Text Mining for Biology and Biomedicine

For a list of recent related titles from Artech House,
turn to the back of this book.

Text Mining for Biology and Biomedicine

Sophia Ananiadou
John McNaught

Editors

ARTECH HOUSE

BOSTON | LONDON
artechhouse.com

Library of Congress Cataloging-in-Publication Data
A catalog record for this book is available from the U.S. Library of Congress.

British Library Cataloguing in Publication Data
Text mining for biology and biomedicine. — (Artech House bioinformatics series)
1. Biology—Terminology—Data processing 2. Medicine—Terminology—Data processing 3. Keyword searching 4. Data mining
I. Ananiadou, Sophia II. McNaught, John
570.2'85574

ISBN-10: 1-58053-984-x

Cover design by Igor Valdman

685 Canton Street
Norwood, MA 02062

International Standard Book Number: 1-58053-984-x

10 9 8 7 6 5 4 3 2 1

Contents

1

Introduction

Sophia Ananiadou and John M°Naught

1.1 Text Mining: Aims, Challenges, and Solutions

With an overwhelming amount of biomedical knowledge recorded in texts, it is not surprising that there is so much interest in techniques that can identify, extract, manage, integrate, and exploit this knowledge, and can discover new, hidden, or unsuspected knowledge. In the past few years, there has been an upsurge of research papers on the topic of text mining from biomedical literature. The primary goal of text mining is to retrieve knowledge that is hidden in text, and to present the distilled knowledge to users in a concise form. The advantage of text mining is that it enables scientists to efficiently and systematically collect, maintain, interpret, curate, and discover knowledge needed for research or education.

A commonly accepted definition of text mining, to which we also adhere, is provided by Hearst [1]. She characterizes text mining as the process of discovering and extracting knowledge from unstructured data, contrasting it with data mining, which discovers knowledge from structured data. Under this view, text mining comprises three major activities: information retrieval, to gather relevant texts; information extraction, to identify and extract a range of specific types of information from texts of interest; and data mining, to find associations among the pieces of information extracted from many different texts.

In this book, we leave aside information retrieval aspects. These have been widely investigated and reported, and most biologists use information retrieval, in some form or another, in their daily work. Indeed, the search engine

experience is so pervasive and familiar that it becomes hard to appreciate that there may be something beyond information retrieval. However, in the text mining world, information retrieval is but the first step. Instead of leaving the user with the problem of having to read several tens of thousands of retrieved documents, text mining offers the possibility of extracting precise facts from a retrieved document set, and of finding interesting associations among disparate facts, leading to the discovery of new or unsuspected knowledge. We thus focus here on activities subsequent to information retrieval—some of which also may help improve classic information retrieval (e.g., terminology management, the subject of Chapter 4).

It is not only the amount of unstructured textual data that poses problems. This type of data is increasing in volume at such a pace that it is becoming difficult to discover knowledge and to generate scientific hypotheses without the use of text mining techniques. Significantly, hypothesis generation relies on background knowledge, and is crucial in scientific discovery. The pioneering work by Swanson et al. on hypothesis generation [2] is mainly credited with sparking interest in text mining techniques in biology. Text mining aids in the construction of hypotheses from associations derived from vast amounts of text that are then subjected to experimental validation by experts. It is being applied in numerous areas, such as finding functional relationships among genes, establishing functional annotations, discovering protein-protein interactions, interpreting array experiments, associating genes and phenotypes, and so forth.

This increased interest in text mining in biology is also evident through numerous dedicated workshops,[1] tutorials, and special tracks at major conferences in bioinformatics and natural language processing (NLP). Examples of these conferences include the Pacific Symposium on Biocomputing, Intelligent Systems for Molecular Biology, the Association for Computational Linguistics Annual Meeting, and the International Conference on Computational Linguistics.

Perhaps the most concrete evidence of recognition of the strategic significance of text mining is seen in the creation of the U.K. National Text Mining Centre (http://www.nactem.ac.uk), which has been recently set up to provide services in text mining to the academic community. Moreover, the Centre is concentrating its activities on text mining in biology, due to the amount of user demand in this area.

