a technique named query expansion. The idea is to preprocess the user query and add synonyms to it. PubMed for example expands the user query "mouse" into "mouse or mice". However it does not include synonyms such as “mus musculus”, “mus domesticus” or “murine” nor does it include the 29 subconcepts known by the Medical Subject Headings, introduced in section 2.2.2. Query expansion typically increases the recall to the expense of the precision as more irrelevant documents are retrieved on the side.

The user is interested in a high precision, in other words the number of documents that have to be read should be small. Systems with a high precision tend to have lower recall. The algorithm described in chapter 3 aims at optimizing both recall and precision by maximizing the harmonic mean between the two measures, named $f$-measure, read section 2.9.1 on evaluation measures.

Section 2.4 characterizes existing biomedical search engines and compares features focusing on Information Retrieval, see table 2.13, and Knowledge Retrieval, see table 2.15. While many of the existing systems use query transformation techniques non of the systems use ontological background knowledge to rank literature abstracts and to explore document sets.

It is assumed that the user wants to find answers to questions which actually can be answered with text passages from the documents or a subset of the concepts in the background knowledge. It is further assumed that the background knowledge and the documents corpus deal with the same domain. Answers to questions which are short text passages of literature abstracts are named citation answers, answers which can be given in the form of a collection of ontology concepts and their definitions are named glossary answers. The idea in both cases is to reason over the relations captured in the background knowledge to answer questions.
CHAPTER 4. ONTOLOGY-BASED LITERATURE SEARCH

Citation answers. For example the user wants to know how aspirin works. It is possible to answer this question by providing one or more statements if the corpus contains documents mentioning the drug and an adequate concept of the background knowledge in proximity. The user does not need to know this concept nor its synonyms. The entry point to answer the question will be the categories of the background knowledge. Every level of the hierarchically structured knowledge comprises potential answers. The more familiar a user is with the domain the deeper he or she can select an entry point.

One possible scenario to answer the question is: the user sends the keyword "aspirin" to the system. The system shows a table of content comprising identified concepts in the retrieved documents. The user selects an appropriate concept among the displayed categories in the table of content. An appropriate concept to answer the above question is BIOLOGICAL PROCESS, a rather general concept in the biomedical field. Users familiar with the biomedical domain may select a more specific concepts such as REGULATION OF CELLULAR METABOLIC PROCESS. The system now displays highlighted text passages mentioning the keyword and the selected concepts and its descendants. Possible outcomes for the above question are:

Aspirin exerts its unique pharmacological effects by irreversibly acetylating a serine residue in the cyclooxygenase site of prostaglandin-H(2)-synthases (PGHSs). (PMID: 18242581)

NO-donating aspirin inhibits the activation of NF-kappaB in human cancer cell lines and Min mice. (PMID: 18174252)

In the first passages CYCLOOXYGENASE (PATHWAY), a biological process, is mentioned. In the second passages the ACTIVATION OF NF-KAPPA B, a regulating process in the cell, is mentioned. Aspirin has various effects on biological processes in an organism. Both answers represent correct and useful answers to different audiences depending on their background.

Glossary answers. A user is interested in learning which diseases are associated to a well known virus. In this scenario documents with mentions of any disease in proximity of the virus are relevant. The answer to the question would be a list of diseases linked to the text passages stating the relation to the virus. The answer might be a ranked list of diseases sorted by the frequency of cooccurrence in the literature.

For the query "HIV" a system might identify the following passage in a publication:

Hepatitis B, C seroprevalence and delta viruses in HIV-1 Senegalese patients at HAART initiation (retrospective study). (PMID: 18551596)

Hepatitis is disease defined in the MeSH terminology. MeSH defines 6527 descendants of the concept DISEASE. The citation shows a close relation between HIV infection and this disease.

The following section describes algorithms developed to find answers to biomedical questions based on a semantically annotated literature corpus using ontological background knowledge.

4.2 Methods

For the studies carried out in this work texts from the public citation database PubMed were used, section 2.6.1 gives an introduction on PubMed. Furthermore the publicly available Gene Ontology and MeSH, introduced in section 2.2.2, were used as background knowledge. The algorithms presented here are in principle not limited to the biomedical domain. However the background knowledge and the document corpus need to cover the same domain and
4.2. METHODS

terminology. GO and MeSH were specifically developed for the biomedical domain which is covered by the PubMed citation database.

4.2.1 Query expansion with terminologies

When users formulate search queries they do not bear the entire terminology of the domain in mind. A user searching for "programmed cell death" might be unaware of the Latin equivalent "apoptosis". A biologist searching for a gene named "BRAC1" can easily miss one of the gene's six synonyms when googling. It is impractical for a person interested in the biological process of cell cycle to include all 539 descendant concepts defined in the Gene Ontology. A doctor investigating eye diseases may be interested in all reports mentioning one of the 253 subconcepts of eye diseases listed in MeSH. A keyword search including only the concept label will not retrieve all documents relevant for the topic.

Solution:
PubMed's build-in query expansion. Keyword search without synonyms and specialized terminology results in a low recall, relevant documents will be missing in the result. One approach to increase the recall is called query expansion. In the case of PubMed a query expansion is applied by default when the user searches via the web-interface. The NCBI maintains translation tables which replace parts of queries with additional synonymous labels. PubMed's query expansion is based on the MeSH terminology but not on its tree structure. Search terms are matched against translation tables for MeSH headings, journals, author names and collaborators. A PubMed query can be issued via the web-interface by a user or via the web-service provided by the NCBI. If the user enters an entry term for a MeSH term the translation will also include an "all fields" search for the MeSH term associated with the entry term. For example, a search for odontalgia will translate to: “toothache”[MeSH Terms] OR “toothache”[All Fields] OR “odontalgia”[All Fields] because Odontalgia is an entry term for the MeSH term Toothache. MeSH term mappings that include a standalone number or single character do not include a mapping for individual terms in a phrase, e.g. Protein C will not include Protein[All Fields] or C[All Fields]. The NCBI maintains the translation tables manually. If a translation seems not accurate for a search the user can inform the NLM Help Desk. The maintenance of the translation tables requires human resources. To shows the effect of the default query expansion. PubMed's default mechanism was bypassed. Table 4.2 shows keyword searches in PubMed with and without query expansion.

Table 4.2 shows that query expansion serves for a large percentage of documents which otherwise had been missed. Interestingly shows the two queries "alzheimer" and "aspirin" the same numbers for querying with and without expansion. The reason is that both headings have no expansions defined in the NCBI's translation tables although MeSH knows 16 synonyms for Aspirin and 13 synonyms for Alzheimer Disease. In the case of "aspirin" 1,714 documents were not found without searching for its synonyms.

Another problem is that the license agreement for the PubMed web-services allows to send one query every 3 seconds to the system. This usage limitation is a bottleneck for large scale studies. A solution is to setup a separate full text index containing all PubMed citations. A mechanism for query expansion independent of manually maintained translation tables is required then.

The goal was to design a query expansion mechanism which increases the recall for queries containing concept labels of biomedical ontologies. No additional translation tables should be used and no additional manual maintenance of translation tables should be required. The hierarchical vocabularies used for this experiment were the MeSH and the Gene Ontology.

The idea is to identify concepts in the query string and to add a disjunction of synonyms and descendant concepts defined in the background knowledge. To test the results of this
query transformation the resulting document were compared with the results returned by PubMed with the default query expansion using manually maintained translation tables.

The question was whether a query expansion can be implemented using the terminological knowledge of ontologies automatically. An experiment based on the open source fulltext index engine Lucene was set up. Table 4.4 shows the same queries as in table 4.2. The numbers show that similar results can be obtained by expanding keyword queries using the all synonyms of the descendants and the concept itself. The size of the resultsets are larger for all queries. This suggests that the PubMed results are a superset of the results returned by the Lucene index.

Users do not bear the entire terminology of a domain in mind when they formulate search queries. During Document Retrieval this can result in low recall values because documents only mentioning synonyms are missed during a simple keyword search know from web search engines. Classical web search engines do not return document mentioning only apoptosis when searching for "programmed cell death". Query expansion is employed by some search engines like PubMed. However the NCBI maintains translation tables for terms. Another restriction in using the NCBI webservice is that the number of request is limited.

This section showed that it is possible to replace the query expansion mechanism of PubMed by using a full text Lucene index and the synonyms and descendend concepts stored in MeSH and GO. The result are collections of PMIDs of documents containing keywords. In order to use the relations between concepts stored in the background knowledge the content of the documents needs to be semantically marked up. The following section describe the Concept Recognition pipeline developed for this work.

4.2.2 Structuring documents

PubMed serves literature abstracts in XML format. Each document is partitioned into attributes such as title, abstract, journal name, author names, publication date, see table 2.17 in section 2.6.1 for a list of document attributes in PubMed articles. Document attributes can be grouped into text attributes, such as title and abstract text, single-value attributes such as journal name and PMID, and multi-value attributes such as author names. PubMed’s title and abstract attributes are not further structured into lexical or syntactical units.

Ackoff [1] defines information as data that has been given meaning by way of relational connection. Raw text does not contain relational information. It needs to be interpreted by the human reader. The textmining techniques, as introduced in section 2.7, establish relations in between phrases of a text and between entities and external databases. A prerequisite for textmining is the markup of basic lexical units, such as words, syntactical units, such as sentences, and structural units, such as paragraphs, titles and abstracts.

Literature databases serve documents of varying granularity, e.g. PubMed Central serves the full text including the abstract, the background, methods and conclusions section as well as tables and figures. PubMed on the other hand serves only the title and abstract text. While all texts can be segmented into words and sentences the full texts provide more scopes. Some textmining methods make assumptions about the scope of a text passage. The abstract for example typically contains only the main points of the publications, such as the techniques used and the achieved results, while the background section will mention related approaches including techniques and results of other works.

The objective was to implement an efficient data structure specifically optimized for textmining pipelines. Lexical, syntactical and structural information as well as markup information must be easily accessible and modifiable. The data structure should allow different sequences of application of textmining modules, e.g. it must allow to first identify names
## 4.2. METHODS

Table 4.2: A selection of PubMed queries with and without the query expansion applied by the NCBI. The differences in the size of the result set are large in most cases. Only "alzheimer" and "aspirin" return the same numbers with and without expansion. Both headings have no expansions defined in the NCBI’s translation tables although MeSH knows 16 synonyms for Aspirin and 13 synonyms for Alzheimer Disease.

<table>
<thead>
<tr>
<th>Query</th>
<th>Expanded query</th>
<th>default</th>
<th>without expansion</th>
</tr>
</thead>
<tbody>
<tr>
<td>mouse</td>
<td>&quot;mice&quot;[MeSH Terms] OR &quot;mice&quot;[All Fields] OR &quot;mouse&quot;[All Fields]</td>
<td>952,015</td>
<td>389,173</td>
</tr>
<tr>
<td>human</td>
<td>&quot;humans&quot;[MeSH Terms] OR &quot;humans&quot;[All Fields] OR &quot;human&quot;[All Fields]</td>
<td>10,476,155</td>
<td>1,678,825</td>
</tr>
<tr>
<td>programmed cell death</td>
<td>&quot;apoptosis&quot;[MeSH Terms] OR &quot;apoptosis&quot;[All Fields] OR (&quot;programmed&quot;[All Fields] AND &quot;cell&quot;[All Fields] AND &quot;death&quot;[All Fields]) OR &quot;programmed cell death&quot;[All Fields]</td>
<td>151,308</td>
<td>10,076</td>
</tr>
<tr>
<td>aspirin</td>
<td>&quot;aspirin&quot;[MeSH Terms] OR &quot;aspirin&quot;[All Fields]</td>
<td>42,311</td>
<td>42,311</td>
</tr>
<tr>
<td>alzheimer disease</td>
<td>&quot;alzheimer disease&quot;[MeSH Terms] OR (&quot;alzheimer&quot;[All Fields] AND &quot;disease&quot;[All Fields]) OR &quot;alzheimer disease&quot;[All Fields]</td>
<td>47,589</td>
<td>47,589</td>
</tr>
<tr>
<td>bioinformatics</td>
<td>&quot;computational biology&quot;[MeSH Terms] OR (&quot;computational&quot;[All Fields] AND &quot;biology&quot;[All Fields]) OR &quot;computational biology&quot;[All Fields] OR &quot;bioinformatics&quot;[All Fields]</td>
<td>34,301</td>
<td>16,353</td>
</tr>
<tr>
<td>breast cancer</td>
<td>&quot;breast neoplasms&quot;[MeSH Terms] OR (&quot;breast&quot;[All Fields] AND &quot;neoplasms&quot;[All Fields]) OR &quot;breast neoplasms&quot;[All Fields] OR (&quot;breast&quot;[All Fields] AND &quot;cancer&quot;[All Fields]) OR &quot;breast cancer&quot;[All Fields]</td>
<td>192,442</td>
<td>110,309</td>
</tr>
<tr>
<td>liver transplantation</td>
<td>&quot;liver transplantation&quot;[MeSH Terms] OR (&quot;liver&quot;[All Fields] AND &quot;transplantation&quot;[All Fields]) OR &quot;liver transplantation&quot;[All Fields]</td>
<td>55,370</td>
<td>37,112</td>
</tr>
</tbody>
</table>

Table 4.4: Same queries as in table 4.2 expanded with synonyms and descendant terms. The last column shows the size of the search result using all synonyms of the descendants and the concept itself.

<table>
<thead>
<tr>
<th>Query</th>
<th>synonyms and descendants</th>
<th>resultset</th>
</tr>
</thead>
<tbody>
<tr>
<td>mouse</td>
<td>159</td>
<td>968,391</td>
</tr>
<tr>
<td>human</td>
<td>5</td>
<td>10,606,253</td>
</tr>
<tr>
<td>programmed cell death</td>
<td>693</td>
<td>250,602</td>
</tr>
<tr>
<td>aspirin</td>
<td>18</td>
<td>44,025</td>
</tr>
<tr>
<td>alzheimer disease</td>
<td>14</td>
<td>62,434</td>
</tr>
<tr>
<td>bioinformatics</td>
<td>5</td>
<td>34,772</td>
</tr>
<tr>
<td>breast cancer</td>
<td>26</td>
<td>197,382</td>
</tr>
<tr>
<td>liver transplantation</td>
<td>6</td>
<td>55,455</td>
</tr>
</tbody>
</table>
and abbreviations and later sentences and words or vice versa. A simplifying limitation was that no partial overlapping of markups were allowed.

A possible approach to structure text is to represent it as a connected acyclic simple graph. The text encoding projects DocBook and TEI, introduced in section 2.6.3, define XML grammars for structuring texts. Such a data structure can easily be implemented as a DOM (Document Object Model). Several implementations of DOM trees are available. However the available DOM implementations are not optimized for textmining. The Java implementation developed for this work specifically optimizes the addressing of single characters and iterate forward and backward over nodes and axes, while the memory overhand is minimal. Figure 4.1 displays a tree structure segmenting a text into an abstract, sentences and words.

![Figure 4.1](image)

Using this data structure a PubMed citation is represented by a root node comprising all document attributes. The document attribute abstract is further segmented into sentences. The title is tagged as a separate sentence. Such a data structure can be traversed conveniently. (1) A typical way of traversing is for example: visiting all nodes beginning from the root node which are tagged as a sentence. This would iterate over the title and all sentences in the abstract of a PubMed document. (2) A way of accessing all lexical units of a sentence would by traverse all leaf nodes of a sentence which are not tagged as whitespace. (3) For a given word it is easy to lookup an attribute of the document it is mentioned in. A disadvantage of not using standard DOM implementations is that third party query engines can not be used out of the box. However the memory efficiency and simple API was favorable.

**Taggers.** The textmining framework for this work was designed as a pipeline of processing modules named taggers. Taggers annotate text structures by inserting nodes or adding attribute to text nodes. Taggers insert new nodes by addressing a character range and adding a tag. Partially overlapping annotations are not allowed. While this clearly is a restriction in praxis it never occurred as a limitation. The following pseudo-code example illustrates the process of annotating:

```python
text = "I'm afraid I can't let you do that, Dave."
text.annotate(0,41,sentence)
text.annotate(36,40,person_name)
```
This annotates the whole text as a sentence and the word “Dave” as a persons name. The different types of annotations must not be added by the same tagger nor in the same order. Tagger may rely on each other. A tagger may assume the text to be tokenized and sentence segmented before it identifies person names. Table 4.6 lists taggers implemented for this work.

<table>
<thead>
<tr>
<th>Tagger</th>
<th>Annotates</th>
<th>Requires</th>
</tr>
</thead>
<tbody>
<tr>
<td>ElivagarTokenizer</td>
<td>lexical units: such as word tokens, white-space, (opening/closing) brackets, punctuations, (opening/closing) quotes</td>
<td>tokens</td>
</tr>
<tr>
<td>EnglishCompounds</td>
<td>splits word token into compounds if appropriate</td>
<td></td>
</tr>
<tr>
<td>SentenceTagger</td>
<td>syntactical units: sentences</td>
<td>words and tokens</td>
</tr>
<tr>
<td>Part-Of-Speech Tagger</td>
<td>wrapper for heuristic taggers, assigns syntactical categories to tokens</td>
<td>punctuations and sentences</td>
</tr>
</tbody>
</table>

Table 4.6: Taggers implemented in the package Elivagar (see supplementary material). The third column names tag types required before application of the tagger.

The tagging mechanism is similar to XSL transformation known from XML processing. However the implementation used greatly simplifies the handling of text objects and has a minimal overhand. The next section describes how such a tree structure was used to semantically markup PubMed documents and textual phrases of the abstracts and titles.

4.2.3 Annotating documents

The words and phrases in structured text need to be linked to meaning in order to enable reasoning. A human reader interprets the words and grammar of a text to form a meaningful mental model. Humans do this by taking all available contextual information they have into account. In order to process text automatically explicit semantic links need to be established to lexical units found in the sentences. Listing 4.1 shows a PubMed citation structured into document attributes, the title and the abstract. The abstract is segmented into sentences. A tagger identified “purines”, “HIV-1” and “RNA” as a lexical tokens. The other tokens were not marked-up in this example. Additionally “Dimerization Initiation Site” was identified to be the long form of the abbreviation “DIS”. Besides words and phrases two author names, a journal name and an id were marked-up.

Non of these information units are linked to meaningful conceptualizations. A biologist is able to interpret the value of the PMID tag as the unique identifier of a literature citation at www.pubmed.org. Readers of the journal named “Biopolymers” will recognize the name of a printed journal with the unique ISSN 0006-3525 published since 1963 in the United States. Researchers with biomedical background associate “purines” with chemical compounds, “HIV-1” as a type of a virus and “RNA” as macromolecule encoding genetic
information. Furthermore colleagues of the authors will recognize their colleagues names and associate the publication to the correct individuals. However PubMed contains entries from a person named “Sponer J” in the Czech Republic, Germany and the US. Without further contextual knowledge it is not possible to figure out whether PubMed refers to the same individual or to different persons.

Listing 4.1: A structured PubMed citation represented in a simple XML-format. Meta-information is stored as children of the document’s root node. The content contains the document title and the abstract. The abstract is segmented into sentences. As illustrating examples phrases are marked up: “purines”, “HIV-1” and “RNA” were marked-up as tokens. “Dimerization Initiation Site” was identified to be the long form of the abbreviation “DIS”.

Goal: enable reasoning over document content

This shows that further reasoning is not possible without linking information in the text to meaningful concepts. The objective was to link phrases in PubMed abstracts and meta-information of the publication to well defined concepts to enable reasoning over the content.

Reasoning is the process of making inferences from a body of information. Given the information that “HIV-1” is a specific type of RNA viruses, which is provided in the background knowledge, and the citation in the above example refers to “HIV-1”, it is reasonable to conclude that the article mentions a virus. Interestingly the whole text of the PubMed entry and the attached keywords do not refer the concept virus at all. Without the background knowledge it is not possible to automatically conclude that the article mentions a virus. Other conclusions could be: the journal with the ISSN 0006-3525 publishes research about RNA viruses, a person in the Czech Republic named K. Réblová does research on the HIV virus. A person named J. Sponer publishes work together with K. Réblová.

Solution: semantic markups, links to background knowledge

Document annotation is the process of establishing associations between semantic concepts and documents or parts of documents. Semantic concepts can include ontology concepts as well as author identities, dates, Wikipedia entries, protein or gene names and geographic information. A document annotator may wrap a complex concept recognition algorithm, as describes in chapter 3, which takes a structured text and returns identified concepts, or simply translates a document attribute such as the publication date into a computable form or map internal identifiers, such as journal names, to external identifiers such as ISSN numbers. In this work document annotators were implemented as taggers. A
document annotator does not insert new nodes to the structured text but adds attributes of the type ontology concept to phrases, sentences, paragraphs or the document’s root.

Chapter 3 specifically discusses the open problem of Concept Recognition. One outcome of this research is that the current approaches are imperfect leading to wrong and missing annotations. Three types of errors can be distinguished: (1) Annotations can be too general missing the specific meaning in the context. This occurs for example when the recognition algorithms misses the correct boundaries of an entity in the text. (2) An annotation can be too specific. This occurs when the recognition does not penalize the missing substantiation in the context. (3) The recognition pipeline can miss concepts at all if the lexical form is not recognized or the disambiguation model fails to classify the concept correctly because of missing training data.

For a domain expert it is easy to detect these errors by reading the abstract text. However for typical queries the number of abstracts returned by PubMed can easily exceed hundreds of citations. Hence it is impractical to look at the annotation manually. The goal was to make correct statements about a large amount of documents returned for a typical PubMed query. The returned documents contain wrong and missing annotations.

The fact that a typical PubMed query returns large sets of citations can be exploited. Well-known biomedical aspects are recurrently stated in several publications. Dooyeweerd [69] suggests an aspectual analysis of text: To undertake textual analysis, go through phrase by phrase or even word by word, on the grounds that each phrase or word is often there by deliberate human choice, whether that choice is conscious or not. (If a phrase gets in from sheer habit of using and repeating that phrase, then it is perhaps meaningless, and can be ignored.)

The idea is to group textual phrases in scientific publications into aspects. Textual phrases are statements about facts, theories or hypothesis referring to known concepts. It is assumed that often repeated statement are more likely to reflect common knowledge and are less likely artifacts of an imperfect text analysis. On the contrary infrequent statements are more likely to reflect less assured ideas and are more likely to result from textmining artifacts. Recently made statements have the potential to reflect new insights.

A Lucene index was created containing 91,046,321 sentences and titles of PubMed citations of the last four years. Along with each sentence/title the document ID, sentence number and concept IDs of each annotation and its ancestors were stored. This resulted in a 64.1GB large file based index.

The index allows queries of the following type: return an iterator over all sentences in citations containing one or more keywords and are annotated with a concept or an descendant. The concept may be an annotation or an ancestor of an annotation. This schema assumes that a document which is relevant for a concept, is also relevant for a more general concept. Suppose a document mentions the disease TUBERCULOSIS. Following the assumption the same document is also relevant for BACTERIAL INFECTIONS. The background knowledge contains the following relations, the symbol “≫” denotes a relation of type is-a and is transitive:

**Mesh ≫ Diseases ≫ Bacterial Infections and Mycoses ≫ Bacterial Infections ≫ Gram-Positive Bacterial Infections ≫ Actinomycetales Infections ≫ Mycobacterium Infections ≫ Tuberculosis**

This implies: TUBERCULOSIS is a MYCOBACTERIUM INFECTION is a . . . is a DISEASE.

All explicit and implicit semantic links are stored in this Lucene index. This makes querying very efficient but makes re-indexing necessary in case the ontology is changed. Another disadvantage of this schema is the large number of documents resulting from the segmentation of each document into many sentences. A complete re-indexing of many documents is a resource intensive task. The following section investigates the usage of databases to query
large sets of annotated documents.

PubMed comprises more than 18 Million citations. Every day up to 5000 documents are added or revised. The biomedical knowledge stored in MeSH and the Gene Ontology growth. The Gene Ontology project updates its databases on a weekly basis. The re-indexing of all PubMed citations takes several days on a cluster of ten machines. The goal was to reason with changing background knowledge of ten-thousands of concepts over a growing number of millions of documents.

A relation database was designed to allow efficient reasoning over millions of stand-off annotations. Documents were queried separately from the annotations. Annotations were made per abstract but not for sentence. The main database table maps document ids to concept ids (an additional column was added with the foreign key of annotation ids). An index was created over the document id column. Another table was designed which holds for each annotation information about the confidence and the text range of the annotation.

The database can be queried as follows: suppose a biologist is interested in document mentioning diseases. The background knowledge, also stored in the database, is queried for descendant concepts of Diseases, the result is a temporary table comprising 6527 concept ids. The annotations table is joined with the temporary table representing the diseases branch of the ontology. The result is a table of document ids of citations mentioning the concept Diseases or a descendant. The result also contains the references to the annotation details which can be used to highlight the textual evidences generated by the Concept Recognition algorithm.

Currently the document table contains 17,870,689 entries, the annotation table contains 412 million entries for GO and MeSH comprising 46,432 ontology concepts. The database was implemented using the MySQL Server 5.0 on a 16GB machine. The index size is >14GB.

The main purpose of this database is the efficient retrieval of all semantic markups in a set of document ids. This does not include information about concepts which are related to the involved concepts. The following section describes the necessary processing step to create a graph connecting all these concepts.

4.2.4 Ontology Induction

When authors use concepts to express facts, theories or hypotheses they assume that the reader has some knowledge about the concept, either from the previous sections of the publications or from common knowledge. In an ontology-based literature search system this implicit knowledge needs to be reconstructed from the ontology in order to be used for automatic textmining. This process is named Ontology Induction.