This book introduces the field of text mining to those interested in organizing, searching, discovering, or communicating biological knowledge, and aims to arm them with a sound appreciation of its main techniques, concerns, challenges, results, and promising future directions. As seen in other areas

1. A recent example is the *First International Symposium on Semantic Mining in Biomedicine*, organized by the European Bioinformatics Institute, in April 2005.

involving the introduction of new technology in the shape of applied information systems, there is a danger of expectations exceeding reality, leading to disappointment and rejection. Thus, a further aim of this book is to critically examine the state of the art, and to make clear what can be expected of the field at present or in the near future. The reader will find extensive summarization and discussion of the research literature in text mining and reported systems, geared towards informing and educating, rather than oriented towards other experts in text mining. To this end, this book has been conceived as a number of complementary chapters, which target core topics. These chapters were specially commissioned from leading experts around the world, and have undergone a strict peer-reviewing procedure. Each chapter takes its own view of its subject matter. However, the reader will find, on occasion, the same topic being discussed from a different point of view in different chapters. This was due to a deliberate policy of encouraging informative discussion, rather than artificial compartmentalization. The reader also will find differences of opinion, of terminology, and of fundamental approach. Text mining is a complex, dynamic area, with many techniques and approaches being tried out. It would be foolhardy to attempt to gloss over the differences that naturally occur due to this dynamism and complexity, or to give the appearance of consensus where there may be none. Where there is consensus, this has been brought out, and where there are differing voices and views, these have been left untouched. Thus, the reader will appreciate which areas are controversial, and which are considered mature and a good foundation to build on. For those wishing an approachable, concise explanation of the concerns, techniques, and information problems of molecular biology, viewed from the perspective of how people interact with information and technology, we recommend the article by MacMullen and Denn [3]. Other overviews are referred to throughout the book.

Significantly, text mining does not just provide existing tools for application to the biology domain. A major reason why text miners have engaged so closely with this domain is that it presents a number of challenges, which have necessitated new and different approaches. Challenges range from having to deal with the particular language of the biologist, to building scalable and robust systems, to presenting the results of text mining in meaningful and informative ways (to the biologist).

Biology also interacts closely with different disciplines (e.g., chemistry and medicine), and this interaction presents further challenges to text miners, who have to deal with interdisciplinary aspects, and user communities with different views over the same knowledge space and with different information needs. An example is that of a cell, which can be described by a bacteriologist, an immunologist, a neurologist, or a biochemist, each from his or her own point of view. Such multidimensionality must be maintained and appropriately managed.

Moreover, text mining in itself is a truly interdisciplinary area. Knowledge of language (linguistics) is needed, alongside natural language processing techniques (Chapter 2), such as part-of-speech tagging, full parsing and shallow parsing (which are combined with information retrieval), information extraction (Chapters 6 and 7), and data mining techniques (Chapter 10). Such a degree of interdisciplinarity, in both the technology domain and the application domains, ensures that no text miner is ever without a challenge.

These and other challenges continually exercise the text mining community, as can be seen in discussion of problems, solutions, and results in the Critical Assessment of Information Extraction in Biology (BioCreAtIvE) evaluations (Section 9.4.3). Many information extraction tasks, which had been considered to give highly successful results when applied to business newswires, suffered a significant drop in performance when they were applied in the biomedical area. It was found that identifying gene names and carrying out functional annotation were among the hardest of tasks. Challenge evaluations, such as BioCreAtIvE, compare the performance of text mining systems on specific biology-oriented tasks, using a gold standard annotated text corpus to enable proper comparison. Indeed, one of the positive outcomes of evaluation challenges is the availability of annotated corpora to be used for subsequent training and testing (e.g., BioCreAtIvE, TREC genomics track [4]), although there is still a great need for more such data. Chapter 8 discusses this and reports on efforts to produce one such corpus.

Terminology presents one of the core challenges for text mining from biomedical literature. Given the amount of neologisms characterizing biomedical terminology, it is necessary to provide tools that will automatically extract newly coined terms from texts, and link them with biodatabases, controlled vocabularies, and ontologies. The importance of this topic has triggered significant research, which has in turn resulted in several approaches used to collect, classify, and identify term occurrences in biomedical texts. Terminological processing also covers such aspects as extraction, term variation, classification, and mapping (Chapter 4).