The objective was to provide users of an ontology-based search engine with explicit links to background knowledge stored in ontologies. The background knowledge comprises taxonomical relations, synonymous labels and human readable definitions. If a user explores citations mentioning Tuberculosis he/she should be provided with the information that tuberculosis is a Mycobacterial Infection and that it is a Disease caused by a specific Bacteria. Additional useful information is that the Tuberculin Skin Test is used for diagnosis and Isoniazid is the current treatment of choice. Mass Vaccinations with BCG against tuberculosis has been a major health intervention.

The idea is to compute a graph which connects all concepts found in the annotations of the retrieved documents to the most general concept defined, the root of the ontology. All concepts of the background knowledge which are part of a path from an annotated concept all the way up to the root are included in the induced ontology tree. The tree contains multiple instances of concepts having more than one parent concept. The set of concepts $\Theta$
4.2. METHODS

represented by the induced ontology can be defined as follows:

\[
\Theta = \bigcup_{i=1}^{d} \bigcup_{t=1}^{a} \text{ancestors}(O, \text{concept}(i, t))
\]

where \( d \) is the number of documents of a result set and \( a \) is the number of annotations in a document. The function \( \text{concept} \) returns the ontology concept associated with a document annotation and \( \text{ancestors} \) returns the set of concepts transitively related to the given concept in the ontology \( O \).

The complexity class of the function to compute \( \Theta \) is quadratic, \( O(d \cdot t) \), while \( d \) is typically very large and \( t \) is typically small, assuming the complexity of \( \text{term} \) and \( \text{ancestors} \) is constant. If \( d \) becomes large the computation becomes inefficient. For a typical query to a document collection such as PubMed \( d \) is between 100 and 1,000,000. Figure 4.2 shows the distribution of result set sizes for biomedical queries send to GoPubMed in the last two years.

![Figure 4.2: Size of the result set returned for PubMed queries send to GoPubMed during the last two years.](image)

The number of documents PubMed returned for all queries to GoPubMed was measured. About 38% of the queries return less than 100 document ids in the result set. Another 37% returned between 100 and 1,000 document ids, 12% return 1000 to 10,000 documents and the rest, 13%, return more than 10,000 documents. This shows that 25% of the queries require to process more than 1000 documents per user query. The distribution follows roughly a power law, many queries result in short results while few queries result in many results.

The difficulty of ontology-based search is that after the retrieval process the annotations of each document needs to be fetched to induce the ontology. This task can be resource intensive. In the worst case scenario all of the documents are new, not yet annotated. This means the Concept Recognition pipeline needs to be executed for each document which takes several milliseconds per abstract on the current hardware and implementation. After this,
Ontology Induction objectives. Ontology Induction has three objectives: (1) Ontology Induction should be independent of the size of the document result set. (2) Ontology Induction should be complete, the induced ontology should contain all concepts associated with at least one document in the result set. (3) Ontology Induction should be correct. All documents annotated with an induced concept or a descendant should be linked to the node in the induced ontology.

One approach to reduce the required resources per query is to sample a representative subset of documents from the result set. This makes the Ontology Induction independent of the result set size, but may fail to accomplish the second and third objective. The question was: How large must a document sample be to nearly accomplish the second objective, the completeness of the induced ontology.

An experiment was conducted in which for fixed query result sets the ontology was induced with different document sample sizes. It was assumed that with a certain sample size the induced ontology does not change much. Seven queries with large result set sizes were selected.

Figure 4.3 shows the sample size used to induced the ontology for typical queries. The induced ontology does not change much when sampling with more than 1000 documents.
4.2. METHODS

This suggests a heuristic sampling the induced ontology with 1000 to 2000 document samples. The question is how to select the samples.

It can be assumed that recent documents are of higher interest to users. Therefore a heuristic sampling the most recent 1000 documents is used. This makes the Ontology Induction independent of the result sets size while accomplishing the second objective, the induced ontology is complete (with minor restrictions). It is clear that the third objective can not be accomplished with the sampling approach. Documents at ranks >1000 are not linked to the induced ontology. A solution to this is to send a new query to the system when the user selects a concept. The new query adds to the original keywords the condition that the selected concept or a descendant is mentioned in the documents.

The induced ontology can be used to explore the initial search result in a hierarchical way. The most general concepts being at the upper levels of a tree hierarchy. Specific concepts are found in deeper levels. Using MeSH and the Gene Ontology as background knowledge can result in graph with up to 13 levels. It depends on the user at which level he/she seeks useful abstraction. However with each level an additional user interaction, e.g. a mouse click, becomes necessary. The following section discusses ideas how to further condense the information in the tree in order to be more concise.

4.2.5 Interpreting Induction Results

When using MeSH and GO on PubMed query the results of a typical induction result contains in the order of hundreds of concepts. GO and MeSH together list 13 root categories comprising 67,561 concepts. Exploring the tree the user can choose between 5-20 subconcepts of the root categories. While exploring the branches can be interesting even for a domain expert it can require several user interactions to arrive at nodes representing an interesting subset of documents. For example when searching for "rab5" the biological process this protein is involved in, ENDOCYTOSIS, is found at level 5. Figure 4.4 shows the induced ontology for the result set provided by PubMed.

The organization of the search results with the induced ontology is a rich source of information. The total number of documents processed for the example query "rab5" was 772. The fact that 420 documents (54%) mention the concept ENDOCYTOSIS clearly indicates that this biological process is of particular interest to the researchers. From the ontology’s structure it becomes clear that ENDOCYTOSIS is a biological process involved in the organization membranes within the cell. Also another biological process sticks out: ESTABLISHMENT OF LOCALIZATION. Research on this topic has focused on the problem how vesicles move in the cell. Furthermore, 241 (31%) of the documents mention a disease. Recent studies suggested that endocytic pathway abnormalities link to the Alzheimer disease.

Exploring the induced ontology reveals insights into the search results of a PubMed query. However informative aspects might be buried in the depths of the background knowledge. The goal is to condense the information provided with the tree so that informative aspects become visible with fewer user interactions. An informative aspect is a set of induced concepts linked to their associated documents representing a salient topic in a document set. The following sections discuss measures to rank informative subsets of induced concepts.

Informative paths

A particular path in the induced tree can represent an informative aspect. In the previous example the path leading from the root node to the concept ENDOCYTOSIS characterizes an important aspect of the recent research related to the protein rab5. The main indicator is that the majority of documents mention the concept. However there are other concepts
Induced Ontology for PubMed query ’rab5’

- Anatomy [735]
- Biological Sciences [751]
- Cellular component [702]
- Chemicals and Drugs [769]
- Biological process [726]
  - Biological adhesion [14]
  - Biological regulation [415]
  - Cellular process [698]
    - Cellular component organization and biogenesis [560]
      - Membrane organization and biogenesis [459]
      - Membrane invagination [420]
      - Endocytosis [420]
    - Developmental process [231]
    - Establishment of localization [544]
    - Growth [56]
    - Immune system process [33]
    - Localization [584]
    - Locomotion [10]
    - Maintenance of localization [16]
    - Metabolic process [326]
    - Multi-organism process [81]
    - Multicellular organismal process [206]
    - Pigmentation [2]
    - Reproduction [50]
    - Reproductive process [44]
    - Response to stimulus [68]
    - Rhythmic process [2]
    - Viral reproduction [38]
- Diseases [241]
- Health Care [311]
- Named Groups [56]
- Natural Sciences [639]
- Organisms [670]
- Techniques and Equipment [549]
- Molecular function [478]

Figure 4.4: This figure shows the induced ontology for the PubMed result to the query “rab5”. The top level MeSH and GO categories are alphabetically ordered. The GO branch Cellular process is expanded to the concept endocytosis, the topic process most intensively for the protein. The induced ontology shows 16 other biological processes related to the protein rab5. The numbers denote the amount of the linked documents.
representing even more documents. One approach of identifying an informative path is to
traverse the tree from the root node along the most populated node at each level. This is
named the Richest Path.

**Richest Path.** The Richest Path $\Phi$ of a node $x$ is defined as:

$$\Phi(x) = m \cup \Phi(m)$$

where $m = \text{maxsize}(\Theta, \text{children}(x))$

where $\text{maxsize}$ returns the concept with the most documents associated for a set of induced
concepts $\Theta$ and $\text{children}$ returns the concepts directly related to the given concept.

The richest path typically represents the most obvious aspect of a document set. Querying
PubMed for "ribosome" the richest path $\Phi$ for the root of the Gene Ontology is shown in
figure 4.5

```
GENE ONTOLOGY [829]
  ➩ CELLULAR COMPONENT [829]
  ➩ CELL [818]
  ∈ CELL PART [818]
  ➩ INTRACELLULAR [801]
  ∈ INTRACELLULAR PART [795]
  ➩ INTRACELLULAR ORGANELLE [775]
  ➩ INTRACELLULAR NON-MEMBRANE-BOUND ORGANELLE [664]
  ➩ RIBOSOME [626]
```

Figure 4.5: The richest path for the query "ribosome". The symbol "≫" denotes an is-
a relation and "∈" denotes a part-a relation. A ribosome can be characterized as a non-
membrane-bound intracellular organelle which is an intracellular part of the cell.

The result shown in figure 4.5 was computed using 1000 of the most recent documents
mentioning ribosome. Exploring the richest path from top to bottom the user learns without
having to read the documents in the result set: 83% of the documents can be classified based
on the cellular component branch of the Gene Ontology. The user can conclude from this
that a concept named ribosome likely is related to a cellular component or it is one itself.
Exploring the richest path further the majority of the associated documents relate to the
intracellular organelles, 775 of the documents. Eventually the user learns that a ribosome is
an intracellular organelle itself by following the richest path. At each point in the path the
definitions of the concepts help understanding the sub-domain. For example the definition
given in the Gene Ontology for RIBOSOME is: An intracellular organelle, about 200 angstrom
in diameter, consisting of RNA and protein...

Interestingly the richest path must not necessarily contain the keyword itself. This allows
to infer a formal conceptualization for a query term which is not part of the ontologies
terminology. For example querying for "breathing" the user learns quickly from the richest
path of the Gene Ontology: BIOLOGICAL PROCESS ➩ METABOLIC PROCESS ➩ CELLULAR
METABOLIC PROCESS ➩ GENERATION OF PRECURSOR METABOLITES AND ENERGY…that
breathing relates to a (metabolic) biological process connected to the generation of energy
in the body. Note that breathing is not a concept in the Gene Ontology nor is it a synonym
of a concept known by the background knowledge.

It is important to note that the richest path can be computed beginning from any ref-
ence node. For example a pharmacist might want to explore the richest path beginning
from the MeSH branch CHEMICALS AND DRUGS. In a graphical interface it is desirable to
minimize the number of clicks to reach an informative node. While exploring deep paths provides a lot of contextual information such as the sister concepts at each level, it can be impractical having to navigate all the way down to a concept of the desired specificity.

Possible future work is to investigate other types of informative paths and to analyse their population with documents. The Richest Path is the path with the most documents associated. However there might be other indicators for informative paths. For example the number of documents associated with each node in any path might reveal the point in the path when document by many sister concepts, possibly indicating an interesting level of detail. The fan out at each node in a path might indicate broader or narrower research topics. The length of paths might indicate the level of abstraction of a topic. Finally, a path may be informative because it contains one or more single informative concepts. The following section introduces measures for informativeness of a single concept.

Informative concepts

Having to browse a large hierarchy requires many interactions. The user needs to click through the branches of the ontology. With a large background knowledge this can be time consuming and the expanded tree quickly becomes multitudinous.

The goal is to select a small set of concepts characterizing the result set without the need for additional user interaction. Such concepts represent important aspects of the documents content. Such aspects may origin from different levels of granularity and may serve as shortcuts into the background knowledge.

Solution:
tf-idf weight for ranking concepts

Top Categories In information retrieval and text mining the tf-idf weight is a widely used statistical measure to evaluate how important a word is to a document in a corpus. The measure can be used to weight the importance of a concept to a search result. The importance of a concept increases proportionally to the number of times the concept is mentioned in the result set but is offset by the frequency of the mentions in the whole corpus.

The classical tf-idf weight is computed by the product of the term frequency and the inverse document frequency. The term frequency is the number of times a given term appears in document. This count is normalized to prevent a bias toward longer documents. The inverse document frequency measures the global importance of the term. It is defined by the logarithm of the quotient of the size of the corpus and the number of documents containing the given term.

In ontology-based search the two measure are defined as follows: The term frequency (in the following named concept frequency) is the number of times a given concept is mentioned in the document result set. There are actually two interpretations of this definition. The first, named “concept frequency”, counts the explicit mentions in the documents, the second, named “induced concept frequency”, takes the structure of the ontology into account by summing up the explicit mentions and the mentions of the descendant concepts. The concept frequency \( cf \) of a concept \( c \) in a document set \( D \) is defined as:

\[
    cf(D, c) = \frac{\sigma(D, c)}{|D|} \quad \text{(concept frequency)}
\]

where \( \sigma(D, c) \) defines the number of documents in \( D \) mentioning \( c \) explicitly. The induced concept frequency \( icf \) of a concept \( c \) in a document set \( D \) over an ontology \( \Omega \) is defined as:

\[
    icf(\Omega, D, c) = \frac{\sigma(D, \omega(\Omega, c))}{|D|} \quad \text{(induced concept frequency)}
\]
4.2. METHODS

where \( \omega(\Omega, c) \) defines the set of concepts containing the concept \( c \) and its descendants in the ontology \( \Omega \).

The inverse document frequency (in the following named inverse concept frequency) can also be defined in two different ways. The first variant counts the explicit mentions of a concept in the document corpus. The second variant takes the structure of the ontology into account by summing up the explicit mentions and the mentions of the descendant concepts. The inverse concept frequency \( idf \) of a concept \( c \) in a document corpus \( C \) is defined as:

\[
idf(C, c) = \log \frac{|C|}{\sigma(C, c)} \quad (inverse \ corpus \ frequency)
\]

where \( \sigma(C, c) \) defines the number of documents in \( C \) mentioning \( c \) explicitly. The induced inverse document frequency \( iidf \) of a concept \( c \) in a document corpus \( C \) over an ontology \( \Omega \) is defined as:

\[
 iidf(\Omega, C, c) = \log \frac{|C|}{\sigma(C, \omega(\Omega, c))} \quad (induced \ inverse \ corpus \ frequency)
\]

From the above definitions of concept frequency and inverse corpus frequency four different definitions for a weighting measure can be derived:

\[
 cf - idf = cf \cdot idf \quad (local \ count - global \ count)
\]

\[
 icf - idf = icf \cdot idf \quad (local \ induced - global \ count)
\]

\[
 cf - iidf = cf \cdot iidf \quad (local \ count - global \ induced)
\]

\[
 icf - iidf = icf \cdot iidf \quad (local \ induced - global \ induced)
\]

The question was: Which measure provides the best way to rank the importance of a concept to a search result? To assess the quality of the ranking measures domain experts were ask to provide a typical PubMed query for their research topic. The experts came up with 20 queries shown in table 4.7.

The experts ranked a selection of concepts according to their relevance to the topic. The selection was the union of the 40 top ranked concepts by the four different weighting schemes. The list was sorted alphabetically and presented to the expert which assigned scores to each concept. A score of 10 denoted the concept was very important for the topic a score of 0 denoted the concept was not important at all for the topic. Table 4.9 shows the decisions of an expert for the PubMed query "pancreas cancer prognosis".

The provided rankings were used as the gold-rankings to compare the weighting functions. The 40 best ranked concepts by each function were taken and compared to the gold-ranking. At each rank \( r \) the quality of the concept ranking \( q \) was evaluated as follows:

\[
 q(r) = \sum_{i=1}^{r} score(c_i) \cdot \frac{r - i + 1}{r}
\]

where \( score \) is the score for the concept \( c \) provided in the gold-ranking. The results of the evaluation for the query "pancreas cancer prognosis" are shown in figure 4.6.

The results suggest that the locally induced global count (icf-idf) performs worst. This is an expected outcome because local frequencies are added up for all descendants and are therefore overweighted in the measure. The influence of global frequencies is minimal. The three other measures do not have such a disproportion. The local count - global count (cf-idf) and the local induced - global induced (icf-iidf) take the sum of all descendant concept mention for both factor and perform equally best.

| Problem: each single ranking is represents the individual experts bias |
### Table 4.7: This table shows 20 PubMed queries provided by domain experts. The experts ranked a selection of concepts according to their relevance to the topic. The selection was made using the union of the 40 top ranked concepts by the four different weighting schemes.

<table>
<thead>
<tr>
<th>PubMed Query</th>
<th>PubMed results</th>
<th>Evaluated concepts</th>
</tr>
</thead>
<tbody>
<tr>
<td>alzheimer</td>
<td>48,022</td>
<td>77</td>
</tr>
<tr>
<td>aspirin</td>
<td>42,052</td>
<td>80</td>
</tr>
<tr>
<td>bioinformatics</td>
<td>33,337</td>
<td>69</td>
</tr>
<tr>
<td>brand m[au] dresden[ad]</td>
<td>34</td>
<td>92</td>
</tr>
<tr>
<td>breast cancer</td>
<td>190,676</td>
<td>80</td>
</tr>
<tr>
<td>chemokine receptor</td>
<td>16,533</td>
<td>97</td>
</tr>
<tr>
<td>endosome</td>
<td>9,085</td>
<td>90</td>
</tr>
<tr>
<td>HspA pylori</td>
<td>32</td>
<td>81</td>
</tr>
<tr>
<td>IL16 and PDZ</td>
<td>14</td>
<td>93</td>
</tr>
<tr>
<td>insulin resistance</td>
<td>42,500</td>
<td>75</td>
</tr>
<tr>
<td>Lipoprotein</td>
<td>125,283</td>
<td>79</td>
</tr>
<tr>
<td>liver transplantation</td>
<td>54,891</td>
<td>74</td>
</tr>
<tr>
<td>nickel binding proteins pylori</td>
<td>53</td>
<td>81</td>
</tr>
<tr>
<td>oligomerization AND modeling</td>
<td>17</td>
<td>75</td>
</tr>
<tr>
<td>pancreas cancer prognosis</td>
<td>6,148</td>
<td>75</td>
</tr>
<tr>
<td>polyproline II</td>
<td>427</td>
<td>66</td>
</tr>
<tr>
<td>rab5</td>
<td>772</td>
<td>95</td>
</tr>
<tr>
<td>simons k[au] dresden[ad]</td>
<td>46</td>
<td>92</td>
</tr>
<tr>
<td>tuberculosis</td>
<td>167,926</td>
<td>75</td>
</tr>
<tr>
<td>zerial m[au]</td>
<td>121</td>
<td>89</td>
</tr>
</tbody>
</table>

Figure 4.6: This figure shows the quality of the concept ranking for the topic "pancreas cancer prognosis" at different ranks for the four weighting functions: cf-idf, icf-idf, cf-iidf and icf-iidf. Local Induced - Global Count (icf-idf) performs worst compared to all other rankings. Local Induced - Global Induced slightly dominates at high ranks.
Table 4.9: The ranking of concepts provided an expert for the PubMed query "pancreas cancer prognosis".

<table>
<thead>
<tr>
<th>Carcinoma, Pancreatic Ductal</th>
<th>Surgery</th>
<th>10</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adenocarcinoma</td>
<td>Prognosis</td>
<td>9</td>
<td>4</td>
</tr>
<tr>
<td>Neoplasm Staging</td>
<td>Tissue Extracts</td>
<td>9</td>
<td>4</td>
</tr>
<tr>
<td>Carcinoma</td>
<td>Adult</td>
<td>8</td>
<td>3</td>
</tr>
<tr>
<td>Digestive System Neoplasms</td>
<td>Aged</td>
<td>8</td>
<td>3</td>
</tr>
<tr>
<td>Neoplasm Metastasis</td>
<td>Middle Aged</td>
<td>8</td>
<td>3</td>
</tr>
<tr>
<td>Neoplasms, Glandular and Epithelial</td>
<td>Palliative Care</td>
<td>8</td>
<td>3</td>
</tr>
<tr>
<td>Pancreatectomy</td>
<td>Pathological Conditions, Signs and Symptoms</td>
<td>8</td>
<td>3</td>
</tr>
</tbody>
</table>

Pancreatic Neoplasms | Aged, 80 and over | 8 | 2 |
Pancreatocoduodenectomy | Enzymes and Coenzymes | 8 | 2 |
Digestive System Diseases | Hominidae | 7 | 2 |
Digestive System Surgical Procedures | Patients | 7 | 2 |
Lymph Nodes | Quality of Health Care | 7 | 2 |
Neoplasms | Specialties, Medical | 7 | 2 |
Neoplasms by Histologic Type | Biological Sciences | 7 | 1 |
Neoplasms by Site | Biological process | 7 | 1 |
Neoplastic Processes | Chemicals and Drugs | 7 | 1 |
Pancreatic Diseases | Chordata | 7 | 1 |
Pancreatitits | Complex Mixtures | 7 | 1 |
Survival Analysis | Diagnosis | 7 | 1 |
Survival Rate | Environment and Public Health | 7 | 1 |
Digestive System | Epidemiologic Measurements | 6 | 1 |
Endocrine Gland Neoplasms | Epidemiologic Methods | 6 | 1 |
Endocrine System Diseases | Epidemiologic Study Characteristics | 6 | 1 |
Lipase | Evaluation Studies | 6 | 1 |
Lymph | Health Occupations | 6 | 1 |
Mortality | Health Services Administration | 6 | 1 |
Pancreas | Investigative Techniques | 6 | 1 |
Pancreatic Extracts | Mammals | 6 | 1 |
Pancrelipase | Named Groups | 6 | 1 |
Treatment Outcome | Natural Sciences | 6 | 1 |
Carboxylic Ester Hydrolases | Outcome and Process Assessment (Health Care) | 5 | 1 |

Catarrhini | Outcome Assessment (Health Care) | 5 | 1 |
Drug Therapy | Persons | 5 | 1 |
Esterases | Primates | 5 | 1 |
Hydrolases | Techniques and Equipment | 5 | 1 |
Recurrence | Vertebrates | 5 | 1 |
Retrospective Studies | | 5 | |
CHAPTER 4. ONTOLOGY-BASED LITERATURE SEARCH

Figure 4.7: This plot shows the average quality for four ranking functions on 20 queries. The function \( \text{icf} - \text{idf} \) performs worst, \( \text{icf} - \text{iidf} \) has slightly lower performance than the other two weighting functions. The other two, not inducing locally, rank the importance of concepts for a topic equally good.

Interestingly the local count - global induced measure (cf-iidf) performs almost equally good as the best two measures. It is not clear whether this is due to the low number of rankings compared or whether this is a consistent result. In some cases the experts evaluated the rankings of this measure better than others. The computational cost for the local count - global induced measure (cf-iidf) are higher than the normal local count - global count (cf-idf). It is clear that the scorings made by an individual expert were highly subjective. In order to assess the performance of the ranking function more objectively the curves of all plots were averaged and plotted in the same way.

Figure 4.7 clearly shows a trend for the performance of the ranking functions: the function \( \text{icf} - \text{idf} \) (local induced global count) performs worst, \( \text{icf} - \text{iidf} \) (local induced global induced) has slightly lower performance than the other two weighting functions. The other two rank the importance of concepts for a topic equally well when compared with the gold-ranking.

From these results it can be concluded that taking the normal frequency counts of the concepts mentioned in the document result set result in a better weighting measure than taking the induced frequencies. For the induced concept frequencies computed for the whole corpus it does not changed the quality whether to induce the frequencies or not. Table 4.10 shows similar rankings for the two best weighting functions. The low quality for the \( \text{icf} - \text{idf} \) (local induced - global count) ranking can be explained by the fact that local frequencies are over weighted. The global statistics are essentially ignored because the frequencies of concepts seen in the result set add up overproportionally. The slight inferiority of the \( \text{icf} - \text{iidf} \) ranking might not be significant considering the few gold-rankings used.

**Problem:** How to rank the linked documents?

The top ranked concepts give a good insight into the content of the documents retrieved for the query "pancreas cancer prognosis". Section 4.2.7 is discussing how the induced ontology, informative paths and concepts enable ontology-based question answering.
4.2. METHODS

<table>
<thead>
<tr>
<th>local count - global count (cf-idf)</th>
<th>local count - global induced (cf-idi)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pancreatic Neoplasms</td>
<td>Pancreatic Neoplasms</td>
</tr>
<tr>
<td>Pancreatitis</td>
<td>Pancreatitis</td>
</tr>
<tr>
<td>Prognosis</td>
<td>Pancrelipase</td>
</tr>
<tr>
<td>Pancrelipase</td>
<td>Prognosis</td>
</tr>
<tr>
<td>Pancreas</td>
<td>Pancreas</td>
</tr>
<tr>
<td>Neoplasms</td>
<td>Patients</td>
</tr>
<tr>
<td>Adenocarcinoma</td>
<td>Pancreatectomy</td>
</tr>
<tr>
<td>Carcinoma</td>
<td>Adenocarcinoma</td>
</tr>
<tr>
<td>Pancreatectomy</td>
<td>Carcinoma</td>
</tr>
<tr>
<td>Neoplasm Metastasis</td>
<td>Treatment Outcome</td>
</tr>
<tr>
<td>Treatment Outcome</td>
<td>Neoplasm Metastasis</td>
</tr>
<tr>
<td>Pancreatectomy</td>
<td>Neoplasms</td>
</tr>
<tr>
<td>Survival Rate</td>
<td>Survival Rate</td>
</tr>
<tr>
<td>Patients</td>
<td>Pancreatectomy Duodenectomy</td>
</tr>
<tr>
<td>Aged</td>
<td>Aged</td>
</tr>
<tr>
<td>Middle Aged</td>
<td>Middle Aged</td>
</tr>
<tr>
<td>Diagnosis</td>
<td>Surgery</td>
</tr>
<tr>
<td>Surgery</td>
<td>Neoplasm Staging</td>
</tr>
<tr>
<td>Survival Analysis</td>
<td>Survival Analysis</td>
</tr>
<tr>
<td>Neoplasm Staging</td>
<td>Lymph</td>
</tr>
<tr>
<td>Drug Therapy</td>
<td>Lymph Nodes</td>
</tr>
<tr>
<td>Lymph</td>
<td>Aged, 80 and over</td>
</tr>
<tr>
<td>Lymph Nodes</td>
<td>Retrospective Studies</td>
</tr>
<tr>
<td>Aged, 80 and over</td>
<td>Recurrence</td>
</tr>
<tr>
<td>Carcinoma, Pancreatic Ductal</td>
<td>Evaluation Studies</td>
</tr>
<tr>
<td>Retrospective Studies</td>
<td>Drug Therapy</td>
</tr>
<tr>
<td>Recurrence</td>
<td>Carcinoma, Pancreatic Ductal</td>
</tr>
<tr>
<td>Evaluation Studies</td>
<td>Adult</td>
</tr>
<tr>
<td>Palliative Care</td>
<td>Palliative Care</td>
</tr>
</tbody>
</table>

Table 4.10: Both weighting functions of concepts for the “pancreas cancer prognosis” provide similar rankings.
This section focused only on the concepts and how to condense the induced ontology tree to make it even more useful. The documents linked to the induced concepts are discussed in the following section. The question was, how should the documents being ranked.