The high degree of term variation in biomedicine creates problems when we want to map termforms in text to existing controlled vocabularies or to concept labels of ontologies. It also hampers information retrieval tasks, since many relevant documents are not retrieved, due to a mismatch of index terms with the user's queries. A very prolific type of term variation is abbreviation—approximately 64,000 new abbreviations were introduced in 2004 in the biomedical literature alone (Chapter 5). The specificity of biomedical abbreviations makes their identification a challenging task, and demands specific solutions that are different from those for general language processing. Work on abbreviation identification has led to the construction of biomedical abbreviation databases, such as AcroMed, SaRAD, and the Stanford Biomedical Abbreviation Database.

Knowledge resources, such as the Unified Medical Language System (UMLS) [5], contain an abundance of nomenclatures, controlled vocabularies, and ontologies (Chapter 3). Ontologies are crucial for knowledge discovery in biomedicine. They form the link between terms in texts and biological databases. They are of great relevance to text mining, because they can be used to add meaning. They also can be used for semantic annotation of texts (Chapters 5, 7, and 8). However, in order to be useful, they must be internally consistent, offer (jointly) wide coverage of the domain while allowing multiple classifications of entities, and support interoperability. Currently, information available in existing biomedical resources, such as ontologies, is not sufficient for text mining applications, for a number of reasons. These resources are often focused on human users, have limited lexical and terminological coverage of different biological domains, experience problems with update and curation due to new terms being created daily, and are difficult to integrate to achieve wider coverage and consistency. Text mining shows its flexibility and usefulness. Not only can it process texts to extract facts and associations, it also can be used to support update of ontologies and to ensure wide coverage. Text mining can aid tasks, such as curation of biomedical databases, by consistently applying ontological annotation, and it can help in ontology construction and update, yielding increased ontological coverage of biological entities.

The issue of resources concerns all the authors in this collection. The text mining community in general is aware that there is a "resource bottleneck." There is a lack of large-scale, richly annotated corpora to support training of machine learning algorithms, development of computational grammars, and evaluation of text mining components. Such resources are expensive and time-consuming to produce. There is also a lack of appropriate knowledge resources: lexicons, terminologies, and ontologies. We are very precise in our use of "appropriate" here, as will become apparent in discussions in Chapters 2, 3, 4, and 7. We mean "appropriate for text mining." Sophisticated information extraction of the kind required to handle biotexts requires access to such resources. However, there is a lack of special language lexicons: computational lexicons that encode the special language behavior of terms in the domain, including, crucially, that of terminological verbs. Terminological description must become more than mere recording or listing of forms. The linguistic behavior of terms must be captured formally and explicitly in computational lexicons to drive text-mining systems.

Moreover, the field needs to tidy up its terminological act, and to recognize that there is a world of difference between, on the one hand, an index term, a controlled vocabulary item, a concept label, a thesaurus descriptor, and a subject heading; and, on the other hand, a terminological, linguistic term and its family of variants, all of which (including the canonical head term or entry term itself) are *necessarily attested* in text (Chapter 4, Section 4.5.1). Nothing can be

considered a term unless it is at least attested. Rule-based information extraction systems rely on finding information on textual forms, partly through dictionary look-up, and thus rely on finding information about terms as they occur in texts. The systems struggle to achieve anything sensible in that regard, by looking up resources that are divorced from the reality of the textual term. However, ontologies are also urgently required. We need to be able to assign domain-specific meanings to terms, to use these meanings in building up representations of facts and events, and to navigate and perform inference based on ontological relationships. As Chapter 3 makes clear, not all ontologies are consistently constructed following best practice in ontological design. Reengineering is indicated in such cases to render them more useful for applications such as text mining. However, even given good ontologies and good lexicons, there still is a further need to establish the missing link between lexicon and ontology, which provides mappings from terms in lexicons to corresponding concepts in ontologies.