### 4.2.6 Aspect ranking

The result of the Ontology Induction for the query "rab5" was visualized with figure 4.4 in section 4.2.5. The protein rab5 is believed to regulate endocytosis. The path containing the concept endocytosis holds the most document associations in the branch of biological processes. Also both top weighting functions rank the concepts among the very top categories, rank 3 (cf-idf) and rank 6 (cf-idiif).

Each node in the induced ontology represents an aspect of the query result, a concept linked to all documents mentioning it or a descendant. Figure 4.4 shows that 420 out of 772 documents mention endocytosis for the query result on the topic "rab5". An aspect can be a well known idea or a new information to the user. In any case he/she might want to confirm this information. This can be done by reading the abstracts relating the query keywords to the selected concept. The goal is to rank the abstracts in such a way that the most relevant for the selected aspect are ranked high.

An aspect is the sum of facts, theories and hypothesis which relate a concept to a topic. A document is relevant for an aspect if its content explicitly states a relation between the topic and a concept.

An optimal method would establish a semantic link between concept and the topic. This is the task full sentence parsers, introduced in section 2.7. The methods available can produce good results. However the training costs and computational resources required are still high. Considering that for a typical aspect hundreds of documents and thousands of sentences need to be parsed, full sentence parsing does not scale for this task.

**Solution:** Proximity ranking A heuristic approach of aspect ranking is based on the distance of textual mentions. Given a document which was retrieved for one or more keywords the document is more relevant for the aspect if it mentions the concept or one of its descendants in close proximity, e.g. the same sentence. Formally the proximity rank of a document $D$ is defined as:

$$\text{rank}(D) = \max\left(\frac{1}{1 + |\text{pos}(D,c) - \text{pos}(D,k)|}\right); \ c \in C; \ k \in K$$

where $C$ is the set of concepts and its descendants, $K$ is the set of keywords relevant for the query and $\text{pos}$ is defined as the character position of a sub-string denoting a keyword or a concept mention. An approximation of the above formula is:

$$\text{rank}'(D) = \begin{cases} 1 & \text{contains(sentence}(D,c),k) \\ 0 & \text{otherwise} \end{cases}$$

Well known ideas are typically recurrently stated in the literature. Therefore, such ideas are supported by many quotations. The next section describes a prototype for a full question answering tool which makes use of this observation and the methods describes in the section previous.

### 4.2.7 Question answering

Question answering is an information retrieval task. Given a collection of documents, e.g. all PubMed citations, the system should be able to retrieve answers to questions. Typically
the questions are formulated by human users in natural language and the system needs to analyse the question phrase first. The result can be a text stating what the system believes is the answer to the question, a citation answer, or a ranked list of concepts, a glossary answer. Links to the textual evidences are helpful to confirm the answers.

The methods described above can be used to search the literature for answers by using a background knowledge stored in an ontology. The workflow can be summarized as follows:

1. Document retrieval using expanded keyword search
2. Structuring of retrieved texts and annotation with concepts
3. Computing the induced ontology
4. Selecting informative aspects
5. Print representative citations for informative aspects

In the section 4.2.5 several definitions of informativeness were given. It is difficult to select a single best answer without further information on the users intentions. In contrast to the classical search engines which rank documents based on predefined heuristics such as search history, geographic location or browsing behavior, the result of the ontology-based search is the outline of the entire content. This completeness is a strength of ontology-based literature search, since it allows a serendipity search.

For a full sentence question answering system more assumptions have to be made. The task is to interpret a full user question. This includes the type of the question, the entities and the concepts which relate to each other. The task of question answering was repeatedly subject to the textmining competitions [119, 120, 117, 118, 108].

A prototype was developed to answer biomedical questions fully automated using the methods described in this chapter. Figure 4.8 displays a screenshot of the web-interface: (1) The user enters a full sentence question. (2) The type of the question is determined. It this prototype only questions for instances of a known concept are allowed. e.g. “Which diseases are related by the HIV virus?” (3) The system eliminates stopword. In this example the words “are”, “by”, “the” are eliminated. (4) The system tests noun phrases detected in the sentence whether they represent known concepts of the background knowledge. In this case two noun phrases can be identified: “diseases” and “aids”. The first one is a known concept in the MeSH terminology and has 6527 subconcepts. (5) The remaining noun phrases are used as keywords in a search on the Lucene full text index described in section 4.2.1. (6) Sentences retrieved during the retrieval process are filtered for mentions of diseases. In this case the sentences must be annotated with the MeSH term DISEASES as an ancestor of a textual annotation. (7) The ten most frequently mentioned descendant concepts of the concept at question are presented as the answer to the question. Each answer is supported by a list of sentences mentioning both the query keywords and a disease. The answers to the question: “Which diseases are associated with aids?” are shown in figure 4.8.

This prototype demonstrates that this method is sound and produces precise results. The crucial points are: parsing the query for the right answer type, concept recognition on one sentence without context, and definition of informativeness.

Future work:
Allow more question types

Bibliometric information such as most prominent authors for a topic or important journals covering a research field as well as temporal information such as the amount of research activity in a field can be answered visually in a standardized way. The following section describes methods to analyse the bibliometric information of a document collection.
Which diseases are associated with AIDS?

Most-relevant biomedical concepts:

- Acquired Immunodeficiency Syndrome explain...
  ...AIDS. PMID: 15937234 explain more.

- HIV Infections and Acquired Immunodeficiency Syndrome explain...
  ...HIV infection and AIDS. PMID: 17633357 explain more.

- Tuberculosis and Acquired Immunodeficiency Syndrome explain...
  ...AIDS and tuberculosis - a lethal combination. PMID: 15523763 explain more.

- Lymphoma and Acquired Immunodeficiency Syndrome explain...
  ...NK/T-cell lymphoma in AIDS. PMID: 17246251 explain more.

- Syndrome and Acquired Immunodeficiency Syndrome explain...
  ...Fisher-Casali-Bitré overlap syndrome in advanced AIDS. PMID: 17391707 explain more.

- Cryptococcosis and Acquired Immunodeficiency Syndrome explain...
  ...Smoking and cryptococcosis in AIDS patients. PMID: 16367823 explain more.

- Hepatitis and Acquired Immunodeficiency Syndrome explain...
  ...Coxal flexural joint syndrome. PMID: 15351235 explain more.

- Hemophilia A and Acquired Immunodeficiency Syndrome explain...
  ...The tragic story of AIDS in the hemophilia population, 1982-1984. PMID: 17313907 explain more.

- Histoplasmosis and Acquired Immunodeficiency Syndrome explain...
  ...Colonial histoplasmosis in AIDS. PMID: 16169784 explain more.

- Leishmaniasis and Acquired Immunodeficiency Syndrome explain...
  ...Intestinal leishmaniasis in a patient with AIDS. PMID: 17125123 explain more.

Figure 4.8: The prototype can answer “Which” questions by decomposing the natural language questions and using the background knowledge of MeSH and GO to present relevant answers in the form of aspects. Each list item is a concept defined in the background knowledge and groups textual passages from PubMed abstracts which mention the keywords and the concept together.
4.3. EVALUATION OF ONTOLOGY-BASED LITERATURE SEARCH WITH GOPUBMED

4.2.8 Bibliometric analysis

All outcome of research is captured in the scientific literature, which grows exponentially. At a closer glance it turns out that some research areas shrink, while others take off. Bibliometric analyses aim to shed light on such developments and to identify emerging trends. Bibliometric analysis helps to understand the social process of publishing by investigating co-author and co-citation networks. These analyses are useful to take a birds-eye view onto research.

**Prominent authors.** An author can be seen as a prominent contributor to a research area if he/she published significantly in the field. It is widely agreed that quantity is not a measure for quality. Citation indexes, such as the SCImago Journal Rank and the Thomson Impact Factor, aim at ranking research publications for their impact on a research field. Taking such a weighting schema for the impact of an authors work one can define an indicator of the rank of an author \( a \) for a concept \( c \) represented in a document result set \( D \):

\[
\text{rank}(a, D) = \frac{\sum \text{index}(d)}{|D|}; \; d \in D; \; r(a, d) = 1
\]

where \( \text{index} \) is an indicator function for the research impact of a document and \( r \) is 1 for a document co-authored by \( a \) otherwise 0.

Alternatively the authors rank can be defined for a concept of an ontology as:

\[
\text{rank}'(a, c) = \frac{\sum \text{index}(d)}{|D_\Omega(c)|}; \; d \in D_\Omega(c); \; r(a, d) = 1
\]

where \( D_\Omega(c) \) is defined as the documents associated with concept \( c \) in the induced ontology over all documents and \( r \) is 1 if document \( d \) is co-authored by \( a \) otherwise 0.

Similar rankings were defined for journals, cities and counties. The ordered list of authors, journals, cities and countries can be easily visualized in tabular form. The computation of \( \text{rank}' \) is resource intensive but can be precomputed. On a cluster of 10 machines the computation took 2 days.

4.3 Evaluation of Ontology-based Literature Search with GoPubMed

Ontology-based Literature Search is a new approach to explore the scientific literature. Currently there are no benchmarks available to evaluate the quality of ontology-based search engines.

When people search, they have questions in mind. GoPubMed is the first semantic search engine using background knowledge to help answering biomedical questions. GoPubMed (1) retrieves PubMed abstracts for your search query, (2) detects ontology terms from the Gene Ontology and Medical Subject Headings in the abstracts and (3) allows the user to browse the search results by exploring the ontologies and displaying only papers mentioning selected terms, their synonyms or descendants.

The objective was to assess the quality of the GoPubMed search engine by evaluating how the system supports question answering. The goal was to answer biomedical questions with a minimum of user interactions with the system. The number of interactions made by the user of a web application can be measured by the number of clicks required to answer a question.
Two sets of questions were selected to evaluate the systems quality. The first is the official set of question of the 2006 TREC competition. The second set of questions were selected to demonstrate features of GoPubMed not tested by the TREC questions. In the third section the results of past bibliometric analyses are compared to the results of GoPubMed analysis.

4.3.1 Answering TREC Question with GoPubMed

The TREC Genomics Track 2006 [118] is an annual activity of the information retrieval community aiming to evaluate systems and users. For the evaluation of biomedical search engines a new single task was developed that focused on retrieval of passages (from part to sentence to paragraph in length) with linkage to the source document. Topics are expressed as questions and the systems were measured on how well they retrieve relevant information at the passage, aspect, and document level. The participating systems returned passages linked to source documents. Judges rated the returned passages and grouped them by aspect.

The following questions of the TREC Genomics Track 2006 were answered using GoPubMed. There were three types of queries: (1) questions for the role of a gene or protein in a disease, (2) the inter-relation of biological entities and (3) the biological function of an entity.

For the first type of questions the query keywords were the protein name or synonyms if provided by the algorithm described in Hakenberg et al. [111] which is implemented in GoPubMed. Then clicking in the induced ontology tree on the respective disease the linked documents ranked according to the aspect are scanned. Only correct evidences on the first page of answers were considered.

For the second type of questions the two entities and their synonyms were entered as keywords and answers from the top categories were considered. The top categories were computed with the “cf-iidf” ranking described in section 4.2.6.

For the third type of questions the biological entity was entered as the query string and the respective biological function was selected in the ontology. Again only answers provided on the first page were considered. The TREC questions of all years are numbered. The questions of the year 2006 begin with 160 and end with 187.

Roles of a genes and proteins in a diseases

#160: What is the role of PrnP in mad cow disease?  "Since 2004, significant associations between bovine spongiform encephalopathy (BSE) susceptibility in cattle and frequencies of insertion/deletion (ins/del; indel) polymorphisms within the bovine prion protein gene (PRNP) have been reported. PMID: 18399944"

GoPubMed presents this answer after searching for "PrnP" followed by two intuitive clicks: (1) PrnP is a protein, so we use the protein name expansion by clicking the option: "Expand your query with synonyms for PrnP". (2) The disease branch of MeSH lists the official name among the top concepts, ENCEPHALOPATHY, BOVINE SPONGIFORM. Figure 4.9 shows the induced ontology for the query "PrnP". The disease ENCEPHALOPATHY, BOVINE SPONGIFORM, alias Mad Cow Disease, links to 77 publications mentioning the disease and PrnP.

The TREC benchmark lists 32 answers mentioning a mutation in PrnP playing a role in mad cow disease. This shows that the important aspect, mutated variants of PrnP are related to mad cow disease, can be found with a simple search in GoPubMed. Three intuitive user interactions. The induced ontology holds answers to many more aspects related to the
4.3. Evaluation of Ontology-Based Literature Search with GoPubMed

Figure 4.9: This screenshot shows the induced ontology for the query "PrnP" in GoPubMed. Encephalopathy, Bovine Spongiform, the official name for "Mad Cow Disease" is selected. It links to 77 publications in PubMed. Additionally, Scrapie, Creutzfeldt-Jakob Syndrome and other diseases are listed in the disease branch of MeSH.

Figure 4.10: This screenshot shows the first text snippet the user sees when clicking on Encephalopathy, Bovine Spongiform. The keyword "PriP" is a synonym of "PrnP" and is highlighted in yellow. The disease’s synonym, Bovine spongiform encephalopathy, is highlighted in green. The snippet is ranked highest because it is a recent publication plus keyword and concept is mentioned in the same sentence.
CHAPTER 4. ONTOLOGY-BASED LITERATURE SEARCH

Figure 4.11: A full abstract shown in the document view of GoPubMed for the query "BRCA1 AND BARD1". The abstract is shown when selecting concept DNA repair. Note that DNA repair is not mentioned literally in the text. Instead the Concept Recognition algorithm detected interstrand crosslink repair as a synonym of nucleotide-excision repair which is a descendant of DNA repair.

Inter-relation of biological entities

**#168: How does BARD1 regulate BRCA1 activity?** "BARD1 regulates BRCA1-mediated transactivation of the p21(WAF1/CIP1) and Gadd45 promoters. (PMID: 18243530)"

GoPubMed presents the title of this publication as an answer after searching for "BRCA1 AND BARD1" followed by one click on the option "Expand your query with synonyms for BRCA1, BARD1". The third text snippet clearly mentions the inter-relation of the two proteins. GoPubMed shows by default not the whole abstract. One sentence is selected as a text snippet which mentions keywords and concepts. In this example no concept was selected but the two keywords are mentioned in the text snippets.

GoPubMed classifies the 132 articles using MeSH and the Gene Ontology. The GO branch biological process lists important concepts such as regulation of progression through cell cycle with 67 evidences like: "The BRCA1 tumor suppressor exists as a heterodimeric complex with BARD1, and this complex is thought to mediate many of the functions ascribed to BRCA1, including its role in tumor suppression." Interestingly the snippet was ranked high because tumor suppression is a Gene Ontology synonym of regulation of progression through cell cycle.
Another important aspect is represented by the second biological process DNA repair. Clicking on it shows the evidence: "Cells deficient in the Werner syndrome protein (WRN) or BRCA1 are hypersensitive to DNA interstrand cross-links (ICLs), whose repair requires nucleotide excision repair (NER) and homologous recombination (HR). PMID: 16714450". The abstract, shown in figure 4.11, was ranked high because it mentions the keywords and the concept interstrand crosslink repair which was identified by the Concept Recognition algorithm. The concept is a synonym of nucleotide-excision repair which is a descendant of DNA repair. For the aspect “DNA repair” there are 52 more evidences listed in the induced ontology.

Biological function of entities

#177: How do Bop-Pes interactions affect cell growth? "The nucleolar PeBoW-complex, consisting of Pes1, Bop1 and WDR12, is essential for cell proliferation and processing of ribosomal RNA in mammalian cells. (PMID: 16738141)"

GoPubMed presents this snippet when searching for "Bop AND Pes" followed by a click on the option "Expand your query with synonyms for Bop, Pes" and filtering for the MeSH heading Cell Growth Processes. The advanced search feature in GoPubMed allows for filtering with concept branches. The advanced search Bop AND Pes +*mesh#“Cell Growth Processes” retains only documents mentioning cell growth. The above snippet is shown at position 1 out of six documents when using this advanced search query and expanding the protein names.

The TREC gold-standard lists snippets of the full text of PubMed Central documents as valid answers to the questions. For question #177 the benchmark lists 7 snippets which were accepted by the curators as an answer to the question. For example the passage: "Interestingly, a potential homologous complex of Pes1-Bop1-WDR12 in yeast (Nop7p-Erb1p-Ytm1p) is involved in the control of ribosome biogenesis and S phase entry. In conclusion, the integrity of the PeBoW complex is required for ribosome biogenesis and cell proliferation in mammalian cells. (PMID: 16043514)" was accepted to answer question number 177.

All questions of the TREC 2006 Genomics Track could be answered with GoPubMed. Section 6.5.3 in the appendix lists the questions and answers given by GoPubMed compared with one of the accepted TREC answers. It is important to note that the answers of GoPubMed are based on the abstracts and not on the full texts as used by the TREC participants.

Table 4.12 shows the summary of results of this evaluation. The column Advanced Query shows the query syntax submitted to the system for each query. It contains the keywords of the question plus the concept branch relevant to answer the question. The column "Ex" denotes whether the Protein/Gene name expansion was used or not. The Aspect column contains the concept under which the answer was found. IS gives the number of documents used to induce the ontology. RS is the number of documents linked to the aspect. Pos is the position in the snippet list the answer was found. UI is the number of user interactions required to find the question.

Despite the fact that this evaluation was carried out with the abstract texts only, in contrast to the full texts used during the TREC evaluation, GoPubMed is able to answer all questions with a minimum of user interactions required. The advanced queries used in this experiment can easily be replaced by a simple keyword search plus a click on the concept in the ontology. In most cases the concepts selected here are top categories of the queries. In the other cases a search in the background knowledge with the option "Find related categories ..." quickly locates the appropriate concepts.
The number of official TREC answers per question varies between 0 and 593 snippets. The answers mostly cover not only one aspect of the topic. For example one TREC answers to the question number 160 is related to mutations: "Nineteen mutations of the PrP gene are associated with inherited human prion disease... (PMID: Pmid: 7642588 Span: 19641-86)" another aspect, the pathogenesis in cattle and sheep, is covered by the correct answer: "bovine spongiform encephalopathy in cattle, and scrapie in sheep are members of a family of infectious neurodegenerative mammalian diseases known as the transmissible spongiform encephalopathies. During disease pathogenesis, a protease-resistant form of prion protein (PrP) accumulates in the brain and other tissues of infected animals... Pmid: 7852415 Span: 4349-483".

In GoPubMed the spectrum of relevant answers to the questions is reflected in the induced ontology. The user can explore the aspects by navigating through the induced hierarchy. The top categories helps to identify important aspects without the need to dive into the hierarchy.

4.3.2 Ontology-based bibliometric analyses with GoPubMed

Recently much research has been devoted to the analysis of the biomedical literature. This interest has been sparked by the growth in literature, but also by the availability of abstracts, full papers, and bibliometric data. Researchers have been specifically interested in automatically extracting information from free text such as protein names [91, 290, 124], ontology terms [247, 27, 64], and protein interactions [265, 90, 125].

Underlying all of the above textmining applications is the literature, which grows overall. But at a closer glance it turns out that some research areas shrink, while others take off. Bibliometric analyses aim to shed light on such developments and to identify emerging trends. Such analyses date back to the 60s [209] and typically focus on research topics [95], specific journals [31], or the researchers themselves [186, 187, 209, 110, 48, 34]. Garfield and Melino [95] investigate e.g. the research on programmed cell death. Despite programmed cell death being described 25 years ago, it took some 15 years until journals such as “Cell Death and Differentiation” emerged and the number of publications in the field in general took off. As an example for an analysis of a specific journal Boyack analysed the emergence and development of topics covered by PNAS [31]. A very active area of research aims to understand the social process of publishing by investigating co-author and co-citation networks [186, 187, 209, 110, 48, 34]. Such analyses allow one to identify authors in an organisation, who work interdisciplinarily and connect otherwise unconnected co-author networks [187], to animate citations of key publications over time [48], evolution of author and publication networks [34], and to understand how groups form [110]. All of these analyses are useful to take a birds-eye view onto research. This section links such analyses to the ontology-based literature search engine GoPubMed to support the discovery of trends on topics interesting for molecular biologists.

GoPubMed extracts GO terms from all 18.000.000 PubMed abstracts and allows users to explore their search results with the Gene Ontology. GoPubMed’s association of GO concepts with abstracts is a valuable resource to understand how a research topic - represented by a GO concept - develops. It shows how many articles were published over time, which authors are most prolific for the topic, which journals cover the topic best, and which countries publish most on the topic. The use of an ontology for these analyses is very important as it includes synonyms and subconcepts. As an example, Garfield and Melino [95] point out that during the 60s and 70s researchers in the US used “programmed cell death” while their European colleagues used “apoptosis”. In this analyses, these are treated as equivalent with the help of the underlying ontology. Also it is important to consider subconcepts as
some papers may mention GTPases in general, while others refer to specific GTPases such as Ran, Rac, Rho, etc. Again, the use of the ontology ensures that an analysis of GTPases will include all specific GTPases.

Besides research topics, authors and places were analysed. The results show their publishing activity over time and the topics covered. Finally, the whole biomedical literature was analysed to identify the journals, which mention most GO concepts. Assuming that GO captures the background knowledge of a molecular biologist, these are the most important journals for the molecular biologist.

The experiment was structured as follows: a bibliometric analysis focusing on “important topics” investigating apoptosis and endosome was carried out. Second, an investigation of “important places” shows how organisations and places can be classified, and the analysis of “important journals” summaries the main topics covered by a journal. Finally, the whole Gene Ontology was analysed for the 20 most important journals for a molecular biologist.

It is important to mention that any attempt of a quantitative analysis of the literature, however sophisticated, must be interpreted by informed judgment. Absolute citation frequencies may be misinterpreted. It is not the intention to judge the significance of the contributions of individual scientists or institutions.

Important topic: Apoptosis

In 1997, Garfield and Melino analysed the scientific literature on programmed cell death (apoptosis) [95] using the ISI’s Science Citation Index. Using the date of publication, frequency, citation and co-citation of papers, they analysed the development of the field, the countries most active, the main journals, and key authors. They found that there was a significant increased impact of articles on programmed cell death after 1990 and that it was one of the hottest topics in 1997. Some of the countries among the most active are the US, UK, Germany, Australia and France. Journals most actively publishing on apoptosis are Immunology, Blood, FASEB Journal Cancer Research, Biological Chemistry, PNAS and Oncogene. The most cited authors are AH Wyllie, SJ Korsmeyer and GT Williams as well as later on position 17 and 22 JC Reed and PH Krammer.

Such information is valuable to quickly get an overview over a new field. Although the author stated in a later addendum that they made a mistake so Strasser and Vaux were not mentioned although they had to be on rank 2 and 3. Also other research fields like nitric oxide and p53 are mentioned as similarly active research fields at this time.

Bibliometric analyses as above are valuable but difficult to produce. Especially at the beginning emerging trends may be known under different names. As Garfield and Melino point out, initially the term apoptosis was used more frequently in Europe while the US coined the same topic programmed cell death. Besides synonyms such analyses should consider papers, which do not mention apoptosis explicitly but implicitly.

As example consider papers, which mention RELEASE OF CYTOCHROME C FROM MITOCHONDRIA or CASPASE ACTIVATION. Since the release of cytochrome c from the mitochondrial intermembrane space into the cytosol leads to caspase activation and is an early step of apoptosis. Hence papers, which mention these terms should also be considered as covering apoptosis.

These two problems - the use of synonyms and the inclusion of specialisations of terms - can be addressed with an ontology, a structured, hierarchical vocabulary, which defines synonyms as well as is-a and part-of relationships. The Gene Ontology defines e.g. the synonyms APOPTOSIS and TYPE I PROGRAMMED CELL DEATH. Furthermore, it defines that CASPASE ACTIVATION is part of APOPTOTIC PROGRAM, which is part of APOPTOSIS.
4.3. EVALUATION OF ONTOLOGY-BASED LITERATURE SEARCH WITH GOPUBMED

Figure 4.12: PubMed abstracts per year mentioning apoptosis (including synonyms and specialisations of the term)

Since GoPubMed indexes all PubMed abstracts with the GeneOntology our analyses include the consideration of synonyms and specialisations. For apoptosis we find very similar results to Garfield and Melino. The exponential increase of publications after 1990 can be seen in figure 4.12. The figure also shows that this trend continued after 1997, when Garfield and Melino’s analysis was published. While Garfield and Melino claim that apoptosis is a hot topic, we can quantify this claim to some extent.