Such efforts, which are required to establish and maintain appropriate annotated corpora and lexical/ontological resources, are by no means trivial—community efforts are clearly indicated here. Fortunately, the field can benefit from earlier work on standardization of resources for language engineering applications (e.g., EAGLES/ISLE [6], ISO TC 37 SC 4 [7], and TEI [8]). However, because of the specificity of biomedicine, porting general NLP standards to biomedicine has limitations, and further work is required to specialize NLP standards for practical use in text mining from biology texts.

The future existence of such resources holds out the promise of greater benefits from text mining, including the possibility of being able to fully exploit annotated, structured data from biodatabases and annotated experimental data, together with the results of processing unstructured data through text mining, to yield true integration of heterogeneous resources in the search for new knowledge (Chapter 10).

Before describing the contents of each chapter, we note in conclusion that the flexibility and applicability of text mining extends to endowing existing collections of text with added value. That is, not only can text mining extract facts and associations from text, it also can add back in such information, in the form of metadata, to enable sophisticated and precise searching. Additionally, we must be aware that it is crucial for any extracted facts to be grounded in the literature. Currently, about the only commonly available method of linking texts to biology databases is through their PubMed unique identifier (PMID). Text mining techniques have a key role to play in linking scientific articles to biology databases. Applied in these ways, text mining can open up many possibilities to enrich digital libraries and open access collections, and to connect them with curated databases.

In closing, we trust that we have given the reader an appetite to learn more about text mining for biology. Now make way for the main course. First, a closer look at the menu.

1.2 Outline of the Book

In Chapter 2, Hahn and Wermter provide an introduction to the general techniques of NLP, since a basic understanding of these techniques is necessary to fully appreciate discussions in following chapters. They adopt a level-oriented approach, showing how the different linguistic levels (e.g. morphology, syntax, and semantics) account for the various phenomena to be found in biomedical text. Each level has associated processing components, with typically several types of components implementing different approaches to tackling phenomena at some level. They focus on practical processing issues concerning biomedical text, and they present a general, natural language system architecture for text mining. This shows how the different processing components and domain resources can be applied, starting from the raw text, to yield various kinds of analyses. Furthermore, this architecture acts as a point of reference for the reader throughout the book, allowing rapid appreciation of how some particular type of component or resource discussed in later chapters relates to the overall text-mining task.

Chapter 3, by Bodenreider, presents a description and discussion of major, publicly available lexical, terminological, and ontological resources that can be used to support text mining. As is made clear, there is often no sharp distinction made between these three different kinds of resources. This can lead to confusion, and can hamper attempts at exploitation for text mining. To help the reader understand the nature of problems involved in mapping wordforms to corresponding entries in various resources, a sample text is taken as a basis for extended discussion. Major lexical, terminological, and ontological resources are then described, and commonalities and distinguishing characteristics are discussed. Next, the different complementary roles of the three types of resources in entity recognition and fact extraction are examined. The author concludes that much work still needs to be done to provide adequate terminological coverage in the domain, and to produce or reengineer ontologies according to consistent, formal principles.

In Chapter 4, Ananiadou and Nenadic examine how text mining can facilitate terminology management in biomedicine. Efficient term management is crucial for mining biomedical literature, since terms are a key means for communicating knowledge. They examine the basic concepts of term and terminology, the relation between concepts and terms, and issues related to term variation and term ambiguity. They also present approaches for automatically

recognizing and structuring terminology from literature, and for discovering associations between terms that can then be used for terminology management.

The main aim of automatic terminology management is to provide tools to bridge the gap between in-text terms and domain concepts, and to facilitate effective mining of scientific literature and integration of databases. It also aims to develop and apply efficient and accurate methods for building and updating terminological resources, and for supporting on-the-fly recognition of terms in text.