Considering the number of papers between 1991 and 1997 apoptosis ranks at position 16 in comparison to other GO concepts at the same level as apoptosis or deeper in the GO hierarchy. An analysis of the countries also confirms Garfield and Melino finding though Australia’s high rank in 1997 has diminished. All of the relevant journals identified by Garfield and Melino are confirmed by our analysis. Their six journals rank in the top 5 and at position 8 (Blood). The analysis reveals additionally Biochemical and Biophysical Research Communications and Nucleic Acids Research at position 6 and 7 as highly relevant for apoptosis.

The ranking for most actively publishing authors is different in this analysis than the results of Garfield and Melino. This is due to the fact that in contrast to them only the number of papers mentioning apoptosis could be used to rank the authors and not the citations and impact of the papers. However the two highly ranked authors Reed and Krammer also rank very high in the results. This shows that considering citations of articles gives in general a significantly different ranking than simply using term mentioning frequency. To improve this information about citations for each abstract would be needed.

One can conclude that this approach of indexing the usage of GO concepts in literature abstracts leads to very similar results when compared with the approach of Garfield and Melino. The main differences are: (1) that this approach is fully automated and always up to date, while their results date back to 1997, (2) synonyms and specialisations are considered, (3) and most of all this analysis is available online for any of the 24,000 GO concepts.

Important topic: endosome

Let us consider another example besides apoptosis, namely GO’s cellular component endosome, which includes subterms such as the early and late endosome. As shown in the bottom of Figure 4.13, one can see that research in this areas has steadily increased in superlinear fashion. Clearly research related to the endosome is a hot topic at the moment. As may be expected the literature is dominated by countries from North America, Europe and Japan. However, a small part is attributed to Singapore, which is significant due to its small size. Table 4.13 shows the main journals and authors.
Table 4.13: Statistics for ENDOSONE (Art. = number of articles containing ENDOSONE).
Top left: Most prolific journals. Top right: Most prolific authors.

<table>
<thead>
<tr>
<th>Journal</th>
<th>Art.</th>
<th>Author</th>
<th>Art.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Biological chemistry</td>
<td>777</td>
<td>P D Stahl</td>
<td>59</td>
</tr>
<tr>
<td>Cell biology</td>
<td>420</td>
<td>H J Geuze</td>
<td>52</td>
</tr>
<tr>
<td>Cell science</td>
<td>261</td>
<td>T Berg</td>
<td>44</td>
</tr>
<tr>
<td>Molecular bio. of the cell</td>
<td>250</td>
<td>J Gruenberg</td>
<td>44</td>
</tr>
<tr>
<td>Virology</td>
<td>180</td>
<td>I Mellman</td>
<td>44</td>
</tr>
<tr>
<td>PNAS</td>
<td>178</td>
<td>B I Posner</td>
<td>41</td>
</tr>
<tr>
<td>Immunology</td>
<td>152</td>
<td>J J Bergeron</td>
<td>40</td>
</tr>
<tr>
<td>Bioch. et biophy. acta</td>
<td>144</td>
<td>K Sandvig</td>
<td>36</td>
</tr>
<tr>
<td>European j. of cell bio.</td>
<td>134</td>
<td>M Zerial</td>
<td>36</td>
</tr>
<tr>
<td>Biochemical journal</td>
<td>131</td>
<td>G Griffiths</td>
<td>35</td>
</tr>
<tr>
<td>Traffic</td>
<td>114</td>
<td>B van Deurs</td>
<td>35</td>
</tr>
<tr>
<td>EMBO journal</td>
<td>113</td>
<td>R G Parton</td>
<td>35</td>
</tr>
<tr>
<td>American j. of physiology</td>
<td>104</td>
<td>A S Verkman</td>
<td>31</td>
</tr>
<tr>
<td>Bioch. &amp; biophy.res.comm.</td>
<td>90</td>
<td>H Stenmark</td>
<td>29</td>
</tr>
<tr>
<td>Cell and tissue research</td>
<td>90</td>
<td>S R Pfeffer</td>
<td>29</td>
</tr>
<tr>
<td>Others journals</td>
<td>6389</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Important place: Dresden

Bibliometric analyses can also be applied to get an overview over organisations and places. Evaluating the topics covered by publications whose affiliation mentions Dresden reveals e.g. that biomedical research in Dresden is focused on the following topics: antiporter activity, pregnancy, apoptosis, cell proliferation, viral nucleocapsid, cytosol, exogen, microtubule, spindle, fever, gastrulation, lactation, cytokinesis, endosome, autosome, vasodilation, nucleolation, phosphorylation, wound healing, dendrite, lipid raft, RNA interference, cytoskeleton, angiogenesis, cell migration, inflammatory response, mismatch repair, vacuole, collagen type I, fibrinolysis, insulin secretion, vascular endothelial growth factor receptor binding, phagocytosis, cellular respiration, pore complex, chromatin.

This gives an immediate impression and individual topics can be traced back to researchers and groups. E.g. RNA interference is a hot topic in Dresden with a high-throughput RNAi screening facility in place, which has lead to numerous publications including high-profile papers in Nature.

Important journal: Which are the 10 most frequently used GO terms in Nature, Cell and Science?

Similar to the analysis Boyack [31] carried out for PNAS, we can analyse other journals. As an example, we looked at the 10 most frequently mentioned terms in Nature, Cell, and Science. Some terms appear frequently in all of the major journals, like exogen, apoptosis, mutagenesis, cytokinesis, antiporter activity, DNA replication and phage assembly. Some terms are mentioned more often in one of the journals in comparison to others. E.g. Cell was found to list articles on transcription initiation, endoplasmic reticulum membrane, nucleosome, protein targeting, protein-ER targeting and regulation of cell cycle more frequently, which reflects its focus on molecular cell biology very well. Science (Weekly) was found to list abstracts containing T-cell activation, nucleic acid transport, regulation of action potential, carbon dioxide transport and response to carbon dioxide more than other journals.
4.3. EVALUATION OF ONTOLOGY-BASED LITERATURE SEARCH WITH GOPUBMED

**Figure 4.13**: Screenshot of GoPubMed statistics for the concept endosome. Top: Most prolific countries. Bottom: Articles on endosome over time.

and Nature *nucleic acid transport, myosin, regulation of action potential and generation of action potential.*

**20 Journals for the Molecular Biologist**

The PubMed database is the main source for literature abstracts in the biomedical field covering thousands of journals, dating back to the 60s, including millions of authors, and several million abstracts. The GeneOntology on the other hand is a large vocabulary of over 24,000 terms covering many aspects of interest for a molecular biologist. Assuming that GO reflects the topics of interest for a molecular biologist, we wish to analyse how much of the PubMed literature might be actually of interest, which GO terms are mentioned most frequently, and which journals are the most relevant.

**How many articles mention GO terms?** PubMed is growing at a tremendous pace and in 2004 alone there were 598278 new abstracts registered. But how much of this is relevant to a molecular biologist as there are is also very general articles such as "Relative efficacy of the proposed Space Shuttle antimotion sickness medications", "Point-counterpoint: should physicians accept gifts from their patients? No: Gifts debase the true value of care", "The why and wherefore of empowerment: the key to job satisfaction and professional advancement". Nonetheless, our analysis shows that nearly half (47.9%) of all articles in English mention at least one GO term. This figure is quite high, as e.g. Nature, which is certainly the accepted journal for any molecular biologist, has only in 38.5% of its articles at least one GO term.
Which are the 10 most frequently used GO terms per year? A specialization level of greater than 5 was chosen in the Gene Ontology to compare research topics in two years 1972 and 2004. That means high-level ontology terms were ignored.

GO concepts mentioned in abstracts frequently in both year were: kidney development, lung development, response to X-ray, response to virus and phosphorylation. Articles from 1972 mentioned additionally following GO concepts more frequently: microsome, cellular respiration, phage assembly, antiporter activity, central nervous system development, DNA replication, alkaline phosphatase activity, nucleic acid transport, regulation of balance, renin activity, lysosome, salivary gland (determination/morphogenesis) and ovulation.

In contrast to that the following new topics were subjects of research in 2004: apoptosis, donor preference, endothelial cell (activation/morphogenesis), equator specification, exogen, angiogenesis, visual perception, inflammatory response, interferon-gamma biosynthesis, response to reactive oxygen species.

Figure 4.14: Topic evolution for the most mention GO concepts since 1972. The list of GO concepts is ordered by the research interest over time. Topics like apoptosis, transduction, donor preference, cell proliferation and necrosis are of increasing research interest whereas liver development, pregnancy and kidney development show an relatively decreasing rate of mentions in PubMed abstracts.

In figure 4.14 the evolution of the most frequently used GO concepts over time is shown. One can clearly see that topics like apoptosis, transduction, donor preference, cell proliferation and necrosis are of increasing research interest whereas liver development, pregnancy and kidney development show an relatively decreasing rate of mentions in PubMed abstracts.

Which 20 journals mention the most GO terms? Finally, let us turn to the “20 journals for the molecular biologist”, which are as partially shown in Table 4.14. Biological chemistry, PNAS, Biochimica et biophysica acta, Immunology, Biochemical and biophysical research communications, American journal of physiology, Biochemistry, Biochemical journal, Brain research, Cancer research, Virology, FEBS letters, Bacteriology, Blood, Endocrinology, Cell biology, European journal of biochemistry, FEBS, Infection and immunity,
Molecular and cellular biology, and Nature. Other journals such as Science rank at position 32, EMBO at 26, Cell at 52. It is interesting that many biochemistry journal rank in the top positions.

<table>
<thead>
<tr>
<th>Pos.</th>
<th>Journal</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Biological chemistry</td>
</tr>
<tr>
<td>2</td>
<td>PNAS</td>
</tr>
<tr>
<td>3</td>
<td>Biochimica et biophysica acta</td>
</tr>
<tr>
<td>4</td>
<td>Immunology</td>
</tr>
<tr>
<td>5</td>
<td>Biochemical and biophysical research communications</td>
</tr>
<tr>
<td>6</td>
<td>American journal of physiology</td>
</tr>
<tr>
<td>7</td>
<td>Biochemistry</td>
</tr>
<tr>
<td>8</td>
<td>Biochemical journal</td>
</tr>
<tr>
<td>9</td>
<td>Brain research</td>
</tr>
<tr>
<td>10</td>
<td>Cancer research</td>
</tr>
<tr>
<td>11</td>
<td>Virology</td>
</tr>
<tr>
<td>12</td>
<td>FEBS letters</td>
</tr>
<tr>
<td>13</td>
<td>Bacteriology</td>
</tr>
<tr>
<td>14</td>
<td>Blood</td>
</tr>
<tr>
<td>15</td>
<td>Endocrinology</td>
</tr>
<tr>
<td>16</td>
<td>Cell biology</td>
</tr>
<tr>
<td>17</td>
<td>European journal of biochemistry / FEBS</td>
</tr>
<tr>
<td>18</td>
<td>Infection and immunity</td>
</tr>
<tr>
<td>19</td>
<td>Molecular and cellular biology</td>
</tr>
<tr>
<td>20</td>
<td>Nature</td>
</tr>
</tbody>
</table>

Table 4.14: 20 journals for the molecular biologist.

This shows the high relevance of the selection of the journals. In the online version the user can browse all abstracts of a selected journal mentioning a previously selected GO term via GO.

Summary of analysis

As early as 1965 [209] researchers have been interested in bibliometric analysis as they give an overview over a research field and can help to discover trends and developments in research. With the availability of online literature databases such as PubMed, this interest has increased further [95, 31, 186, 187, 209, 110, 48, 34].

A key problem for bibliometric analyses is the use of synonyms and specialisations of concept labels. To address this problem, the GeneOntology was used. Using this terminology, synonymous terms such as apoptosis and programmed cell death and specialisations such as caspase activation, which is an early stage of apoptosis can be grouped. The Concept Recognition algorithm was used to identify GO concepts. Thus, all GO concepts for all PubMed abstracts were available. This rich resource was used for a large-scale ontology-based bibliometric analysis of PubMed.

The method is in line with existing, but often out-dated, analyses such as Garfield and Melino’s work on apoptosis [95]. This shows that the overall method is sound.

The method is available at large scale, for all GO concepts and it considers their synonyms and specialisations in the analysis. The online analysis is always up-to-date. The results are publicly available via the ontology-based search engine GoPubMed. For each GO
concepts this service provides the most prolific journals, authors, and countries, as well as the development of publications over time.

Besides considering topics, it was illustrated how to use this resource to get an overview over journals and places and organisations including a list of the 20 journals, which mention most GO terms, and can be considered as the main publications for molecular biologists.

This study presents the first ontology-based, large scale, online available, up-to-date bibliometric analysis for topics in molecular biology represented by GO concepts.

4.4 Summary and Discussion

The objective of this work is to study how ontological background knowledge can be used for literature mining, question answering and bibliometric analyses. One goal is to develop reasoning algorithms for mining information from text based on semantic markups in literature abstracts. Another question is who ontologies can be used to answer biomedical questions with a corpus of literature abstracts. Finally it is evaluated who a corpus of annotated literature abstracts can be used for automatic trend analysis and bibliometric statistics.

An important finding of this study is that domain ontologies such as the Gene Ontology and the Medical Subject Headings are a rich source for terminological and relational knowledge in the biomedical field which can be used for literature mining.

4.4.1 Induced Ontology

The algorithms developed in the previous chapter are used to establish semantic links to literature abstracts stored in the citation database PubMed. The result is a annotation database containing more than 420,000,000 semantic markups of text ranges in scientific texts. The biomedical concepts come from the Gene Ontology and MeSH. The advantages in using these controlled hierarchical vocabularies are: (1) they provide many synonyms and different spelling variants and (2) they relate all concepts in a taxonomic hierarchy. For example MeSH defines the concept DISEASES and organizes all diseases as descendant concepts while grouping them into more specific subconcepts like BACTERIAL INFECTIONS. The Gene Ontology defines two relations. The is-a relation and the part-of relation. The is-a relation organizes concepts in a taxonomic structure. Most importantly the True Path Rule imposed by GO asserts that the transitive character of the is-a relation must always reflect biologically correct statements. The Gene Ontology was initially design to annotate genes and gene products in databases. This study makes the assumption that a document which is relevant for a GO concept must also be relevant for all ancestors of the concept.

Ontologies do not define only subsumption relations. The Gene Ontology also defines part-of relations. The True Path rule is also asserted for part-of relations. If the RHOPTRY MEMBRANE is part of the RHOPTRY and RHOPTRY is part of the APICAL COMPLEX. Then it must also be biologically correct to state: “RHOPTRY MEMBRANE is part of the APICAL COMPLEX”. Using the same analogy as for the is-a relation this study assumes that if a document is relevant for RHOPTRY MEMBRANE then it must also be relevant for APICAL COMPLEX.

Hierarchical relations among MeSH descriptors are indicated by tree numbers assigned to the descriptors. For example, the tree number [C01.252.410.040.552.846] indicates that TUBERCULOSIS is a descendant of MYCOBACTERIUM INFECTIONS [C01.252.410.040.552]. MeSH does not define the relations between the concepts formally. However it can be assumed that a document relevant for TUBERCULOSIS is also relevant for MYCOBACTERIUM INFECTIONS.
4.4. SUMMARY AND DISCUSSION

The Induced Ontology, introduced in this work, is defined as a graph connecting all concepts identified in a document collection with their ancestors up to the root concepts of the terminologies. Associated to each node in this graph are the documents mentioning the concept or a descendant. This allows a new way of exploring a large collection of documents. A related approach is used by Vivisimo [263]. However no background knowledge in the form of controlled terminology is used. Instead Vivisimo uses document clustering to organize web resources in a hierarchical way. The advantage of using a controlled vocabulary is a consistent navigation graph. While the labels assigned a document cluster are text snippets from a representative document the labels of the induced ontology are the controlled labels of defined semantic concepts with definitions and relations to other concepts. Concepts display at each level of the Induced Ontology have the same super-concept. The labels of document clusters at the same level group document according to some distance measure but are not necessarily siblings or represent the same abstraction. The semantic concept used in the induced ontology are the same used to annotate other resources such as biological experiments [13]. They represent a consensus in meaning. However no background knowledge in the form of controlled terminology is used. Instead Vivisimo uses document clustering to organize web resources in a hierarchical way. The advantage of using a controlled vocabulary is a consistent navigation graph. While the labels assigned a document cluster are text snippets from a representative document the labels of the induced ontology are the controlled labels of defined semantic concepts with definitions and relations to other concepts. Concepts display at each level of the Induced Ontology have the same super-concept. The labels of document clusters at the same level group document according to some distance measure but are not necessarily siblings or represent the same abstraction. The semantic concept used in the induced ontology are the same used to annotate other resources such as biological experiments [13]. They represent a consensus in meaning.

An interesting question is how to rank the relevance of documents for a concept. This question was addressed by defining the notion of informativeness for nodes and paths in the induced ontology graph. Two measure were developed: (1) The Richest Path, which are those descendants of a concept linked to the most documents at each level in the hierarchy, and (2) The Top Categories, which are the concepts best representing a sub-branch of the Induced Ontology.

Both measures use statistical information of the document collection. The document contained in the Richest Path are those mentioning descendants of the most mentioned concepts. In other words, the Richest Path reveals the most frequently discussed topics in the document collection. As asserted knowledge is more frequently mentioned in the literature than new or uncertain knowledge the Richest Path is a good way to explore the most obvious facts in a document collection. This is again different from hierarchical document clustering based on similarity because the relations between concepts in the Richest Path are formally defined in the concepts definitions. Thus the path is self-explaining.

For the Top Categories, single concepts independent of the location in the ontology graph, it was evaluated whether hierarchical information can improve the quality of the standard tf-idf ranking for the relevance. The results show that there was no improvement in the quality of the ranking when weighting the relevance of a concept with the tf-idf measure. In other words it was of no help to sum up the frequencies of mentioning of all descendant concepts when computing the rank for a concept.

This result was expected for the “tf” part of the formula (induced local count) as the weight of the “idf” part is diminish drastically. Using the “induced global count” did not result in a better ranking either.

The Top Categories can be computed for any branch of the Induced Ontology. This allows to display the most relevant concepts for any sub-topic represented by an ontology
4.4.2 Question Answering

The hypothesis was that ontology-based search can answer biomedical questions. The evaluation of 28 real biomedical questions showed that it is possible to find highly relevant answers to all questions using the ontology-based search engine at www.GoPubMed.org. A typical way to answer a question like **Which diseases are related to HIV?** is to search the document corpus for documents mentioning the object "HIV" and then navigate to an appropriate concept representing the subject of the question **DISEASES**. The Induced Ontology shows the children of the concept **DISEASES** and the Top Categories denote the most relevant descendants of the branch. As all descendants are diseases or types of diseases the Top Categories contain highly relevant concepts for the answers to the question. Moreover, the Induced Ontology holds textual evidences for all identified concepts. In other words the Induced Ontology knows in which sentences the mentions of the relevant concepts were made by the author. The subset of all evidences for one concept was named aspect. An aspect is the sum of facts, theories and hypothesis which relate a concept to a topic. A document is relevant for an aspect if its content explicitly states a relation between the topic and a concept.

An interesting sub-task was to define the relevance of each document linked to an aspect. In other words: Which documents are likely to provide the best evidence for an aspect. For example the fact that HIV patients often suffer from hepatitis is discussed in a publication with the title **"Natural history of hepatitis C virus infection in HIV-infected individuals and the impact of HIV in the era of highly active antiretroviral therapy: a meta-analysis."** PMID: 18784461.

The notion of proximity ranking was introduced. The measure ranks those documents high which mention keywords, e.g. of a search query, in close distance to the concept or descendant of an aspect. In the above example "HIV" is mentioned in the same sentence with **HEPATITIS** C which is a descendant of **DISEASE**s in the MeSH hierarchy. The manual evaluation of 28 TREC questions showed that for most answers at least on valid evidence could be found within the first 3 documents. See table 4.12 for details. This shows that ontology-based search can answer biomedical questions efficiently. In most cases a simple keyword search, the selection of a Top Category and the reading of maximal 3 short evidence snippets could locate a quotable text passage from a literature abstract to answer the question.

Ontology-based question answering is limited by the quality of the semantic annotations and the content of the background knowledge. Imperfect Concept Recognition can result in wrong or missing concepts in the Induced Ontology. The result is that some answers can not be found or that too many irrelevant concepts are contained in the Induced Ontology. Another reason why some questions can not be answered is that the background knowledge is incomplete or terminology of a sub-domain is missing at all. The current implementation for example does not use any ontology covering protein or mutations of genes. Questions for such types of concepts must be answered with normal keyword searches, i.e. the query must be manually extended with synonyms and descendant concepts.

4.4.3 Bibliometric Statistics

Finally the goal was to use the entirely annotated document corpus to analyse bibliometric trends in the literature. This chapter presents the first ontology-based, large scale, online
4.4. SUMMARY AND DISCUSSION

available, up-to-date bibliometric analysis for topics in molecular biology represented by GO concepts.

The method is in line with existing, but often out-dated, analyses such as Garfield and Melino’s work on apoptosis [95]. This shows that the overall method is sound.

The method is available at large scale, for all GO concepts and it considers their synonyms and specialisations in the analysis. The online analysis is always up-to-date. The results are publicly available via the ontology-based search engine GoPubMed. For each GO concept this service provides the most prolific journals, authors, and countries, as well as the development of publications over time.

It is important to point out that quantity is not an indicator for quality for scientific publications. The current bibliometric analysis does not take citation indexes into account. It is planned to take the SCImago Journal Index into account when computing the bibliometric statistics.

Summary of contributions.

To summarize the contributions made in this chapter:

• Recover implicit knowledge: The induced ontology contains a large amount of useful information. This chapter shows how reasoning over background knowledge can be used make implicit knowledge in PubMed abstracts explicit.

• Question Answering: This chapter shows that ontology-based search with GoPubMed is able to answer real biomedical questions with a minimum of user interactions in a web browser.

• Bibliometric Statistics: This chapter presents the first ontology-based, large scale, online available, up-to-date bibliometric analysis for topics in molecular biology represented by GO concepts.
Chapter 5

Design of GoPubMed

The goal is to design a semantic search engine which enables an intelligent search using background knowledge of ontologies. The system should support to quickly answer questions and survey large collections of relevant documents.

1) Requirement Analysis: General requirements for an ontology-based search engine and specific requirements in the biomedical publishing domain are discussed.

- The stakeholders of the system are identified.
- The application domain is characterized.
- The goals for the implementation are identified.
- Function and nonfunctional requirements are identified from different viewpoints.
- Acceptance criteria are defined.
- Typical use cases are documented.

2) Design: The top-level architectural design for GoPubMed is documented.

- Different variants of the GUI layout are discussed.
- The top-level architectural design is documented using static and dynamic models.
- The main components are characterized.

3) Implementation: GoPubMed is now a mature software project. Versions 2.0 and 3.0 are based on implementations done for this thesis. The current system is maintained and further developed by a spin-off company of the TU Dresden which licensed the technology.

- Four major releases of the software are described.
- The commercialization of the technology and surveys document the user acceptance of the system.
- Three offsprings using GoPubMed’s technology are described.
CHAPTER 5. DESIGN OF GOPUBMED

5.1 Introduction

Finding relevant literature is an important and difficult problem. The amount of literature available online today is enormous. Ingenta (www.ingenta.com), an online index of 17,000 periodicals, has 7 million articles going back to 1988. Infotrieve (www.infotrieve.com) indexes over 20,000 journals with 15 million citations. CiteSeer (www.citeseer.com), a digital library, covers over one million articles and 22 million citations. Other important examples of literature search engines are Google Scholar, Scopus, Scirus and Forschungsportal.Net. Databases for scientific literature are growing at an astonishing rate. PubMed, a biomedical literature database, has grown by 754,003 cited documents in the last year and covers now more than 18 million abstracts of scientific literature, although about half of them are retractions and corrections [247]. With 871 million searches in PubMed in 2007 it is clear that researchers spend a considerable amount of time searching the scientific literature.

Great quantities of knowledge and information are available to researchers through millions of documents. But without effective ways of accession a lot of it will remain unnoticed by the readers due to the overwhelming amounts of text.

Classical search engines have limitations.

- Classical search engines do not provide search results spread over multiple documents. The answers are a ranked list of single documents represented as a text excerpt and a URL.

- The search result is a ranked hitlist. The costs for the user, the time spend to find relevant documents, quickly add up if the keywords are relevant for multiple topics. To increase the precision the user has to refine his query iteratively, which typically reduces recall at the same time.

- Results comprising many relevant documents are poorly represented by a few documents in the beginning of a hitlist.
5.1. INTRODUCTION

• Classical search engines are unaware of synonyms and relational information of common terminology. Some engines aim at producing a higher recall by expanding the query with synonyms, this technique typically reduces the precision of the systems.

• Meta information characterizing the entire collection of relevant documents is not part of the response of a classical search engine.

The goal is to design a semantic search engine which enables an intelligent search using background knowledge of ontologies. The system should support to quickly answer questions and survey large collections of relevant documents.