Chapter 5, by Chang and Schütze, goes into detail on a particular type of term variation, namely abbreviation. The fact that an entire chapter is devoted to this topic is indicative of the problems that abbreviations pose, both to humans and to text mining applications. Where there is failure to properly handle abbreviations, a text mining system will demonstrate reduced accuracy. In essence, handling abbreviations means linking an abbreviation with its expanded form(s). Unfortunately, there is variation in the interpretation of what constitutes an abbreviation, which makes the task harder. A particular area of disagreement concerns how one deals with abbreviation where the abbreviated form apparently maps to some long form, but is arguably abbreviating an even larger form, as in *RNA Polymerase I (Pol I)*. Following a discussion on the problems of identifying candidate abbreviations, details are given of the three main classes of method used to determine whether the candidates are in fact abbreviations: heuristic, alignment, and NLP-based methods. The advantages and disadvantages of these approaches are briefly discussed, and further detail is provided about the algorithm used to construct the Stanford Biomedical Abbreviation Database. A discussion of variations in abbreviation complements that of Chapter 4, and reports on results of abbreviation clustering methods. Since more than 20% of abbreviations occur with no expanded form in the same text, a section is devoted to the problem of finding the correct long form elsewhere. The author concludes that, while there are still unsolved problems in handling abbreviations, current algorithms and abbreviation databases provide invaluable support to text mining.

In Chapter 6, Park and Kim describe the task of named entity recognition (NER) in biomedicine. NER is one of the most widely studied areas of information extraction, not only in biomedicine but also in general language. The authors focus on the different approaches adopted for NER in biomedicine, such as dictionary-based, rule-based, machine learning, and hybrid methods. NER is mainly a classification task, which aims to use the recognized entities as slot fillers in specified templates (e.g., protein-protein interactions, gene-disease relationships, and so forth). The problems reported with the recognition of entities, such as ambiguity and variation, are similar to the problems reported in automatic terminology management. The main difference is that, in term identification, the classification task aims to distinguish between terms and nonterms, while in NER, the classification task focuses on specific classes of entities (e.g., genes and proteins). Adapting NER for more fine-grained classes is

a more challenging task. A final, nontrivial, step in NER is the grounding of the recognized entities with the relevant entry identifiers in biomedical resources.

Chapter 7 concentrates on rule-based approaches to simple fact and complex event information extraction (IE). McNaught and Black start by placing IE in its general context, referring to the Message Understanding Conferences (MUC), which were influential in defining the nature, scope, and measurement of IE tasks. They then offer a classification of approaches to IE in biology. This classification demonstrates the wide variety and combinations of approaches that have been adopted. Each type of approach is critically assessed. Approaches range from straightforward matching of patterns in text, to full-scale syntactic parsing, to sophisticated use of ontologies. A core concern is the extent to which some approaches are able to deliver abstract representations of facts and events, which can be subjected to subsequent data mining or integrated in knowledge bases to enable reasoning, rather than to deliver extracted textual strings or their simple transforms. The authors conclude that the field would benefit both from a greater concentration on characterizing the "informational structure" of domain texts, through an approach simultaneously considering syntax and semantics, and from being able to take greater advantage of ontological information about action and event concepts. The authors reinforce calls for further efforts to establish terminological, lexical, and ontological resources capable of supporting advanced information extraction.

Chapter 8, by Kim and Tsujii, deals with text collections, corpora, and corpus annotation. The major collections favored for biotext mining are briefly described, and characterized in terms of their vocabulary profiles. Corpora derived from these collections are then described. For text mining purposes, the distinguishing features of a corpus are that it is representative of the chosen domain, and, crucially, that it is annotated, possibly with several different, but complementary, types of annotation. An annotated corpus can be used to provide training data for machine learning algorithms, to ensure good coverage of grammars, and to act as a gold standard for evaluation purposes. Annotations can cover linguistic information as well as conceptual information. For example, biological entities and relations between entities can be annotated. The types of annotations employed in various corpora of interest for biotext mining are described and discussed, and the need for appropriate annotation guidelines is emphasized, to avoid ambiguity of, or inconsistency in, interpretation among human annotators. As with other types of resources, harmonization and standardization are issues of current major interest, discussed in relation to corpus annotation schemes and the use of Extensible Markup Language (XML). A fundamental decision for a corpus builder is whether or not to employ stand-off annotation (i.e., whether to separate annotations from the original text or to embed the annotations in the original text). Stand-off annotation allows for multiple overlapping annotations, but causes problems for validation. Finally,

various tools that aid editing, searching, and annotating of corpora are discussed, including semiautomatic annotation, which helps mitigate the cost and time involved in corpus annotation.