A number of systems are using semantic technologies, introduced in section 2.3, to improve web search. The knowledge encoded in ontologies is also use in other fields than literature search. Section 2.5 describes ontology-based tools in the life sciences. Some systems available on the web are closely related to GoPubMed. The web search engine Vivisimo\(^1\) uses document clustering to provide users with a hierarchy to browse relevant documents. No background knowledge in the form of ontologies or terminologies is used. The medical search engine MedStory\(^2\) groups results into top level categories and offers query refinements based on frequent phrases in the relevant documents. No definition is provided and no relation to other concepts. Powerset\(^3\) uses the background knowledge of Wikipedia to answer full sentence questions. Powerset analyzes the document corpus using Natural Language Processing techniques. Section 2.7 introduces biomedical textmining methodologies. The user enters a full question and the system responds with a list of possible answers in the form of text passages from Wikipedia. The background knowledge used is stored in natural language text not in structured ontologies. The user needs to be able to formulate a question containing enough information for the system to find relevant answers and the system must be able to correctly parse the sentence. It is often difficult for the user to formulate a precise question and often it is even more difficult for parser to understand a complex question.

There is currently no other web search engine using ontology background knowledge to organize search results on the web. The existing paradigms can be classified as keyword-based and natural language based (NLP). Figure 5.1 shows keyword-based and natural language based paradigms in relation to the ontology-based paradigm. The answers of keyword and NLP-based searches are typically single documents, while ontology-based searches offer an outline of all relevant documents. NLP and ontology-based searches use semantic technologies to structure text, while keyword-based search is based on character n-grams. Keyword and ontology-based searches take word phrases as input, while NLP search takes full questions as input.

An ontology-based literature search system serving hundreds of users, handling thousands of concepts and indexing millions of documents is a medium sized software project. The current version of GoPubMed has 371,477 lines of code and 916 Java classes. It is developed and maintained by six full time engineers. A structured software development process is necessary to engineer a healthy and extensible system.

This chapter discusses general requirements for an ontology-based search engine and specific requirements in the domain of the life sciences. The requirements specification for the academic product GoPubMed and its architectural design are documented. GoPubMed is online and freely available at www.gopubmed.org.

---

\(^1\) vivisimo.com
\(^2\) medstory.com
\(^3\) powerset.com
Figure 5.1: Three search paradigms

5.2 Requirements Analysis

The goal was to design a system that supports the answering of biomedical questions by processing search results of PubMed using the background knowledge of biomedical ontologies. The system must be capable of serving hundreds of users, ten thousands of concepts and millions of documents while responding immediately.

5.2.1 Stakeholders

The development process of the software involved various individuals, groups and institutions. The free biomedical search engine is primarily a research platform. Numerous offsprings were developed as scientific prototypes, e.g. GoCell, GoPatents, GoWeb or Go3R. Members of the academic research group are mainly interested in easy programming APIs separating them from implementation details and allowing quick prototyping. Their problems are mainly concerned with simply programming access to functionality of the platform and simple access to the underlaying background knowledge and document resources. They prefer high level programming languages such as Java or scripting languages to access the functionality and the data. Academic staff can not maintain large software systems. The biggest potential for GoPubMed as a research platform lies in the development of the ontology-based paradigm. The risk is that new ideas might require the re-design of the entire architecture. As a consequence researchers may start developing separate branches which are difficult to merge into the main branch.

The university is interested in high quality publications documenting the research advances made by the academic staff. At the same time it seeks awareness through the popularity of the search engine and the acquisition of new research projects. The university needs to make sure that the academic product meets all legal regulations. It is interested in a barrier free accessibility. The university can provide hardware infrastructure to serve many users. A limitation is that an academic product can not advertise on the website.
5.2. REQUIREMENTS ANALYSIS

A spin-off company was funded to maintain the search engine. The company is interested in selling separate products based on the latest developments. The spin-off licensed the technology from the university. It is interested in developing business models allowing a high revenue from such products. The close connection to research groups allows efficient technology transfer. A constraint for the company is that any development carried out by the company’s staff must be profitable in order to develop a healthy enterprise. Commercial interests might conflict with academic interests. Separate development branches might become difficult to merge.

The developers of the company are interested in well documented and maintainable code. New developments might need to be re-implemented and tested whether they meet the previously identified requirements. Academic staff is primarily interested in new features rather than optimal performance.

A scientific literature search engine relies on public citations databases such as PubMed from the NCBI. The NCBI is a tax supported governmental institution. Their interest is in the increased usage and popularity of their database and in keeping a high quality for their users, which are mainly biomedical researchers around the world. License agreements are made with external users of the database. The license agreement for PubMed allows a limited number of queries on the their database per second. Another restriction of such databases is that the citations served to and processed by an external tool must not be altered and up-to-date at all times. The NCBI is interested that third-party services maintain the the NCBI’s high quality standards. The main potential of citation databases is the integration of the scientific work of many researchers and providing a means to search for keywords and meta-information.

Users of literature databases are interested in relevant search results with a minimum of user interaction. The integrity of the provided data as well as the relevance of the entries is most crucial. User of PubMed are potential users of GoPubMed as well. Users of GoPubMed expect the same search results as in PubMed. A biomedical search engine based on PubMed must ensure the correctness and completeness of the search results. Users need simple access to the original papers directly or via their subscriptions. The interest of users is the benefit provided by an ontology-based search engine over a the classical search engine. The benefit the search engine can provide is more relevant document with fewer user interactions. Another benefit for users when they need to spend less time to find the relevant information. A big potential comes from users annotating web content while browsing articles.

Another group of stakeholders are the developers of domain ontologies. Their main interest are not ontology-based search technologies. Therefore they design their ontologies unaware of requirements of such tools. The Gene Ontology is designed to describe attributes of genes and gene products but not text. On the other hand they aim to create a common vocabulary in this domain and provides therefore a rich source of terminological information. MeSH is primarily concerned to categorize biomedical abstracts. Other ontologies are mainly designed for automatic reasoning and lack to provide useful hints to connect to natural language. The relations defined in ontologies reflect their intended use. Such relations might not be useful for ontology-based search. Taxonomic relations are currently in the main focus of ontology-based search. The biggest potential comes from the increasing amount of computable background knowledge. It is the main hypothesis that this structured and computable background knowledge improves literature search and overcomes limitations of classical search.

Network and database administrators are interested in maintaining the security of the infra-structure. Maintenance of the systems by external developers must not put the security of the university’s infrastructure at risk. A great potential comes from their expertise in setting up a professional hardware and database infra-structure and the optimization of it.
CHAPTER 5. DESIGN OF GOPUBMED

5.2.2 Domain Analysis

The Biomedical Publishing Domain is shaped by large publishers offering thousands of electronic journals on a subscription basis to researchers. Each journal has a focus on the selected research domain. Authors of research articles transfer their copyrights to the journals which in turn provide a peer review process to ensure high quality of publications. Additionally they provide the infrastructure to make the publication accessible over a long period of time. Journals seek a high reputation by attracting high quality papers. Their goal is the limit access to the publications to their subscribers and attract as many subscribers as possible. Authors want their work read by as many colleagues as possible. Recently some journals take an alternative approach for publishing papers. The authors pays a fee for a publication in an open-access journal. The fee is used to cover the expenses of the editing and reviewing process. The published papers are freely accessible to any researcher. The biggest potential of electronic journals is the provided literature in computable formats. The biggest limitations arise from non-free journals limiting access to subscribers.

Citation databases provide free access to literature abstracts of open and non-open access journals. A keyword based search can be executed on the title, abstracts and other meta-data of the publication. An interested reader needs a subscription to the journal if he/she wants to read the full text of the publications. The biggest potential of citation databases is the centrally managed and comprehensive representation of the available literature with simple access via web-services. A limitation is the limited information available for each article.

5.2.3 Problem description

Classical search engines have limitations. Classical search engines can not provide answers which are spread over multiple documents. The response to a keyword search is a ranked list of references. Each reference links to a single document which contains relevant information. The pieces of information in separate documents are not aggregated or inter-linked.

A problem for the user occurs when a keyword is relevant for different topics, e.g. “Ken and Barbie” may refer to children toys or to genes regulating genitalia development. A classical search engine is not aware of such different topics. It remains the users task to filter out irrelevant documents by reading them and manually identify discriminating terminology to refine the initial query string. The user needs to deal with boolean logic syntax of the particular search engine. Studies have shown that most users have great difficulty specifying queries in Boolean format and often misjudge what the results will be [105, 130]. The process may require several iterations until irrelevant documents are filtered out.

Another problem occurs if the number of relevant documents exceeds one browser page. Users almost never look beyond the first page of search results and most of users pay only attention to the first two snippets on a result page [104]. Users rather re-formulate the query than scrolling down. However results comprising hundreds of relevant documents are not adequately represented by a handful of text snippets.

Simple keywords searches for concepts with many synonyms have the problem that they miss many relevant documents. The user has the problem to collect all known terminology comprising a topic. For example MeSH lists more than 6,500 diseases and their synonymous terminology. Classical search interfaces are not designed to include hundreds of query strings. Automatic query expansion helps to solve this problem but results typically in much larger result sets. This increases the problem that large result sets are poorly represented by a few documents in the beginning of a hitlist.

Meta information of a document collection such as the origin of the documents, e.g. ge-
5.2. REQUIREMENTS ANALYSIS

Ogographic location or research affiliations, or bibliometric information, e.g., age of documents are not part of classical search results. A user of a classical search engine needs to extract meta information manually from a large set of documents in order to compile bibliometric statistics. This problem is additionally hampered by the fact that the number of queries allowed per day is typically limited for individual IP addresses.

To summarize the problems of classical search engines:

- Classical search engines do not provide search results spread over multiple documents. The answers are a ranked list of single documents represented as a text excerpt and a URL.
- The typical search result is a ranked hitlist. The costs for the user, the time spend to find relevant documents, quickly adds up if the keywords are relevant for multiple topics. To increase the precision the user has to refine his query iteratively, which typically reduces recall at the same time.
- Results comprising many relevant documents are poorly represented by a few documents in the beginning of a hitlist.
- Classical search engines are unaware of synonyms and relational information of common terminology. Some engines aim at producing a higher recall by expanding the query with synonyms, this technique typically reduces the precision of the systems.
- Meta information characterizing the entire collection of relevant documents is not part of the response of a classical search engine.

5.2.4 Goals

The goal is to overcome limitations of classical search. With the availability of large amounts of text in the web coupled with powerful indexing techniques and sophisticated concept recognition algorithms it becomes possible to used background knowledge of controlled terminologies such as ontologies and taxonomies to facilitate search.

Ontology-based search requires large text corpora as they are now easily accessible in the web. Large corpora require efficient indexing mechanisms. Lucene is one example of an open source indexing project capable of handling millions of documents. Ontologies and taxonomies are required to control the vocabulary of domains investigated by independently working researchers worldwide. Dozens of domain ontologies have been developed and are freely available. Indexing natural language texts with semantic concepts requires sophisticated information extraction techniques as they have been developed for many areas.

The goals of this software project are defined as follows:

- Overcome limitations of classical search, offer a solution to each problem identified for classical search engines
- Provide biologists with powerful scholarly search tool. A PubMed user should be able to easily use GoPubMed without loss of trust in the quality of the results.
- The use of background knowledge for search should be intuitive. The learning curve for a biologist familiar with the domain should be low.
- The system should be able to handle as many documents as modern classical search engines without the loss of functionality.
• The system should be easily configurable to allow experimenting with new research prototypes.

The goal is to design a semantic search engine which enables an intelligent search using background knowledge of ontologies. The system should support to quickly answer questions and survey large collections of relevant documents.

5.2.5 User requirements

A number of functional and non-functional requirements can be identified. The requirements were collected from different viewpoints including the users viewpoint (FR01 - FR16), the authors viewpoint (FR17-FR19), the developers viewpoint (FR20-FR26), administrators viewpoint (ER01 - ER02), institutions viewpoint (ER02 - ER03), licensing client (FR17-FR36, QR01 - QR08, ER01 - ER02).

Functional Requirements

Functional requirements may be calculations, technical details, data manipulation and processing and other specific functionality that show how a use case is to be fulfilled.

FR01: Keyword Search
Retrieval of PubMed documents via keyword search with same query syntax as PubMed.

FR02: Induced Ontology
Hierarchically organized outline of all articles for a PubMed query.

FR03: Top Categories
Short list of highly relevant concepts for the query result shown.

FR04: Quantitative Indicators
Graphical indicators shown how many relevant documents fall into a branch of the background knowledge.

FR05: Web Access
The search engine should be accessible through a standard web browser with standard JavaScript functionalities, i.e. the mainly used browsers such as FireFox and Internet Explorer.

FR06: Snippet presentation
The document results should be presented with the citation title and the abstract.

FR07: Titles only
PubMed offers a view showing only document titles. Users frequently request this feature. GoPubMed should offer a similar view.

FR08: Link to original sources
A browser link should refer to the original documents without the need to type anything.

FR09: Highlighting
In the document view recognized concepts and entities should be emphasized. This includes relevant concepts and query keywords.

FR10: Concept Definitions
Offer textual definitions of automatically recognized concepts in the abstracts.

FR11: Export formats
Users need a mechanism to export their search results to popular formats such as EndNote, Bibtex and XML.

FR12: Personalization
The web-interface should offer a clipboard and a search history per user.
FR13: Advanced Search

Experienced users need to be able to formulate advanced searches which automate the navigation in the ontology. For less experienced users a web form should support this.

FR14: Search in Background Knowledge

Users need to be able to search for concepts in the ontologies.

FR15: Query Expansion

Expansion of query string for recognized entities.

FR16: Bibliometric Reports

For any query result shown a bibliometric report comprising top authors, countries, cites, publication volume over time.

FR17: Include Publication Quality

Compile reports weighting the quality of the publication according to a public citation index.

FR18: Collaboration Network

Visualize co-authorship among researchers.

FR19: Author disambiguation

Distinguish authors with same name and initials.

FR20: Common Code Base

All spring-offs should be based on one code-base with minimal need for adoptions to other branches when new features are added.

FR21: Hot Deploy

Allow deploy of new features without restart of whole system.

FR22: Configure&Play Resources

Document stores, annotators and ontologies can be added with simple configuration.

FR23: Multiple Servers

Tasks can be shifted to different machines.

FR24: Hot Debugging and Instrumentalization

Allow to debug the system remotely and monitor health status of separate components.

FR25: Problem Alerts

The system monitors itself and alert upon unusual states.

FR26: Invulnerability

Denial of Service attacks and misuse of the system is detected.

FR27: External Visualization

Allow import of report data into external visualization tools.

FR28: Extend Background Knowledge

Allow editing background knowledge.

FR29: Author Feedback

Allow authors of publications to give feedback on annotation quality.

FR30: Private Queries

Users want to conceal what they are searching for.

FR31: Local Search

Clients want to search local document repositories.

FR32: Multiple Ontologies

Users want to use more than one ontology at the same time.

FR33: Import Ontologies

Users want to import OWL, OBO and RDF ontologies.

FR34: Generate Ontologies

Users want to generate new ontologies or extend branches based on texts they have.

FR35: Ontology Versioning

Keep track of changes in the editable ontologies. Concept Mapping required.

FR36: Merge Ontologies

Merge ontologies with similar concepts. Concept Mapping required.

Quality Requirements

Functional requirements are supported by non-functional requirements, which impose constraints on the design or implementation (such as performance requirements, security, or
reliability).

QR01: Indexed document up-to-date
GoPubMed must know exactly the same documents as PubMed. PubMed adds 2000-5000 new citations per day.

QR02: Millions of documents
PubMed currently holds 18 million documents. GoPubMed needs to handle the same magnitude.

QR03: Answer Time
The answer time should be comparable with PubMed’s answer time.

QR04: Hundreds of users
The website must be accessible by hundreds of users at the same time.

QR05: Thousands of ontology concepts
The used ontologies contain more than 60,000 concepts. The number of concepts per ontology and the number of used ontologies is increasing. The system should be able to handle 100,000 and more concepts. This includes indexing as well as graphical presentation in a web-browser.

QR06: Completeness of results
It is important for previous PubMed users to see exactly the same documents for an identical search. This includes also the order of the retrieved documents.

QR07: Uptime
The search engine should be working with an uptime of close to 100%.

QR08: Thousands of results
20% of the searches result in result sets with thousands of documents. Such results need to be presented adequately without the need to reformulate the query string.

External Requirements
Requirements which arise from factors which are external to the system and its development process e.g. interoperability requirements, legislative requirements, etc.

ER01: License agreements
The license agreements with ontology providers, citation databases, third party open source providers need to be fully respected. This includes that the citations can not be changed or presented wrong, the ontologies can not be changed or presented wrong.

ER02: Security regulations
The public search engine is hosted by the TU Dresden. All network security regulations must be implemented.

ER03: Researchers access
The main code base in maintained by the spin-off company which works outside of the university. PhD students need to be able to implement prototype quickly and use the required hardware without endangering the stability of the production system.
5.2. REQUIREMENTS ANALYSIS

5.2.6 Acceptance

Acceptance testing generally involves running a suite of tests on the completed system. Each individual test, known as a case, exercises a particular operating condition of the user’s environment or feature of the system, and will result in a pass or fail boolean outcome.

It is difficult to define acceptance test in case of an free academic search engine as no contract is signed with a customer. However it is reasonable to assume that PubMed users are the first potential users of GoPubMed. Therefore one can derive the following success criteria:

S01: Content Any query to the system must return the same information as users of the original search engine would expect.

S02: Speed The response time must be comparable to PubMed (2-5 seconds).

S03: Benefit Ontology-based search must retrieve better results than classical search.

S04: Stability The system must be available at all times.

S05: Support Ontology-based search is a new search paradigm. User support should leave no questions unanswered for more than one day.

5.2.7 Use case

Typical use cases are described in Doms et al. [65], Backofen et al. [16]. A researcher wants to know which enzymes are inhibited by levamisole. A keyword search for levamisole inhibitor produces well over 100 hits in PubMed. To find out about specific functions, the researcher has to go through all these papers. He/she is interested in the relevant enzymatic functions. From the first titles it is immediately evident that levamisole inhibits alkaline phosphatase. A less well-known fact is however still buried in the abstracts. The abstract The effect of levamisole on energy metabolism in Ehrlich ascites tumor cells in vitro with PMID 2947578 is ranked very low (position 89 on 7/2/2005) by PubMed. The abstract states that levamisole also inhibits phosphofructokinases. Most readers will miss this statement.

Even if the user would try to reduce the number of papers by filtering out the ones mentioning levamisole inhibitor (e.g. query PubMed for levamisole inhibitor NOT phosphatase), he or she would miss the less obvious hits like phosphofructokinase, if both terms occur in the same abstract. Thus, even advanced PubMed queries with Boolean logic cannot always properly structure the search results.

Figure 5.2 shows a small fraction of GO. The available formats for GO are OBO XML, RDF, flat file and a relational database. Any query language used for GoPubMed in the later version must be able to reason over one of those formats - preferably over the standard format RDF.

For keeping the articles index up to date it would be useful to have some mechanism for automatic indexing of new available articles in the PubMed database. With the fully built index of all PubMed articles it is then possible to ask for all research publications on “Small GTPases” (and related concepts) but excluding abstracts related to a specific small GTPase like “Rho”.

How do we know if we succeeded?

How is GoPubMed typically used?
CHAPTER 5. DESIGN OF GOPUBMED

▷ Biological process
  ▷ Cellular process
    ▷ cell communication
      ▷ signal transduction
        ▷ intracellular signaling cascade
          ▷ small GTPase mediated signal transduction
            ▷ Rac protein signal transduction □
            ▷ Ras protein signal transduction □
            ▷ regulation of small GTPase mediated signal transduction
              ▷ regulation of Rho protein signal transduction
                ▷ positive regulation of Rho protein signal transduction
                ▷ negative regulation of Rho protein signal transduction
              ▷ Rho protein signal transduction
                ▷ regulation of Rho protein signal transduction
                  ▷ positive regulation of Rho protein signal transduction
                  ▷ negative regulation of Rho protein signal transduction
                ▷ regulation of signal transduction
              ▷ regulation of small GTPase mediated signal transduction □
            ▷ regulation of cellular process
              ▷ regulation of signal transduction
                ▷ regulation of small GTPase mediated signal transduction
          ▷ Molecular function
            ▷ enzyme regulator activity
            ▷ GTPase regulator activity
              ▷ small GTPase regulatory/interacting protein activity
        ▷ Cellular component

Figure 5.2: The GO sub-hierarchy containing terms related to small GTPases. ▷ symbolizes an is_a relation and □ symbolizes a part_of relation. Nodes marked □ hide more children related to “small GTPase mediated signal transduction”. Note: this tree view is stripped down to the concepts of GO necessary to explain our example. The subtree related to regulation of Rho protein signal transduction present two times because this GO term has multiple parents. The relations in GO are graph-shaped, we show here a simplified hierarchical representation.

5.2.8 GUI Design

The Graphical User Interface of an ontology-based search engine is a particularly difficult design decision. Academics users are accustomed to clear and simple interfaces. The induced ontology tree is particularly difficult to integrate into a simple interface. The expanded tree can very quickly become tall when the user expands branches. If the user needs to scroll the induced tree the risk is that the user looses focus of the query form and the most relevant documents at the top of the list. On the other hand a frame-based design in which each part of the interface has its own canvas and is scrollable can lead to an uneconomic layout.

Figure 5.3 shows four different graphical layouts four a search engine with the basic parts: query input area, document view and induced ontology. Solid borders denote frame borders and dashed borders denote borders of DIV-tags in a dynamic HTML page. The vertical arrows denote scrollable panes and the vertical arrows denote resizable areas.

The frameless GUI design (d) in figure 5.3 is the most flexible variant. It allows drag and drop of items within the ontology as well as the input fields and documents. The input
Figure 5.3: Four experimental GUI design for GoPubMed. (a) This design was used in the first prototype deployed in 2004, see figure 5.7 for a screenshot. A problem is the used space in the header frame. (b) This design variant includes the input field in the document view. The disadvantage is that the input area moves when the document view is scrolled. (c) This layout is used to the current version of GoPubMed, see screenshot 5.13. Less space is unused in the header and the input area is always visible when documents are scrolled. The disadvantage is that the frame architecture makes communication between the parts of the GUI complex. Another problem is that the frame borders disallow to display information across the borders. e.g. drag and drop of information or and auto-completion box opening in the input frame can not expand across the frameborder. (d) A frameless layout. A drawback is that the ontology view moves when document are scrolled. This can be avoided with a more complex dynamic HTML interface. AJAX technology is required to update parts of the interface independently.
fields can provide an auto-completion box which temporarily overlays the document view and closes when typing is finished.

5.3 Architectural Design

The top-level architectural components of GoPubMed can be grouped into data and information handling components. Information handling components are sub-typed into Information Retrieval, Information Extraction and Information Presentation. Other categories of software components deal with persistence and communication.

What are the main components of the system?

**Document Sources.** Each origin of documents is represented by an implementation of a DocumentSource. This may include remote web-services such as the one of the NCBI serving PubMed documents but also local repositories of documents. Other implementations may include abstractions from Google and Yahoo web APIs or wrappers to local Lucene indexes. The main functionalities of DocumentStores are: fetching of singles or collections of documents. For remote stores a caching strategy may be required. A store may be searchable or not. The result of a search is a DocumentResultSet which denotes an iterator over a potentially very large collection of documents. DocumentSources are involved in data handling, persistence and communication.

**Document Annotators.** A core functionality of an ontology-based search engine is the markup of documents with background knowledge. DocumentAnnotators are pure in Information Extraction components. The only task is to establish semantic link between documents or parts of documents (e.g. attributes and substrings of attributes) and ontology concepts. Example of DocumentAnnotators implemented for GoPubMed are WikiTermAnnotator, MeSHHeadingAnnotator, GOTermAnnotator and ProteinGeneNameAnnotator

**Attribute Annotator.** A document is represented in the system by a list of attributes, such as title text, publication data or abstract. Different types of attributes require different markup strategies. A DocumentAnnotator uses several AttributeAnnotators to create semantic annotations.

**Extractor.** Extractors implement Information Extraction algorithms. They can be used by several AttributeAnnotators.

**Annotation Stores.** The number and types of annotators are not fixed. A stand-off annotation scheme allows adding new markups without the need to re-annotate the whole document corpus. AnnotationStores collect markup information produced by a DocumentAnnotator and stores them. The main purpose is to quickly recover the result of complex annotation process and to query the document corpus for markups.

**Ontology Manager.** OntologyManagers provide a unified interface providing functionality around ontologies. An Ontology Manager can be composed by sub-managers to merge ontologies.
5.3. ARCHITECTURAL DESIGN

**Searcher.** The searcher takes the query string from the HTTP response and applies the QueryParser to analyse the syntax of the query. The query keywords are send to the Searcher which uses for example a searchable DocumentSource to produce a SearchResultSet. In case the query contains an advanced search the AdvancedSearcher queries an AnnotationStore and retrieves an advanced result set. The Searcher combines the two results into one CachedDocumentResultSet which provides an iterator over all relevant documents.

**Inducer.** The Inducer component uses the OntologyManager to process a DocumentResultSet produces the Induced Ontology. It uses a TermFetcher which retrieves annotations from the AnnotationStore to associate documents with term annotations.

**Tree Generator.** The TreeGenerator takes the InductionResult and produces the “Induced Ontology” view shown on the left side of the webpage, see figure 5.3. It is implemented as an HTTP servlet and dynamically visualizes the ontology tree depending on what branches are opened by the user.

**Document Generator.** The DocumentGenerator implements another HTTP servlet and produces a paginated view of a iterator over a collection of documents. It uses annotation information to markup titles and abstract texts which semantic links by using an AnnotationFetcher.

Figure 5.4 shows the main components collaborating for the default use case in which a user searches with a keyword query. A DocumentSource is searched with a parsed query. The outcome is a DocumentResultSet which contains documents already annotated and being cached in the system. Another subset of documents was annotated previously but is currently not cached in the system. The stored annotations have to be fetched from the AnnotationStore. The result is an AnnotatedDocument. The third portion of the document ids in the DocumentResultSet is unknown and has to be annotated using AttributeAnnotators which use Extractors and are called by DocumentAnnotators. AttributeAnnotators use OntologyManagers to link annotations to ontology concepts in the background knowledge.

Figure 5.5 shows the static model of the presentation layer of GoPubMed. The graphical user interface in the users browser communicates with several HTTP servlets to display for example the document view and the tree view.

This section documented the top level architecture of GoPubMed. The software components can be grouped into categories such as Information Retrieval, Information Extraction and Information Presentation. Other component types include persistence and communication. For the implementation details please refer to the supplementary materials.
Figure 5.4: This sequence diagram shows the default use case in which a user searches with a keyword query. The result contains cached, previously annotated and new documents. The outcome is a DocumentResultSet which is an iterator over AnnotatedDocuments.
Figure 5.5: Static Model of the presentation layer of GoPubMed 3.0
CHAPTER 5. DESIGN OF GOPUBMED

5.4 Version history of GoPubMed

The prototype of the first ontology-based search engine went online in 2004. It was based on ideas originating from an academic prototype described in Delfs et al. [62]. Since then it was developed into the a fully functional search engine in the life sciences. GoPubMed is now an academic product serving 1,200 of queries per day. Figure 5.6 shows the major development steps of the project. Three major releases were done during the active phase of the European research project REWERSE (IST-2004-506779). After the successful release of the system under the URL www.gopubmed.org end of 2004, screenshot shown in figure 5.7, one year later the version 3.0, shown in figure 5.8, was released. The funded startup company took over the maintenance of the servers and contributed significantly to the performance and the stability of the system. The search engine deployed in the beginning of 2006 was well received by the blogging community. It included precomputed bibliometric statistics for all Gene Ontology concepts as shown in figure 5.9. The Thompson Impact Factor was used to weight the individual publications of an author, as shown in figure 5.10. The next release version 3.0.8, shown in figure 5.11, introduced important new features such as multiple ontology annotations, improved tf-idf ranking of the top categories, protein and gene name annotations (based on Hakenberg et al. [111]), Wikipedia links, proximity ranking of the documents and author collaboration networks, see figure 5.12. Several of the new features were introduced to fulfill the requests of Transinsight’s first commercial client, Unilever. In the end of 2007 another release introduced several community features into GoPubMed, see screenshot 5.13. The author profiles are now disambiguated based on citation data and document annotations found in abstracts using the algorithm named “Chinese Whispers” [23]4. This reduced the problem that PubMed only provides the last name and the initials. Editable author profiles are now an integral part of the search engine. This allows users and authors of the system to edit and maintain a researcher’s list of publications and affiliations.

5.4.1 Community curation effort

Ontology-based literature search relies on sophisticated text-mining. While there are intelligent techniques to reach quality close to that of humans, those techniques depend on good training data. For an ontology-based search engine it is very important to distinguish the meaning of ontology terms in free text. This is in some cases a difficult task which needs training data for the machine learning algorithms.

Since version 3.6 GoPubMed offers a curation mode weaved into the web interface, see figure 5.14. Users can register with a single click, no login procedure is required.

5.4.2 User acceptance

The acceptance of GoPubMed was assessed in small scale surveys in oral form and larger online surveys. Table 5.8 shows the survey questions given to users. Table 5.6 lists positive and negative feedback given by authors in an online survey.

PhD students stated that they do not use PubMed every day and find GoPubMed useful. Group leaders use PubMed on a daily basis. In general the users liked the graphical interface and found that it is sufficiently documented by the provided help pages. Internal statistics show that most users do not read beyond the first result page. Typically users do not click more than 5 times after a keyword search. Users stated that they find the highlighting of the identified concepts and keywords find useful.

4Implementation by Matthias Zschunke
Figure 5.6: Timeline of GoPubMed releases, foundation of spin-off company and first customers for the ontology-based search platform
CHAPTER 5. DESIGN OF GOPUBMED

Figure 5.7: The screenshot shows the search result for authors named “Pizauro”. The result shows 26 documents classified using the Gene Ontology. The concept soluble was selected and one abstract is displayed. Ten other Gene Ontology concepts were identified using the Local Term Alignment method developed in Doms [67].

Table 5.6: Feedback given in an online survey with PubMed authors.
5.4.3 Reuse of component frameworks

The components for the presentation layer are build around the document driven component framework Cocoon from the Apache Foundation. XML documents are first class citizens in this framework. Generation, transformation and serialization components can be configured and composed into pipelines with XML files.

A disadvantage of this architecture is that communication between components has to be serialized into XML streams. This creates additional overhead when components exchange messages. As a result some components implement multiple concerns. For example the annotation of documents and the markup of the XML presentation. Performance reasons have let to the decision to reduce the number of components implemented as with Cocoon pipeline components. In a future version the XML-based component framework is planed to be replaced completely. The advantage of Cocoon is that the documents view can be composed by different components all dealing with separate concerns. However the transformed documents had to be serialized to XML streams for every stage. This additional step and the relatively slow XSL transformation create the biggest overhead is the processing of a query.

Figure 5.5 shows a static component model. The \texttt{TreeGenerator} and \texttt{DocumentGenerator} are implemented as Cocoon components creating XML streams which are later transformed via XSL transformations. The \texttt{CachedSearcher} takes a query string as input and returns the document ids of the search result. This component also deals with advanced searches in which a searchable annotation stores is queried. The \texttt{CachedInducer} takes the document ids relevant for a search and produced the information needed to display the tree view. The component may call annotators in case new documents are contained in the search result. The \texttt{DocumentGenerator} is the entry point for a web request. At the end of the pipeline the HTML content visible in the document view frame is produced. The \texttt{TreeGenerator} produces the output for the ontology view in the left frame, compare with figure 5.3.

5.5 GoPubMed’s offsprings

The architecture of GoPubMed is component based and easy to configure. Document sources, ontologies and information processing components are configured with XML files. The component lifecycle is managed with the Apache Excalibur component framework. This makes it easy to setup new search engines with different types of documents.

\textbf{GoWeb}. GoWeb is a semantic search engine for the life sciences. It combines the simplicity of keyword based search with semantic information. GoWeb uses the Yahoo-BOSS based search service and analyzes the search result summaries from the search service. The semantic information in GoWeb is based on biomedical relevant ontological background knowledge and databases. It is extracted from the search result summaries. GoWeb identifies concepts from the Gene Ontology (GO) and Medical Subject Headings (MeSH). Also protein and gene names are identified.

The cooccurrences of concepts and keywords in the search summaries is the basis for the semantic filtering. The semantic filter of GoWeb helps to reduce your potentially long list
### PubMed

<table>
<thead>
<tr>
<th>Question</th>
<th>Answer</th>
</tr>
</thead>
<tbody>
<tr>
<td>How often do you use PubMed?</td>
<td>Daily</td>
</tr>
<tr>
<td>What are the last literature searches you did?</td>
<td>New articles in the field of pancreatic cancer research</td>
</tr>
<tr>
<td>What keywords did you use in your searches?</td>
<td>myNCBI version of pancrea* and cancer</td>
</tr>
<tr>
<td>How many papers do you normally follow up from a PubMed search?</td>
<td>5</td>
</tr>
<tr>
<td>For how many specific proteins do you follow literature?</td>
<td>10</td>
</tr>
</tbody>
</table>

### GoPubMed

<table>
<thead>
<tr>
<th>Question</th>
<th>Answer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is the web interface usable and clear?</td>
<td>Yes</td>
</tr>
<tr>
<td>Did you like/dislike the user interface?</td>
<td>like</td>
</tr>
<tr>
<td>Is the structure of the page clear (tree, abstract window, search fields)?</td>
<td>yes</td>
</tr>
<tr>
<td>Is there enough help information and examples?</td>
<td>yes</td>
</tr>
</tbody>
</table>

### Functionality

<table>
<thead>
<tr>
<th>Question</th>
<th>Answer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Did you find the highlighting of ontology terms helpful?</td>
<td>very</td>
</tr>
<tr>
<td>Did you get an overview over your search results from the tree on the left?</td>
<td>yes</td>
</tr>
<tr>
<td>Did you manage to navigate efficiently through the tree?</td>
<td>yes</td>
</tr>
<tr>
<td>Did you find any papers you would probably have missed with PubMed?</td>
<td>yes</td>
</tr>
<tr>
<td>What do you like/dislike about using the tree to explore your search results?</td>
<td>Speed of server could be improved</td>
</tr>
</tbody>
</table>

### Wishes

<table>
<thead>
<tr>
<th>Question</th>
<th>Answer</th>
</tr>
</thead>
<tbody>
<tr>
<td>What functionality is missing in GoPubMed?</td>
<td>Direct link to full text, saved searches</td>
</tr>
<tr>
<td>Would you like GoPubMed to find protein names and hyperlinks which lead to further sequence/structure information for the proteins?</td>
<td>Yes</td>
</tr>
<tr>
<td>Would you like GoPubMed to use other ontologies such as Mesh?</td>
<td>Yes, KEGG would be fine</td>
</tr>
<tr>
<td>Would you like GoPubMed for the Web (GoGoogle)?</td>
<td>Interesting idea perhaps in combination with Google scholar</td>
</tr>
<tr>
<td>Would you like GoPubMed to be able to compile all relevant literature for a set of possibly 100 proteins you are working with? This compilation would show all relationships and common features of the proteins as documented in the literature.</td>
<td>This would be great</td>
</tr>
</tbody>
</table>

Table 5.8: User survey about GoPubMed V1.0 in May 2005. Answers from biomedical researcher and user of GoPubMed.
5.5. GOPUBMED’S OFFSPRINGS

Figure 5.8: The second release of GoPubMed (internal version 3.0) beginning of 2006 included the restructuring of the architecture which provided the basis for the next two versions of the software. The Graphical User Interface was re-designed allowing result sets of size 500-1000 instead of 100 in the first version, static bibliometric pages for single ontology terms, highlighting of semantic links and keywords in the abstract texts and titles. The position of a citation in the original PubMed query was shown beside links to HubMed and Google Scholar. The concept of informative concepts was introduced as Frequent term for the current query. Other improvements included export functions such as BibTex, EndNote and FullText as well as searching in the ontology tree.


Figure 5.9: This figure shows the static bibliometric statistics provided by GoPubMed 3.0. The example shows the plotted research interest in ARACHIDONIC ACID METABOLISM. The upper plot shows the absolute numbers of citations mentioning the term or a descendant concept. The lower plot shows the relative research interest compared to all other topics covered by PubMed citation in the same year in percent. The red line shows the moving average over 5 years.
5.5. GOPUBMED'S OFFSPRINGS

Figure 5.10: In GoPubMed 3.0 the Thompson Impact Factor was used to weight the individual publications of an author. The dynamic tables allowed to sort the rows according to the absolute number of articles of an author, the average impact factor of an authors publications, the maximum or minimum impact factor of an authors publication. One problem approached in a later version of GoPubMed were ambiguous author names.

<table>
<thead>
<tr>
<th>Author</th>
<th>Impact Factor (min)</th>
<th>Impact Factor (max)</th>
<th>Impact Factor (avg)</th>
<th>Number of Publications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yamamoto H</td>
<td>0.01</td>
<td>12.72</td>
<td>3.19</td>
<td>12</td>
</tr>
<tr>
<td>Cases S</td>
<td>0.01</td>
<td>3.03</td>
<td>0.59</td>
<td>13</td>
</tr>
<tr>
<td>Petticy JM</td>
<td>0.01</td>
<td>29.16</td>
<td>6.16</td>
<td>13</td>
</tr>
<tr>
<td>Zhu Z</td>
<td>0.67</td>
<td>2.35</td>
<td>1.43</td>
<td>5</td>
</tr>
<tr>
<td>Fukushima S</td>
<td>0.01</td>
<td>6.65</td>
<td>2.78</td>
<td>5</td>
</tr>
<tr>
<td>Murase K</td>
<td>1.94</td>
<td>6.79</td>
<td>3.21</td>
<td>4</td>
</tr>
<tr>
<td>Shonka AA</td>
<td>0.01</td>
<td>4.02</td>
<td>1.02</td>
<td>2</td>
</tr>
<tr>
<td>Yeung KJ</td>
<td>2.04</td>
<td>2.34</td>
<td>2.14</td>
<td>2</td>
</tr>
<tr>
<td>Evolve S</td>
<td>0.01</td>
<td>0.01</td>
<td>0.01</td>
<td>1</td>
</tr>
<tr>
<td>Ebi M</td>
<td>0.01</td>
<td>0.01</td>
<td>0.01</td>
<td>1</td>
</tr>
<tr>
<td>Murphy GA</td>
<td>2.85</td>
<td>2.99</td>
<td>2.92</td>
<td>1</td>
</tr>
<tr>
<td>Red KR</td>
<td>3.33</td>
<td>4.66</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>Macey DM</td>
<td>0.01</td>
<td>0.01</td>
<td>0.01</td>
<td>1</td>
</tr>
<tr>
<td>Primo A</td>
<td>0.01</td>
<td>0.01</td>
<td>0.01</td>
<td>1</td>
</tr>
<tr>
<td>Fall A</td>
<td>0.01</td>
<td>0.01</td>
<td>0.01</td>
<td>1</td>
</tr>
<tr>
<td>Know J</td>
<td>0.01</td>
<td>0.01</td>
<td>0.01</td>
<td>1</td>
</tr>
<tr>
<td>Urbansky RT</td>
<td>0.01</td>
<td>0.01</td>
<td>0.01</td>
<td>1</td>
</tr>
<tr>
<td>Yuan B</td>
<td>0.01</td>
<td>0.01</td>
<td>0.01</td>
<td>1</td>
</tr>
<tr>
<td>Fukushima Y</td>
<td>0.01</td>
<td>0.01</td>
<td>0.01</td>
<td>1</td>
</tr>
<tr>
<td>Kennedy G</td>
<td>0.01</td>
<td>0.01</td>
<td>0.01</td>
<td>1</td>
</tr>
</tbody>
</table>

Authors mentioning the GO Term most frequently in their abstracts.
CHAPTER 5. DESIGN OF GOPUBMED

Figure 5.11: Beginning of 2007 a release flagged with the internal version 3.0.8 included the following improvements: The list of Top Categories was computed based on the $tf-idf$ ranking described in section 4.2.5, AJAX technologies were introduced to allow for dynamical loading of the ontology tree, beside Gene Ontology term now also Wikipedia categories and MeSH headings were annotated, filtering was now possible with MeSH and GO concepts. Citations in the document view were now re-ranked according to the proximity of keywords and recognized ontology concepts and a dynamic bibliometric analysis for the current query was introduced. This screenshot shows GoPubMed 3.0.8 for a search with the keyword "aspirin".
Figure 5.12: In GoPubMed 3.0.8 the static bibliometric statistics for each concept were extended with an author collaboration network based on the top 50 authors. The example shown here displays the result for the query "gene ontology".
CHAPTER 5. DESIGN OF GOPUBMED

Figure 5.13: In December 2007 a press release in 22 languages was published to announce the new release 3.6 of GoPubMed. Interesting new features included Social Networking Features for biomedical experts, the ability to query up to 10,000 documents, the grouping of the organized information into four answer sections. The What section contained the induced ontology. The Who section contained filters for the disambiguated authors identified in the citations. The Where section allowed for filtering the documents according to geographic parameters and journals. The When section allowed filtering on the temporal basis. An eye-catching new feature showed the author profile of the most prominent author for the current query. The profile in this example shows that the author shown here is an internationally leading author in *Heliobacter pylori*, Gastritis and other topics. Form the publications linked to the author profile the system estimated that the author was at least from 1987 to 2007 affiliated with the Digestive Disease Section of the Veterans Administration Medical Center in Houston, Texas. Authors were able to edit their profiles online. Until today 978 authors edited their profiles.
5.5. GOPUBMED’S OFFSPRINGS

Figure 5.14: GoPubMed 3.6 introduced the Author Curation Tool which allowed users to give feedback on the automatic annotation quality of GoPubMed. Data collected with this tool was used to evaluate the recognition performance of the system and to train disambiguation models to improve the precision of the semantic markups.

from the keyword search to a short list of relevant results.

Go3R. Go3R is a knowledge-based search engine aimed at supporting researchers and companies to lower the suffering of animals by “refining” experimental methods, “reduce” the number of animal trials and “replace” whole animal experiments by alternative methods.

The core of any strategy to reduce animal experiments lies in the availability of relevant information regarding alternative methods. EU Directive 86/609/EEC for the protection of laboratory animals obliges scientists to consider whether a planned animal experiment can be substituted for by another scientifically satisfactory method which is reasonably and practicably available. To meet this regulatory obligation, scientists must consult the relevant scientific literature prior to any experimental study using laboratory animals. The internet enables access to a huge quantity of information. Nevertheless, it is difficult and time-consuming to select adequate information from this vast amount. Moreover, at the end of a query it remains unclear if all the required relevant information actually has been retrieved. This is where a new generation of knowledge-based search technology take effect. In April 2008, the beta version of Go3R (www.Go3R.org), the first knowledge-based search engine for alternative methods in agreement with the 3Rs principle, was released. Go3R is free of charge and enables scientists and regulatory authorities involved in the planning, authorisation and performance of animal experiments to determine the availability of alternative methods in a fast, comprehensive and transparent manner. The technical basis of this search engine is a specific expert knowledge, captured within an ontology. An ontology is a network of - also hierarchically - grouped “concepts” like subject areas, indicative for the respective field of research. It specifies the unambiguous meaning of relevant terms and depicts the complex relationships existing between them. With the help of such an ontology, the content of any document can be semantically determined by the mapping of the unique pattern of concepts and terms utilised in it. An essential step in the development of Go3R has involved the creation of an appropriate ontology by defining those concepts and terms that are relevant for alternative methods in accordance with the 3Rs principle and inferring the unique relations between them. The engine can now assist searchers by pre-sorting the retrieved documents according to their respective pattern of concepts and by attributing them to delimited topics. The result is an “intelligent table of contents” representing the community with the goal to prevent animal suffering.
hit list of relevant concepts and terms used in the documents, which the searcher can then use to navigate through the “thicket of information” of his query result.

**GoGene.** GoGene uses ontologies to organize gene product descriptions. GoGene applies different data sources to find annotations (GO and MeSH terms) for genes. First of all, known annotations of gene products are taken from UniProt. These are complemented by also searching for terms in EntrezGene’s summaries and GeneRIF text snippets and by looking for significant co-occurrences between genes and terms in PubMed abstracts.

GoGene takes a user query to search for genes. Found genes are sorted by concepts of the Gene Ontology (GO) and Medical Subject Headings (MeSH). These concepts, shown under the category What to the left, summarize the search result according to gene related functions, processes, diseases etc. Furthermore, the hierarchy of concepts can be used to semantically explore the result, e.g., after clicking on the concept apoptosis, only genes related to programmed cell death or its sub-concepts are listed. The protein/gene name recognition algorithm use in GoGene won the BioCreAtIvE 2 challenge [181].

GoGene also features an advanced search form that assists to formulate more complex queries to the literature database PubMed and to EntrezGene, a database of nearly 4 million genes in more than 5000 species.

**MousePubMed.** MousePubMed [39] uses ontology-based literature search for developmental biology. MousePubMed was build using vocabularies for mouse anatomy (EMAP), human anatomy (EHDA), mouse genes (from EMAGE), and mouse developmental stages (Theiler) as resources. 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 terms such as “node” or “skin”. Considering only the 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 term label. For instance, there exist 171 individuals with label “epithelium”. These all refer to different body parts at a specific stage in development.

### 5.6 Summary and Discussion

Finding the relevant literature is a difficult and essential task for researchers. The amount of literature is growing fast and classic search portals have limitations. They do not provide answers spread over multiple documents. Users have to deals with boolean search syntax and iteratively refine their query string. Large results are poorly represented by long result lists. Users rarely look at the second page of results. Keyword search does not included synonyms and sub-concepts. Meta-information in documents is not first-class citizen in keyword search.

Three search paradigm can be identified. Beside normal keyword search engines some systems make heavy use of natural language processing. Such systems index only limited document bases and suffer from imperfect algorithms. Another limitation is that users are required to formulate precise questions on the on hand and simple to parse questions on
the other hand. It is often difficult to ask the right question. Natural language questioning does not support serendipity search. Similarity based document clustering does not link documents to knowledge. The resulting hierarchies are inconsistent.

Ontology-based search sits in between the two paradigms and complements them. The entry point is a simple keyword search. Available answers reside in the hierarchical outline of the search results. Ontological background knowledge arranges a consistent exploration graph. Ontology-based search guides serendipity search with expert domain knowledge.

A state-of-the-art search engines for hundreds of users, thousands of terms and millions of documents is not a one man project. GoPubMed has now 371KLOC. Extensive project planning is a critical desideratum. The stakeholders are manifold and have opposing requirements. The biomedical publishing domain is pioneering knowledge based search but also add constraints to innovative systems, e.g. full text access would greatly improve value of services.

The goal was to develop a state-of-the-art search engine which has the potential to change the way biomedical researchers search for scientific literature and to showcase the potential of ontology-based search. GoPubMed has succeeded in both respects. The search engine is growing a user community and serves thousands of queries day by day. The usage numbers are increasing and the idea is very well received in the blogging community. GoPubMed's architecture proved to be easily reused for several successful commercial and academic offsprings.

Ontology-based search is a new paradigm complementary to classical keyword search. It remains to be seen how habits of Google and PubMed users chime together with outlined search. New users need to learn how to make use of the provided background knowledge. Two tutorials [68, 66] were published recently and blogs comment on the use of GoPubMed.

A prerequisite for the success of knowledge-based search are freely available high quality ontologies. The biomedical domain once more acts as a pioneering community. A number of projects promotes the development of structured vocabularies and embraces web standards.

Constructive criticism points out that automatic concept recognition fails on abstracts. The identified concepts are just too general. While this is true for many abstract texts using rather general language its less of a problem for technical papers. In the former case full texts would certainly increase the value of automatic methods to recognize ontology concepts. In the latter case irrelevant technical vocabulary from background sections needs to be filtered out, an additional task which is not required when dealing with abstracts only. Another major challenge is the graphical user interface. An intuitive navigation is a problem far from trivial. Additional research on GUI design and condensing information further is urgently necessary.

Summary of contributions. To summarize the contributions made in this chapter:

- Requirement Analysis: This chapter describes general requirements for an ontology-based search engine and specific requirements in the biomedical publishing domain. The stakeholders of the academic search engine GoPubMed and the application domain are characterized. The goals for the implementation of GoPubMed are defined and functional and quality requirements as well as external requirements from different viewpoints are defined. Criteria for the acceptance were defined and compared with the outcome of acceptance tests.

- Design: The top-level architectural design for GoPubMed is documented. For the default use case the collaboration of the main components of the system is documented by static and dynamic models. Four different GUI layout used for the search engine are discussed and a suggestion for a future layout is made.
Implementation: The main implementation contribution during this work were made for the release versions 2.0 and 3.0. The version 2.0 was the basis for the application for seed funding of the spin-off Transinsight, which as the goal to commercialize the ontology-based search technology. Version 3.0 served as a technology platform for various other scientific prototypes implemented for the European Research project REWERSE including GoProtein, GoPatents and LMOPubMed.

The current version of GoPubMed fulfills all functional requirements except FR17, FR21, FR23, FR26, FR28, FR34, FR35 and FR36 currently under development. FR17: The SCImago index will be used for weighting publication quality. FR21: Hot Deploy requires an architectural change. FR23: Future versions of GoPubMed will be able to send tasks to a job system. FR26: The system is monitoring usage of the system. Currently there are no rules implemented to react on misuse of the system. FR28: Editing of ontologies currently not part of the base system but implemented in the form of an external ontology editor. FR34: The generation of new ontologies from a text corpus was implemented in another work and currently exists as an external web-service. FR35: Ontology versioning is currently implemented as a full backup of the entire ontology. FR36: For ontology merging external tools need to be used.

The quality requirements are all met by the current system. QR03: the answer time is longer when the system is under heavy load. This mostly happens when a large amount of documents have to be (re-)annotated. All external requirement are fulfilled by the current system.

Offsprings of GoPubMed. The technology of the ontology-based search platform GoPubMed is successfully used in several commercial and academic products:

- LMOPubMed is developed for Unilever and is used in-house with an internally developed ontology. The search engine is based on GoPubMed’s technology.

- Go3R is a new search engine developed in cooperation with the Federal Institute for Risk Assessment. The goal is to support research on alternative methods for animal experiments to fulfill the EU directive 86/609/EEC.

- MousePubMed is also based on GoPubMed technology and algorithms. It shows successfully that other domain ontologies can be used for ontology-based search.

Go3R, GoWeb and GoGene are further developed in their own research projects.

Conclusions. GoPubMed is the first ontology-based search engine in the life sciences. The web-based search engine is continuously online at www.gopubmed.org since 2004. Users searching with GoPubMed retrieve exactly the same documents as when they search for PubMed. This ensures the quality of the content. Despite the fact that GoPubMed displays an enormous amount of information the response time is still very good. Users of the systems report a great benefit when using GoPubMed. Several independent user blogs started threads to teach how to use GoPubMed and name it a very useful tool for literature research.

Stability and support are concerns taken over by a spin-off company Transinsight, which licensed GoPubMed technology and was additionally funded by the High Tech Gründerfond. GoPubMed’s technology was already transferred to industry products. Unilever and the Federal Institute for Risk Assessment are developing their own ontologies and use it with GoPubMed search technology.
Chapter 6

Conclusion and future work

This chapter highlights the solutions to the open problems listed in chapter 1. It emphasizes on the scientific contributions to Semantic Search using biomedical ontologies.

6.1 General contributions

This work includes a complete survey on state-of-the-art biomedical search engines. The systems were categorized according to their features supporting Information and Knowledge Retrieval. One conclusion is that some systems semantically process documents but no system uses ontologies to organize search result.

Another contribution is a survey over 15 freely available annotation corpora. There is no corpus directly applicable for the evaluation of Concept Recognition algorithms for Gene Ontology terms. The BioCreAtIvE dataset is the best source for manually confirmed Gene Ontology annotations in full texts. However it is biased toward the annotation of selected proteins.

6.2 Contributions to open problem 1

Open problem 1: Scalable algorithms for ontological concept recognition

The goal is to design an efficient method to annotate free PubMed abstracts with concept labels of the Gene Ontology. Out of a pool of several thousands ontological concepts the system has to detect candidates in a way that tolerates potentially new syntactical variants of known concepts aiming at a high recall. The meanings of ambiguous concept labels have to be distinguished aiming at a high precision. The algorithms have to scale for millions of documents and thousands of concepts.

Chapter 3 describes a new Concept Recognition pipeline which improved the previously reported [67] performance by 25.7% achieving 79.9% precision and 72.7% recall. The result show that a hybrid approach using patterns and information content for candidate identification and a trained disambiguation module with the best systems participating in the BioCreAtIvE workshop [148]. The authors of one of the best approaches based only on patterns and information content [57] argue that their system can have close to 60% precision for the recognition of Gene Ontology concepts. With additional hints about relevant concepts their system achieves up to 93% precision [58]. Another system [50] achieved 80%
precision in the BioCreAtIvE workshop. However the methods recall was very low compared to the other system.

The results were achieved by designing and implementing a set of tools which include: (1) A pre-processing modules for ontology terminology. The result are recognition patterns which are used as rules to recognize concept candidates in biomedical texts. (2) A recognition engine using the patterns. It enables the simultaneous detection of several ten-thousands of concept on one machine. (3) A disambiguation module using Maximum Entropy context models. It was shown that very high recall (97.5%) and precision (91%) values can be achieved with high quality training data.

To access the quality of the pipeline and to facilitate further training for ambiguous ontology labels a curations tool was designed. With this tools a new benchmark comprised of 689 PubMed abstracts and 18,356 curations, personally curated by the original authors of PubMed articles, was created.

The developed Concept Recognition could be used to align ontologies as described in Lambrix et al. [155].

6.3 Contributions to open problem 2

Open problem 2: Methodes for knowledge mining in annotated literature databases
The goal is to develop reasoning algorithms for mining information from text based on semantic markups in literature abstracts. In contrast to classical searches the aim is to provide a table of content outlining the whole result set and providing links to subsets of documents grouped by topics. The goal is also to link bibliometric analyses to ontology-based literature search to discover trends, prominent authors and research institutions as well as important journals for single research topics.

Chapter 4 describes algorithms for a new search paradigm using the background knowledge of ontologies to search for relevant documents and answer biomedical questions. First, semantic links in a large text corpus with more than 18,000,000 citations were established. More than 412 million semantic markups of the Gene Ontology concepts and MeSH headings in PubMed abstracts were created. Second, implicit knowledge was recovered by computing a graph of semantically linked ontology concepts. The resulting graph named Induced Ontology, is potentially very large. Two criteria were developed to identify informative paths and nodes in this graph. This allows to identify the most relevant concepts which best characterize a potentially large document collection. Third, it is shown that questions posed by real biologists in the TREC Genomics Track 2006 can be answered with the Induced Ontology and the background knowledge of the ontologies. Finally, the first ontology-based, large scale, online available, up-to-date bibliometric analysis for topics in molecular biology represented by Gene Ontology concepts is evaluated. It is shown that the method is in line with existing, but often out-dated, analyses.
6.4 Contributions to open problem 3

Open problem 3: Design of a system for ontology-based literature search for the life sciences
The goal is to design a system that supports the answering of biomedical questions by processing search results of PubMed using the background knowledge of biomedical ontologies. The system must be capable of serving hundreds of users, ten thousands of concepts and millions of documents while responding immediately. A community curation tool is to be designed which allows user driven annotation.

Chapter 5 uses the results of the previous chapters and describes the design and implementation of the first ontology-based search engine for the life sciences. The web based search engine is online at www.gopubmed.org since 2004 and serves thousands of users every day. An article describing GoPubMed is cited $\geq$87 times in Google Scholar. Section 2.4 discusses 23 related systems. There is currently no other system using ontologies for literature search. Implementation work for this thesis was mainly done for versions 2.0 and 3.0 of the system. These academic prototypes have lead to the foundation of a spin-off company of the TU Dresden aiming at commercializing the ontology-based search technology. The company develops search engines based on GoPubMed’s technology for several customers including Unilever, BASF and the Federal Institute for Risk Assessment in Germany. Some of the offsprings of GoPubMed are GoWeb, an ontology-based search engine for the web, Go3R, a search engine for alternatives to animal testing and GoGene, a search engine for genes and proteins.

6.5 Future research

A number of academic and research projects have arisen from GoPubMed.

6.5.1 Ontology generation

The methods describes in this work are not limited to the biomedical domain. Currently it is investigated how new ontologies can be generated by extending available domain ontologies or from the scratch. Ontology design is a labor intensive task. It is of great interest in how available literature corpora can be used to generate suggestions for ontology concept labels, synonyms, abbreviations, textual definitions and relations between concepts.

6.5.2 Question answering

The Induced Ontologies are a rich source of information. However Concept Recognition methods can not be perfect. Even human experts disagree on a large number of concept mentions in natural language texts. Therefore the Induced Ontology will always carry irrelevant information and lack of some relevant information hidden in the texts. Recurrently stated knowledge is well represented in the Induced Ontology. It is of great interest how to identify under-represented or new knowledge in document collections. The difficulty here is to distinguish this knowledge from noise in the form of irrelevant markups.
6.5.3 GoEverywhere

An open question is how to build the next generation web search engine using ontologies and textmining. There are numerous problems including: the task to annotate millions of documents per day with an ever growing number of concepts defined in ontologies, the task of efficiently compute the Induced Ontology for millions of result documents and the task of presenting the results in an intuitive way.
Bibliography


Appendix

Background

<table>
<thead>
<tr>
<th>Organism</th>
<th>Complete</th>
<th>Draft assembly</th>
<th>In progress</th>
<th>total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prokaryotes</td>
<td>640</td>
<td>490</td>
<td>478</td>
<td>1608</td>
</tr>
<tr>
<td>Archaea</td>
<td>50</td>
<td>4</td>
<td>29</td>
<td>83</td>
</tr>
<tr>
<td>Bacteria</td>
<td>590</td>
<td>486</td>
<td>449</td>
<td>1525</td>
</tr>
<tr>
<td>Eukaryotes</td>
<td>22</td>
<td>135</td>
<td>177</td>
<td>334</td>
</tr>
<tr>
<td>Animals</td>
<td>4</td>
<td>56</td>
<td>83</td>
<td>143</td>
</tr>
<tr>
<td>Mammals</td>
<td>2</td>
<td>20</td>
<td>26</td>
<td>48</td>
</tr>
<tr>
<td>Birds</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Fishes</td>
<td>3</td>
<td>6</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>Insects</td>
<td>1</td>
<td>20</td>
<td>16</td>
<td>37</td>
</tr>
<tr>
<td>Flatworms</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Roundworms</td>
<td>1</td>
<td>4</td>
<td>13</td>
<td>18</td>
</tr>
<tr>
<td>Amphibians</td>
<td>2</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reptiles</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Other animals</td>
<td>8</td>
<td>17</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>Plants</td>
<td>2</td>
<td>7</td>
<td>31</td>
<td>40</td>
</tr>
<tr>
<td>Land plants</td>
<td>2</td>
<td>5</td>
<td>24</td>
<td>31</td>
</tr>
<tr>
<td>Green Algae</td>
<td>2</td>
<td>7</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>Fungi</td>
<td>10</td>
<td>52</td>
<td>32</td>
<td>94</td>
</tr>
<tr>
<td>Ascomycetes</td>
<td>8</td>
<td>43</td>
<td>23</td>
<td>74</td>
</tr>
<tr>
<td>Basidiomycetes</td>
<td>1</td>
<td>7</td>
<td>4</td>
<td>12</td>
</tr>
<tr>
<td>Other fungi</td>
<td>1</td>
<td>2</td>
<td>5</td>
<td>8</td>
</tr>
<tr>
<td>Protists</td>
<td>6</td>
<td>18</td>
<td>27</td>
<td>51</td>
</tr>
<tr>
<td>Apicomplexans</td>
<td>1</td>
<td>9</td>
<td>7</td>
<td>17</td>
</tr>
<tr>
<td>Kinetoplasts</td>
<td>1</td>
<td>2</td>
<td>5</td>
<td>8</td>
</tr>
<tr>
<td>Other protists</td>
<td>4</td>
<td>7</td>
<td>14</td>
<td>25</td>
</tr>
</tbody>
</table>

|              | 662      | 625            | 655         | 1942  |

Table 6.1: Genome sequencing projects statistics, from NCBI’s website
<table>
<thead>
<tr>
<th>MeSH Heading</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANATOMY</td>
</tr>
<tr>
<td>ORGANISMS</td>
</tr>
<tr>
<td>DISEASES</td>
</tr>
<tr>
<td>CHEMICALS AND DRUGS</td>
</tr>
<tr>
<td>ANALYTICAL, DIAGNOSTIC AND THERAPEUTIC TECHNIQUES AND EQUIPMENT</td>
</tr>
<tr>
<td>PSYCHIATRY AND PSYCHOLOGY</td>
</tr>
<tr>
<td>BIOLOGICAL SCIENCES</td>
</tr>
<tr>
<td>NATURAL SCIENCES</td>
</tr>
<tr>
<td>ANTHROPOLOGY, EDUCATION, SOCIOLOGY AND SOCIAL PHENOMENA</td>
</tr>
<tr>
<td>TECHNOLOGY, INDUSTRY, AGRICULTURE</td>
</tr>
<tr>
<td>HUMANITIES</td>
</tr>
<tr>
<td>INFORMATION SCIENCE</td>
</tr>
<tr>
<td>NAMED GROUPS</td>
</tr>
<tr>
<td>HEALTH CARE</td>
</tr>
<tr>
<td>PUBLICATION CHARACTERISTICS</td>
</tr>
<tr>
<td>GEOGRAPHICALS</td>
</tr>
</tbody>
</table>

Table 6.2: First level of MeSH Headings
<table>
<thead>
<tr>
<th>Title</th>
<th>Domain</th>
<th>Prefix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amphibian gross anatomy</td>
<td>anatomy</td>
<td>AAO</td>
</tr>
<tr>
<td>Biological process</td>
<td>biological process</td>
<td>GO</td>
</tr>
<tr>
<td>C. elegans development</td>
<td>anatomy</td>
<td>WBls</td>
</tr>
<tr>
<td>C. elegans gross anatomy</td>
<td>anatomy</td>
<td>WBbt</td>
</tr>
<tr>
<td>C. elegans phenotype</td>
<td>phenotype</td>
<td>WBPhenotype</td>
</tr>
<tr>
<td>Cell type</td>
<td>anatomy</td>
<td>CL</td>
</tr>
<tr>
<td>Cellular component</td>
<td>anatomy</td>
<td>GO</td>
</tr>
<tr>
<td>Cereal plant trait</td>
<td>phenotype</td>
<td>TO</td>
</tr>
<tr>
<td>Chemical entities of biological interest</td>
<td>biochemistry</td>
<td>CHEBI</td>
</tr>
<tr>
<td>Common Anatomy Reference Ontology</td>
<td>anatomy</td>
<td>CARO</td>
</tr>
<tr>
<td>Dictyostelium discoideum anatomy</td>
<td>anatomy</td>
<td>DDANAT</td>
</tr>
<tr>
<td>Drosophila development</td>
<td>anatomy</td>
<td>FBdv</td>
</tr>
<tr>
<td>Drosophila gross anatomy</td>
<td>anatomy</td>
<td>FBbt</td>
</tr>
<tr>
<td>Environment Ontology</td>
<td>environment</td>
<td>ENVO</td>
</tr>
<tr>
<td>Evidence codes</td>
<td>experiments</td>
<td>ECO</td>
</tr>
<tr>
<td>Fly taxonomy</td>
<td>taxonomy</td>
<td>FBsp</td>
</tr>
<tr>
<td>Foundational Model of Anatomy (subset)</td>
<td>anatomy</td>
<td>FMA</td>
</tr>
<tr>
<td>Fungal gross anatomy</td>
<td>anatomy</td>
<td>FAO</td>
</tr>
<tr>
<td>Human developmental anatomy &amp; abstract version</td>
<td>anatomy</td>
<td>EHDAA</td>
</tr>
<tr>
<td>Human developmental anatomy &amp; timed version</td>
<td>anatomy</td>
<td>EHDA</td>
</tr>
<tr>
<td>Human disease</td>
<td>health</td>
<td>DOID</td>
</tr>
<tr>
<td>Infectious disease</td>
<td>health</td>
<td>IDO</td>
</tr>
<tr>
<td>Mammalian phenotype</td>
<td>phenotype</td>
<td>MP</td>
</tr>
<tr>
<td>Medaka fish anatomy and development</td>
<td>anatomy</td>
<td>MFO</td>
</tr>
<tr>
<td>Molecular function</td>
<td>biological function</td>
<td>GO</td>
</tr>
<tr>
<td>Mosquito gross anatomy</td>
<td>anatomy</td>
<td>TGMA</td>
</tr>
<tr>
<td>Mosquito insecticide resistance</td>
<td>environment</td>
<td>MIBO</td>
</tr>
<tr>
<td>Mouse adult gross anatomy</td>
<td>anatomy</td>
<td>MA</td>
</tr>
<tr>
<td>Mouse gross anatomy and development</td>
<td>anatomy</td>
<td>EMAP</td>
</tr>
<tr>
<td>Mouse pathology</td>
<td>health</td>
<td>MPATH</td>
</tr>
<tr>
<td>OBO relationship types</td>
<td>all</td>
<td>OBO_REL</td>
</tr>
<tr>
<td>Ontology for biomedical investigations</td>
<td>experiments</td>
<td>OBI</td>
</tr>
<tr>
<td>Pathogen transmission</td>
<td>health</td>
<td>TRANS</td>
</tr>
<tr>
<td>Phenotypic quality</td>
<td>phenotype</td>
<td>PATO</td>
</tr>
<tr>
<td>Plant growth and developmental stage</td>
<td>anatomy</td>
<td>PO</td>
</tr>
<tr>
<td>Plant structure</td>
<td>anatomy</td>
<td>PO</td>
</tr>
<tr>
<td>Protein modification</td>
<td>proteins</td>
<td>MOD</td>
</tr>
<tr>
<td>protein ontology</td>
<td>proteins</td>
<td>PRO</td>
</tr>
<tr>
<td>Protein-protein interaction</td>
<td>experiments</td>
<td>MI</td>
</tr>
<tr>
<td>Sequence types and features</td>
<td>biological sequence</td>
<td>SO</td>
</tr>
<tr>
<td>Spatial Ontology</td>
<td>anatomy</td>
<td>BSPO</td>
</tr>
<tr>
<td>Spider Ontology</td>
<td>anatomy</td>
<td>SPD</td>
</tr>
<tr>
<td>Subcellular anatomy ontology</td>
<td>anatomy</td>
<td>SAO</td>
</tr>
<tr>
<td>Suggested Ontology for Pharmacogenomics</td>
<td>health</td>
<td>SOPHARM</td>
</tr>
<tr>
<td>Systems Biology</td>
<td>biochemistry</td>
<td>SBO</td>
</tr>
<tr>
<td>Teleost anatomy and development</td>
<td>anatomy</td>
<td>TAO</td>
</tr>
<tr>
<td>Teleost taxonomy</td>
<td>taxonomy</td>
<td>TTO</td>
</tr>
<tr>
<td>Tick gross anatomy</td>
<td>anatomy</td>
<td>TADS</td>
</tr>
<tr>
<td>Units of measurement</td>
<td>phenotype</td>
<td>UO</td>
</tr>
<tr>
<td>Xenopus anatomy and development</td>
<td>anatomy</td>
<td>XAO</td>
</tr>
<tr>
<td>Zebrafish anatomy and development</td>
<td>anatomy</td>
<td>ZFA</td>
</tr>
</tbody>
</table>

Table 6.3: OBO Foundry candidate ontologies
Table 6.4: MeSH Subheadings are used to better define a topic, narrow retrieval, or express a certain aspect of a main MeSH heading.

<table>
<thead>
<tr>
<th>Database</th>
<th>Content</th>
<th>GOA</th>
<th>Online Resource</th>
</tr>
</thead>
<tbody>
<tr>
<td>CGD</td>
<td>Candida Genome Database</td>
<td>1268</td>
<td><a href="http://www.candidagenome.org">www.candidagenome.org</a></td>
</tr>
<tr>
<td>SGD</td>
<td>Saccharomyces cerevisiae database</td>
<td>6342</td>
<td><a href="http://www.yeastgenome.org">www.yeastgenome.org</a></td>
</tr>
<tr>
<td>MGI</td>
<td>Mouse Genome Informatics</td>
<td>13234</td>
<td><a href="http://www.informatics.jax.org">www.informatics.jax.org</a></td>
</tr>
<tr>
<td>FlyBase</td>
<td>Drosophila database</td>
<td>10548</td>
<td>flybase.bio.indiana.edu</td>
</tr>
<tr>
<td>TAIR</td>
<td>Arabidopsis Information Resource</td>
<td>32524</td>
<td><a href="http://www.arabidopsis.org">www.arabidopsis.org</a></td>
</tr>
<tr>
<td>WormBase</td>
<td>Caenorhabditis elegans database</td>
<td>6270</td>
<td><a href="http://www.wormbase.org">www.wormbase.org</a></td>
</tr>
<tr>
<td>RGD</td>
<td>Rat Genome Database</td>
<td>11640</td>
<td>rgl.mcw.edu</td>
</tr>
<tr>
<td>Gramene</td>
<td>Rice/grasses genome</td>
<td>60961</td>
<td><a href="http://www.gramene.org">www.gramene.org</a></td>
</tr>
<tr>
<td>ZFIN</td>
<td>Zebrafish Model Organism Database</td>
<td>8491</td>
<td>zfin.org</td>
</tr>
<tr>
<td>GeneDB</td>
<td>prokaryotic and eukaryotic organisms</td>
<td>10416</td>
<td><a href="http://www.genedb.org">www.genedb.org</a></td>
</tr>
</tbody>
</table>

Table 6.5: Databases contributing gene product associations to the Gene Ontology Annotations project.
Entry for the Sub Heading "Blood" in the NLM Indexing Manual

19.8.10 /blood /BL, /blood Used for the presence or analysis of substances in the blood; also for examination of, or changes in, the blood in disease states. It excludes serodiagnosis, for which the subheading "diagnosis" is used, and serology, for which "immunology" is used. /blood should be used for in vivo or in vitro studies of substances in the blood, blood cells (including their metabolism), and physical properties of the blood (such as its viscosity, coagulability, etc.).

Blood serotonin levels in schizophrenia.
SEROTONIN / * blood
SCHIZOPHRENIA / * blood

Transportation of sodium in erythrocytes from pregnant women.
SODIUM / * blood
ERYTHROCYTES / * metab
PREGNANCY / * blood
BIOLOGICAL TRANSPORT / physiol

Blood viscosity in hypertension.
HYPERTENSION / * blood
* BLOOD VISCOSITY

Do not use /blood for studies on the hemodynamics of blood, such as its flow, circulation, pressure, etc. These do not usually result from the properties of the fluid itself, but the dynamics of the cardiovascular system which propels it, and thus should be indexed with the subheading /physiology, or /physiopathology in a disease.

Blood pressure in labor.
LABOR / * physiol
* BLOOD PRESSURE

Blood volume in migraine.
MIGRAINE / * physiopathol
* BLOOD VOLUME

The subheading /blood is not an AQ for Category D terms normally found in the blood, so /analysis or /metabolism (as appropriate) should be used instead.

Changes in blood fibrin levels in liver disease.
FIBRIN / * metab
LIVER DISEASES / * blood
(not FIBRIN / * blood)
BIBLIOGRAPHY
224

Established knowledge
Supervised
Unsupervised

[195]

[99]
[162, 163]
[97]

[115]

[78]

[111]

[128]

word cooc, PoS tags

gene symbol context
(n words +/-)
document

text
Medline abstracts
abbreviations in
Medline abstracts

XML tagged abstracts
positional info, PoS

Medline abstracts

Medline abstracts

Data
gene definition
& abstract vector
free text

–

WordNet

–

–

–
UMLS terms
–

–

average link clustering

LSA/LSI, 2nd order cooc

SVM

word count, word cooc
UMLS term cooc
SVM

Approach
cosine similarity

[239, 241]

–

1st ,2nd order context vectors
(coocs within 5 positions)
co-training, collocations

publ.
[237]

[198]
[199]
[211]
text

few tagged data,
WordNet
–
WordNet

Accuracy
92.7%

78.7%

97%P

81%

BioCreAtIvE GN challenge

Experiment
52,529 Medline abstracts
690 human gene symbols
45 ambiguous UMLS terms
(NLM WSD Collection)
BioCreAtIvE-2 GN challenge

inverse co-author graph

85%

96.5%

44%

73.4%

↑ 7-14%

98.5%

86.5%
93%P

85%

naive Bayes, decision trees,
inductive rule training

protein/gene/mRNA
assignment: 9 million words
(mol. biol. journals)
–
35 biomedical abbreviations
build dictionary, use for
abbreviations occurring
with their long forms
–

↑ 9.8%
–
co-training & majority voting
noun coocs,
Markov clustering

170,000 documents,
1013 terms (TREC-1)
(Wall Street Journal)
13 words, ACL/DCI
Wall Street Journal Corpus
24 Senseval-2 words,
Line, Hard, Serve corpora
12 common Engl. words
x 4000 instances
Senseval-2 generic English
–

Journal Descriptor
Indexing (JDI)
motifs from multiple
sequence alignments

[288]

–
–

Backgr. knowledge
5 human gen. dbs
& MeSH
UMLS, Journal
Descriptors
BioCreAtIvE-2 GN
lexicon & text,
EntrezGene,
UniProt, GOA
list of gene senses,
EntrezGene

[176]
[70]


Figure 6.1: A rule based segmentation algorithm was used to split texts into sentences. The input text is assumed to be tokenized before.
# TREC Genomics Track 2006 Questions

<table>
<thead>
<tr>
<th>Question #160: What is the role of PrnP in mad cow disease?</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Query:</strong> PrnP  <strong>Expanded:</strong> yes  <strong>Aspect:</strong> ENCEPHALOPATHY, BOVINE SPONGIFORM</td>
</tr>
<tr>
<td><strong>Results:</strong> 76  <strong>Clicks:</strong> 3  <strong>Position:</strong> 1</td>
</tr>
<tr>
<td><strong>GoPubMed Answer:</strong> Since 2004, significant associations between bovine spongiform encephalopathy (BSE) susceptibility in cattle and frequencies of insertion/deletion (indel) polymorphisms within the bovine prion protein gene (PRNP) have been reported. PMID: 18399944</td>
</tr>
<tr>
<td><strong>TREC Answer:</strong> A genetic form of prion disease can also be produced in mice harboring a PrP transgene with a point mutation PMID: 7592679</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Question #161: What is the role of IDE in Alzheimer's disease?</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Query:</strong> IDE  <strong>Expanded:</strong> no  <strong>Aspect:</strong> ALZHEIMER DISEASE</td>
</tr>
<tr>
<td><strong>Results:</strong> 57  <strong>Clicks:</strong> 1  <strong>Position:</strong> 1</td>
</tr>
<tr>
<td><strong>GoPubMed Answer:</strong> Insulin degrading enzyme (IDE) is one of the principal proteases involved in the degradation of the beta-amyloid peptide, which is the major constituent of senile plaques in Alzheimer’s disease (AD) brains. PMID: 16876916</td>
</tr>
<tr>
<td><strong>TREC Answer:</strong> Recently, IDE was implicated in the degradation of the Alzheimer’s disease beta-amyloid peptide (11, 12). Further studies have suggested that IDE plays a role in the control of extracellular levels of Alzheimer’s beta-amyloid peptide PMID: 10973971</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Question #162: What is the role of MMS2 in cancer?</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Query:</strong> MMS2  <strong>Expanded:</strong> no  <strong>Aspect:</strong> DNA REPAIR (GO)</td>
</tr>
<tr>
<td><strong>Results:</strong> 1  <strong>Clicks:</strong> 1  <strong>Position:</strong> 5</td>
</tr>
<tr>
<td><strong>GoPubMed Answer:</strong> Furthermore, increased MMS2 expression mediates the inhibitor of DNA binding 1 proteolysis and promotes DNA repair. PMID: 16908531</td>
</tr>
<tr>
<td><strong>TREC Answer:</strong> For Trench, the findings are added proof of the guilt of DNA repair pathway genes in hereditary breast cancer. PMID: 11830600</td>
</tr>
</tbody>
</table>
**Question #163:** What is the role of APC (adenomatous polyposis coli) in colon cancer?

**Query:** APC +mesh#"Colonic Neoplasms"  
**Expanded:** no  
**Aspect:** COLONIC NEOPLASMS (MeSH)  
**Results:** 208  
**Clicks:** 1  
**Position:** 7  

**GoPubMed Answer:** One of the critical downstream molecules regulated by APC is beta-catenin; molecular targeting of beta-catenin is, thus, an attractive chemopreventative strategy in colon cancer. PMID: 18521697

**TREC Answer:** The adenomatous polyposis coli (APC)1 gene is a tumor suppressor gene linked to familial adenomatous polyposis (FAP) and to the initiation of sporadic human colorectal cancer PMID: 9015311

**Question #164:** What is the role of Nurr-77 in Parkinson’s disease?

**Query:** Nurr-77 +mesh#"Parkinson Disease"  
**Expanded:** yes  
**Aspect:** PARKINSON DISEASE (MeSH)  
**Results:** 6  
**Clicks:** 3  
**Position:** 1  

**GoPubMed Answer:** The NR4A1-3 (Nur77, NURR1 and NOR-1) subfamily of nuclear hormone receptors (NRs) has been implicated in Parkinson’s disease, schizophrenia, manic depression, atherogenesis, Alzheimer’s disease, rheumatoid arthritis, cancer and apoptosis. PMID: 15956351

**TREC Answer:** They have been implicated in proliferation, differentiation, apoptosis, hypertrophy/remodeling, Parkinson’s disease, schizophrenia, manic depression, atherogenesis, cancer, and autoimmune disease (14-34). In itself, the NR4A1-3 subgroup PMID: 12709428

**Question #165:** How do Cathepsin D (CTSD) and apolipoprotein E (ApoE) interactions contribute to Alzheimer’s disease?

**Query:** CTSD AND ApoE +mesh#"Alzheimer Disease"  
**Expanded:** yes  
**Aspect:** ALZHEIMER DISEASE (MeSH)  
**Results:** 18  
**Clicks:** 2  
**Position:** 3  

**GoPubMed Answer:** Neuropathological changes in scrapie and Alzheimer’s disease are associated with increased expression of apolipoprotein E and cathepsin D in astrocytes. PMID: 1870200

**TREC Answer:** Neuropathological changes in scrapie and Alzheimer’s disease are associated with increased expression of apolipoprotein E and cathepsin D in astrocytes PMID: 9302273
Question #166: What is the role of Transforming growth factor-beta1 (TGF-beta1) in cerebral amyloid angiopathy (CAA)?

**Query:** TGF-beta1 + "mesh"/"Cerebral Amyloid Angiopathy"  
**Expanded:** no  
**Aspect:** Cerebral Amyloid Angiopathy (MeSH)  
**Results:** 3  
**Clicks:** 1  
**Position:** 1  

**GoPubMed Answer:** A recent study showed that transforming growth factor-beta1 (TGF-beta1) induces amyloid-beta deposition in cerebral blood vessels and meninges of a transgenic mouse model of Alzheimer’s disease (AD), and that TGF-beta1 mRNA levels are correlated with cerebral amyloid angiopathy (CAA) in human AD brains. PMID: 15834029

**TREC Answer:** Postmortem brain tissue analyses of AD patients show increased TGF-beta1 expression, which can be closely correlated with the degree of cerebral amyloid angiopathy (CAA) (7), a major pathological feature of AD and related disorders. PMID: 15632190

---

Question #167: How does nucleoside diphosphate kinase (NM23) contribute to tumor progression?

**Query:** nucleoside diphosphate kinase + "mesh"/"Neoplasms"  
**Expanded:** yes  
**Aspect:** Neoplasm Metastasis (MeSH)  
**Results:** 148  
**Clicks:** 3  
**Position:** 1  

**GoPubMed Answer:** Nucleoside diphosphate kinase A (NDPK-A), encoded by the nm23-H1 gene, acts as a metastasis suppressor in certain human tumors such as breast carcinoma. PMID: 15280446

**TREC Answer:** Another tumor suppressor gene that may be associated with pituitary tumorigenesis is the purine-binding factor gene, nm23. In highly metastatic cancer, including breast, hepatic, and colorectal carcinomas, nm23 expression is reduced (64). PMID: 9177361

---

Question #168: How does BARD1 regulate BRCA1 activity?

**Query:** BARD1 AND BRCA1 + "go"/"biological process"  
**Expanded:** yes  
**Aspect:** REGULATION OF PROGRESSION THROUGH CELL CYCLE (GO)  
**Results:** 67  
**Clicks:** 3  
**Position:** 1  

**GoPubMed Answer:** The BRCA1 tumor suppressor exists as a heterodimeric complex with BARD1, and this complex is thought to mediate many of the functions ascribed to BRCA1, including its role in tumor suppression. PMID: 17848578

**TREC Answer:** The protein encoded by BRCA1 interacts in vivo with the BRCA1-associated RING domain (BARD1) protein. Accordingly, BARD1 is likely to be a critical factor in BRCA1-mediated tumor suppression and may also serve as a target for tumorigenic lesions in some human cancers. PMID: 9425226
<table>
<thead>
<tr>
<th>Question #169: How does APC (adenomatous polyposis coli) protein affect actin assembly?</th>
</tr>
</thead>
</table>
| **Query:** APC + *mesh#Actins**  **Expanded:** no  
**Aspect:** ACTINS (MeSH)  
**Results:** 19  **Clicks:** 5  **Position:** 2  
**GoPubMed Answer:** Depletion of APC or Striatin affected the localization of the tight junction protein ZO-1 and altered the organization of F-actin. PMID: 18502210  
**TREC Answer:** Previous studies have shown that these APC protein clusters localize to the ends of microtubules, and that this localization is independent of the actin cytoskeleton. PMID: 9024698 |

<table>
<thead>
<tr>
<th>Question #170: How does COP2 contribute to CFTR export from the endoplasmic reticulum?</th>
</tr>
</thead>
</table>
| **Query:** COP2 AND CFTR + *go#"ER-associated protein catabolic process"**  **Expanded:** yes*  
**Aspect:** ER-ASSOCIATED PROTEIN CATABOLIC PROCESS (GO)  
**Results:** 3  **Clicks:** 3  **Position:** 2  
**GoPubMed Answer:** COPII-dependent export of cystic fibrosis transmembrane conductance regulator from the ER uses a di-acidic exit code. PMID: 15479737  
**TREC Answer:** export of CFTR from the ER is regulated by the conventional coat protein complex II (COPII) in all cell types tested PMID: 11799116 |

<table>
<thead>
<tr>
<th>Question #171: How does Nurr-77 delete T cells before they migrate to the spleen or lymph nodes and how does this impact autoimmunity?</th>
</tr>
</thead>
</table>
| **Query:** Nurr-77 + *mesh#"T-Lymphocytes"**  **Expanded:** yes*  
**Aspect:** T-LYMPHOCYTES (MeSH)  
**Results:** 100  **Clicks:** 2  **Position:** 2  
**GoPubMed Answer:** Nur77 is a nuclear orphan steroid receptor that has been implicated in negative selection when immature T cells are strongly activated through interaction with self peptide-MHC complexes. PMID: 18429814  
**TREC Answer:** Egr-1 has also been reported to regulate delayed early expression of another immediate early gene, nur77 (71), which itself has been implicated in thymocyte cell death PMID: 9034151 |

<table>
<thead>
<tr>
<th>Question #172: How does p53 affect apoptosis?</th>
</tr>
</thead>
</table>
| **Query:** p53 + *go#apoptosis**  **Expanded:** no  
**Aspect:** INDUCTION OF APOPTOSIS (GO)  
**Results:** 409  **Clicks:** 1  **Position:** 1  
**GoPubMed Answer:** p53 is involved in inducing testicular apoptosis in mice by the altered redox status following tertiary butyl hydroperoxide treatment. PMID: 18619952  
**TREC Answer:** p53 induces the expression of genes that mediate withdrawal from the cell cycle (i.e. growth arrest or apoptosis PMID: 9006915 |
<table>
<thead>
<tr>
<th>Question #173: How do alpha7 nicotinic receptor subunits affect ethanol metabolism?</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Query:</strong> alpha7 nicotinic receptor  <strong>Expanded:</strong> no</td>
</tr>
<tr>
<td><strong>Aspect:</strong> ETHANOL METABOLIC PROCESS (GO)</td>
</tr>
<tr>
<td><strong>Results:</strong> 1  <strong>Clicks:</strong> 2  <strong>Position:</strong> 1</td>
</tr>
<tr>
<td><strong>GoPubMed Answer:</strong> Deletion of the alpha7 nicotinic receptor subunit gene results in increased sensitivity to several behavioral effects produced by alcohol. PMID: 15770102</td>
</tr>
<tr>
<td><strong>TREC Answer:</strong> n.a.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Question #174: How does BRCA1 ubiquitinating activity contribute to cancer?</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Query:</strong> BRCA1 +*go#&quot;ubiquitin binding&quot;  <strong>Expanded:</strong> no</td>
</tr>
<tr>
<td><strong>Aspect:</strong> NEOPLASMS (MeSH)</td>
</tr>
<tr>
<td><strong>Results:</strong> 2  <strong>Clicks:</strong> 3  <strong>Position:</strong> 1</td>
</tr>
<tr>
<td><strong>GoPubMed Answer:</strong> The breast and ovarian tumor suppressor BRCA1 catalyzes untraditional polyubiquitin chains that could be a signal for processes other than proteolysis. PMID: 17283126</td>
</tr>
<tr>
<td><strong>TREC Answer:</strong> The finding of constitutive BARD1 missense mutations in breast cancer patients (Thai et al., 1998) and missense mutations within the RING domain of BRCA1 that mediates BARD1 binding activity, has led to the proposal that BARD1 plays a role in tumor suppression in conjunction with BRCA1. PMID: 9832560</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Question #175: How does L2 interact with L1 to form HPV11 viral capsids?</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Query:</strong> L1 AND L2 AND HPV11 +*go#&quot;viral capsid&quot;  <strong>Expanded:</strong> no</td>
</tr>
<tr>
<td><strong>Aspect:</strong> VIRAL CAPSID (GO)</td>
</tr>
<tr>
<td><strong>Results:</strong> 2  <strong>Clicks:</strong> 2  <strong>Position:</strong> 2</td>
</tr>
<tr>
<td><strong>GoPubMed Answer:</strong> The human papillomavirus (HPV) capsid consists of 360 copies of the major capsid protein, L1, arranged as 72 pentamers on a T=7 icosahedral lattice, with substoichiometric amounts of the minor capsid protein, L2. PMID: 12663788</td>
</tr>
<tr>
<td><strong>TREC Answer:</strong> The papillomavirus virion capsid consists of two proteins, L1 and L2. L1 is the major capsid protein and when expressed in eukaryotic expression systems is able to self-assemble into virus-like particles (VLPs) (27,28). The L2 protein has been reported to be involved in DNA binding and encapsidation (29). PMID: 9469843</td>
</tr>
</tbody>
</table>
Question #176: How does Sec61-mediated CFTR degradation contribute to cystic fibrosis?

**Query:** Sec61 + mesh#"Cystic Fibrosis Transmembrane Conductance Regulator"  Expanded: no

**Aspect:** Cystic Fibrosis (MeSH)

**Results:** 6  Clicks: 2  Position: 1

**GoPubMed Answer:** Polytopic proteins are synthesized in the endoplasmic reticulum (ER) by ribosomes docked at the Sec61 translocation channel. It is generally assumed that, upon termination of translation, polypeptides are spontaneously released into the ER membrane where final stages of folding and assembly are completed. Here we investigate early interactions between the ribosome-translocon complex and cystic fibrosis transmembrane conductance regulator (CFTR), a multidomain ABC transporter, and demonstrate that this is not always the case. PMID: 16166089

**TREC Answer:** The mechanism underlying cystic fibrosis transmembrane conductance regulator transport from the endoplasmic reticulum to the proteasome includes Sec61beta and a cytosolic, deglycosylated intermediary. PMID: 10570223

Question #177: How do Bop-Pes interactions affect cell growth?

**Query:** Bop AND Pes + mesh#"Cell Growth Processes"  Expanded: yes

**Aspect:** Cell Growth Processes (MeSH)

**Results:** 6  Clicks: 3  Position: 1

**GoPubMed Answer:** The nucleolar PeBoW-complex, consisting of Pes1, Bop1 and WDR12, is essential for cell proliferation and processing of ribosomal RNA in mammalian cells. PMID: 16738141

**TREC Answer:** Interestingly, a potential homologous complex of Pes1-Bop1-WDR12 in yeast (Nop7p-Erb1p-Ytm1p) is involved in the control of ribosome biogenesis and S phase entry. In conclusion, the integrity of the PeBoW complex is required for ribosome biogenesis and cell proliferation in mammalian cells. PMID: 16043514

Question #178: How do interactions between insulin-like GFs and the insulin receptor affect skin biology?

**Query:** IGF + mesh#Skin + mesh#"Receptors, Insulin"  Expanded: no

**Aspect:** regulation of keratinocyte migration (GO)

**Results:** 2  Clicks: 1  Position: 1

**GoPubMed Answer:** In contrast, when keratinocytes are exposed to UVB in the absence of IGF-1R activation, the keratinocytes are more sensitive to UVB-induced apoptosis, but the surviving keratinocytes retain the capacity to proliferate. PMID: 18216278

**TREC Answer:** IGF-2 was shown to participate in growth-promoting interactions with the IR during development (34). Such IGF regulatory factors could be secreted not only by the keratinocytes, but also by the dermal fibroblasts; their absence in cell culture could lead to further or facilitated deterioration of cellular physiology. PMID: 11181540
<table>
<thead>
<tr>
<th>Question #179: How do interactions between HNF4 and COUP-TF1 suppress liver function?</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Query:</strong> HNF4 +*mesh#&quot;COUP Transcription Factors&quot; <strong>Expanded:</strong> yes</td>
</tr>
<tr>
<td><strong>Aspect:</strong> CARCINOMA, HEPATOCELLULAR (MeSH)</td>
</tr>
<tr>
<td><strong>Results:</strong> 9  <strong>Clicks:</strong> 2  <strong>Position:</strong> 1</td>
</tr>
<tr>
<td><strong>GoPubMed Answer:</strong> Gel-shift assays showed that these motifs are recognized with different affinities by HNF4 and the orphan nuclear receptors chicken ovalbumin upstream promoter transcription factors COUP-TFI and COUP-TFII. In hepatoma cells, the site showing highest affinity for HNF4 appears to be crucial for promoter activity. PMID: 11085951</td>
</tr>
<tr>
<td><strong>TREC Answer:</strong> Chicken ovalbumin upstream promoter transcription factors act as auxiliary cofactors for hepatocyte nuclear factor 4 and enhance hepatic gene expression. PMID: 9717844</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Question #180: How do Ret-GDNF interactions affect liver development?</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Query:</strong> Ret AND GDNF <strong>Expanded:</strong> yes</td>
</tr>
<tr>
<td><strong>Aspect:</strong> HEPATOCYTES (MeSH)</td>
</tr>
<tr>
<td><strong>Results:</strong> 3  <strong>Clicks:</strong> 3  <strong>Position:</strong> 1</td>
</tr>
<tr>
<td><strong>GoPubMed Answer:</strong> When mIMCD-3 cells stably expressing the phosphorylated c-ret receptor were cultured in a type I collagen matrix, they exhibited little GDNF-independent or -dependent branching process formation at early time points compared with the known morphogen hepatocyte growth factor (HGF) (48 h; control, 0.33 +/- 0.33; GDNF, 1.0 +/- 0.58, P = nonsignificant; and HGF, 6.33 +/- 0.33 processes/20 cell clusters, P &lt; 0.001), whereas extended culture (7 days) under serum-free conditions revealed a marked increase in cell survival and the spontaneous development of rudimentary branching process formation. PMID: 10198418</td>
</tr>
<tr>
<td><strong>TREC Answer:</strong> n.a.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Question #181: How do mutations in the Huntingtin gene affect Huntington’s disease?</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Query:</strong> huntingtin +*mesh#Mutation <strong>Expanded:</strong> no</td>
</tr>
<tr>
<td><strong>Aspect:</strong> AUTOSOME (GO)</td>
</tr>
<tr>
<td><strong>Results:</strong> 48  <strong>Clicks:</strong> 2  <strong>Position:</strong> 1</td>
</tr>
<tr>
<td><strong>GoPubMed Answer:</strong> Huntington’s disease (HD) is an inherited autosomal dominant neurodegenerative disease caused by the expansion of a CAG trinucleotide repeat in exon 1 of the huntingtin (htt) gene. PMID: 18067195</td>
</tr>
<tr>
<td><strong>TREC Answer:</strong> Somatic instability of the expanded allele of IT-15 from patients with Huntington disease. PMID: 8733127</td>
</tr>
</tbody>
</table>
Question #182: How do mutations in Sonic Hedgehog genes affect developmental disorders?

Query: sonic hedgehog +*mesh#Mutation  Expanded: no
Aspect: HOLOPROSENCEPHALY (MeSH)
Results: 17  Clicks: 1  Position: 1

GoPubMed Answer: Mutations of the developmental gene Sonic hedgehog (SHH) and alterations of SHH signaling have been associated with holoprosencephaly (HPE), a rare disorder characterized by a large spectrum of brain and craniofacial anomalies. PMID: 15292211

TREC Answer: Mutations in the human sonic hedgehog gene cause holoprosencephaly. PMID: 9302279

Question #183: How do mutations in the NM23 gene affect tracheal development?

Query: NM23 tracheal  Expanded: no
Aspect: GENES, TUMOR SUPPRESSOR (MeSH)
Results: 78  Clicks: 1  Position: 1

GoPubMed Answer: We hypothesize that elevation of nm23-H1 metastasis suppressor gene expression in micrometastatic tumor cells may reduce their subsequent colonization and invasion, and induce differentiation, with a clinical benefit. PMID: 11918081

TREC Answer: Site-directed mutation of Nm23-H1. Mutations lacking motility suppressive capacity upon transfection are deficient in histidine-dependent protein phosphotransferase pathways in vitro. PMID: 15026478

Question #184: How do mutations in the Pes gene affect cell growth?

Query: Pes +*go#*cell growth*+*mesh#Mutation  Expanded: yes
Aspect: CELL GROWTH PROCESSES (MeSH)
Results: 4  Clicks: 1  Position: 1

GoPubMed Answer: Dominant-negative Pes1 mutants inhibit ribosomal RNA processing and cell proliferation via incorporation into the PeBoW-complex. PMID: 16738141

TREC Answer: The contribution of pescadillo to the regulation of cell growth was further substantiated by the identification and characterization of temperature-sensitive pescadillo mutant yeast strains. PMID: 11071894

Question #185: How do mutations in the hypocretin receptor 2 gene affect narcolepsy?

Query: hypocretin receptor 2 +*mesh#Mutation  Expanded: yes
Aspect: NARCOLEPSY (MeSH)
Results: 15  Clicks: 3  Position: 3

GoPubMed Answer: Familial and sporadic forms of narcolepsy exist in both humans and canines. Mutations in the hypocretin receptor 2 gene (Hcrtr 2) cause canine familial narcolepsy. PMID: 11442359

TREC Answer: canine narcolepsy is caused by a mutation in the hypocretin (orexin) receptor 2 gene PMID: 10767319
Question #186: How do mutations in the Presenilin-1 gene affect Alzheimer’s disease?

Query: Presenilin-1 +*mesh#Mutation  Expanded: yes

Aspect: ALZHEIMER DISEASE (MeSH)

Results: 466  Clicks: 3  Position: 1

**GoPubMed Answer:** We report a novel mutation in exon 8 of the presenilin 1 (PSEN1) gene (V261L) associated with early-onset autosomal dominant Alzheimer’s disease and spastic paraparesis. PMID: 18637955

**TREC Answer:** The structure of the presenilin I gene and the identification of six mutations in early onset AD pedigrees. PMID: 8817335

Question #187: How do mutations in familial hemiplegic migraine type 1 (FHM1) gene affect calcium ion influx in hippocampal neurons?

Query: FHM1 +*mesh#Mutation  Expanded: no

Aspect: HIPPOCAMPUS (MeSH)

Results: 2  Clicks: 1  Position: 1

**GoPubMed Answer:** These channels are normally vital for presynaptic Ca2+ entry and neurotransmitter release at many central synapses, raising questions about effects of FHM1 mutations on neuronal Ca2+ influx and inhibitory and excitatory neurotransmission. PMID: 15699344

**TREC Answer:** Based on the differential localization of these subunits and our findings FHM1 mutations may induce different patterns of Cav2.1 dysfunction in different neurons and even between different subcellular compartments of a single neuron. PMID: 15448138

---

**Answering biomedical questions**

Which particular diseases are associated most often with HIV? Query for “HIV” the most relevant topics in the section diseases besides HIV Infections are, ACQUIRED IMMUNODEFICIENCY SYNDROME, HEPATITIS, TUBERCULOSIS, VIREMIA and SYPHILIS. Clicking on tuberculosis retrieves the relevant articles including statements such as “HIV and parasitic co-infections in tuberculosis patients”.

What kind of diseases are caused by HIV? Navigating into the branch of diseases the overview reveal BACTERIAL INFECTIONS AND MYCOSES, VIRUS DISEASES, FEMALE UROGENITAL DISEASES and PREGNANCY COMPLICATIONS, IMMUNE SYSTEM DISEASES and MALE UROGENITAL DISEASES as important categories of illnesses.

Which techniques of treatment are used to help HIV patients? ANTIRETROVIRAL THERAPY are drug regimens, that aggressively suppress HIV replication. The regimens usually involve administration of three or more different drugs including a protease inhibitor.

Who are the top authors for Antiretroviral Therapy and where was this research done by those authors and when? The statistics show that B. Gazzard, J. Montaner and V. Soriano published most actively among others. Brian Gazzard published more than 100 articles from 1988 to 2007 in the Westminster Hospital, London, UK. Julio Montaner published more than 100 articles between 1988 and 2007 with Department of Medicine of University of British Columbia, Vancouver. Vicente Soriano published more
than 80 articles in this field between 1991 and 2007 with the Servicio de Enfermedades Infecciosas, Instituto de Salud Carlos III, Madrid.

**Which are leading centers and scientists for liver transplantation?**

**What are common synonyms for liver transplantation and what is done in this operation?** Hepatic Transplantation and Liver Grafting are commonly used to describe the transference of a part of or an entire liver from one human or animal to another.

**Who is the leading German scientist in the field of liver transplantations and where is he practicing?** Peter Neuhaus works at the Chirurgische Klinik, Universitätshospital Rudolf Virchow, Berlin. He has more than 500 publications in the field.

**What problems make liver transplantations necessary?** Viral-induced cirrhosis (27.1%), Alcoholic liver disease (21%), Liver Neoplasms (15.7%) and Cholestatic liver disease (14.6%) are among the diseases eventually requiring liver transplantations.

**What terminology complements the concept of recurrence?** Recrudescences and Relapses are synonyms of Recurrence which is the return of a sign, symptom, or disease after a remission.

**Which anatomical structure is affected by the bacterium helicobacter pylori?** Gastric Mucosa, the lining of the stomach, consisting of an inner epithelium, a middle lamina propria, and an outer muscularis mucosae are affected by the bacterium.

**Is the research on leukemia decreasing?** A progressive, malignant disease of the blood-forming organs, characterized by distorted proliferation and development of leukocytes and their precursors in the blood and bone marrow. Absolute number of research articles in this field is still growing by roughly 100 articles per year. The relative research interest is decreasing by 0.5% since 1984.

**Who are top authors in leukemia and what are there other fields of interest?** Kantarjian H (Houston, USA), internationally leading author in Leukemia, Myeloid, Chronic, Remission Induction, Leukemia, Myeloid, Philadelphia-Positive, Cytarabine, Leukemia, Myelocytic, Acute. Antineoplastic Combined Chemotherapy Protocols, Antineoplastic Agents, [1987 - 2007] Department of Hematology, University of Texas, M. D. Anderson Hospital and Tumor Institute at Houston 77030.

Keating MJ (Houston, USA), internationally leading author in Leukemia, Lymphocytic, Chronic, Vidarabine, Cytarabine, Remission Induction, Leukemia, Myelocytic, Acute. Antineoplastic Combined Chemotherapy Protocols [1987 - 2007] Department of Hematology, University of Texas, M. D. Anderson Hospital and Tumor Institute at Houston 77030.


---

1PMID: 18031464
Acknowledgments

This thesis has been a source of great joy for me in the last four years. Without the help of some great people this would not have been possible.

First of all, the person who supported me most with ideas, knowledge and guidance has been Michael Schroeder. Under his excellent supervision I learned how to do research and could work together with some remarkable people. His open-mindedness and patience allowed for truly creative work. His gentle but constant pressure helped me not to lose the focus.

When Thomas Wächter and Heiko Dietze were the joining the group the project turned from a academic prototype into a professional technology. With their incredible knowledge and enthusiasm for the idea they fueled also my passion. Loic Royer lives creativity. Discussions with him were always a source of ingenious ideas.

A huge thanks goes to all developers and employees of Transinsight with whom I had the luck to work together on a real life project. This viewpoint outside of academia and the success of a growing company gave me confidence to create something meaningful.

I want to express my deep-felt gratitude to all my colleagues in the Biotechnology Center in Dresden. In particular I want to thank all members of the Bioinformatics group for fruitful discussions and practical support especially when I needed your biological expertise. Thanks Dimitra Alexopoulou for 1,600 curated articles.

Finally I dedicate this work to my family. There was not a single moment in which your support was not there. Thanks to my mother who taught me to make my own decisions, to my father who always believed in my success, to my brother who always was a good host and to Jana who I always carry in my heart and mind.