In Chapter 9, Hirschman and Blaschke discuss the evaluation of text mining in biology. They argue that, in biotext mining and other areas of applied NLP, and in prediction of protein structure, the development of challenge evaluations and shared assessments has helped advance the state of the art to a significant degree. Community efforts to build shared assessment resources (often too expensive for individual groups to construct), and to agree on evaluation methodologies, are seen as vital in helping to drive forward progress in the field. The authors address the key questions of why to evaluate, considering the dimensions of a successful evaluation, and what to evaluate. A paramount consideration is for evaluation to address problems and tasks of interest to biologists. This is critical, both to keeping the technology focused on problems of practical relevance to biologists, and to obtaining adequate amounts of high-quality gold standard data annotated by subject specialists. Biologists are more likely to become involved in gold standard annotation if the results are also useful for other ongoing, daily tasks, such as curation of biodatabases. An important factor in designing challenge evaluations is to incorporate text mining tasks that increase in difficulty, taking as a baseline the current state of the art, while also maintaining overall relevance of tasks to biologists. The authors then detail the organization and results of recent evaluations of text mining in biology. Their overall conclusions regarding challenge evaluation tasks carried out so far are that gene/protein name identification is largely solved, but that the broader biological entity recognition problem requires further effort. A further plea is made for wider-coverage lexical resources to help in entity recognition. Information retrieval-related tasks, such as selecting or ranking papers according to biologically relevant criteria, still present open research problems, although the authors hint that the classic information retrieval model is not well suited to making the fine-grained distinctions necessary for such tasks. By implication, information extraction and data mining may have more to offer here. Finally, the authors note that we are at the start of a long journey, if we look at challenge evaluations in other fields. Thus, there is still a large space of biologically motivated applications in text mining to explore through further challenge evaluations in the years to come. This necessarily long-term activity, however, is dependent on continued funding, and, crucially, involvement by the biology community in the evaluation endeavor.

Chapter 10, by Ng, discusses the integration of the results of text mining with data mining to facilitate the analysis of biological data. Rather than using or linking the data sources independently, the great challenge lies in facilitating a deep integration of textual knowledge in the data mining algorithms. By leveraging the valuable information from the literature in the data mining process, more biologically significant knowledge can be intelligently mined from the

biological data. Ng describes how text mining has been incorporated in homology searches and in sequence-based functional classification. To improve homology searches, text mining techniques are applied to database annotations and Medical Literature, Analysis, and Retrieval System Online (MEDLINE) references, together with PSI-BLAST, based on literature similarities. Solutions to the problem of predicting the biological functions of new genes and proteins include using supervised machine learning techniques (support vector machines), combining sequence information with literature information, or using a combination of text-based and sequence-based kernels to perform classification. Text mining methods also can be integrated with data mining methods in order to provide intelligent interpretations of gene expression clusters.

Acknowledgments

The U.K. National Centre for Text Mining is funded by the Joint Information Systems Committee, the Biotechnology and Biological Sciences Research Council, and the Engineering and Physical Sciences Research Council.

Conventions

Examples of linguistic data are given in italics. Formal items, such as concept labels, semantic categories, index terms, thesaurus descriptors, predicates, and participant roles, are enclosed in single quotation marks.

References

[1] Hearst, M. A., "Untangling Text Data Mining," *Proc. 37th Annual Meeting of the Association for Computational Linguistics*, College Park, MD, 1999, pp. 3–10.

[2] Swanson, D., "Medical Literature as a Potential Source of New Knowledge," *Bulletin of the Medical Library Association*, Vol. 78, No. 1, 1990, pp. 29–37.

[3] MacMullen, W. J., and S. O. Denn, "Information Problems in Molecular Biology and Bioinformatics," *Journal of the American Society for Information Science and Technology*, Vol. 56, No. 5, 2005, pp. 447–456.

[4] "TREC Genomics Track," http://ir.ohsu.edu/genomics/, accessed June 2005.

[5] Bodenreider, O., "The Unified Medical Language System (UMLS): Integrating Biomedical Terminology," *Nucleic Acids Res.*, Vol. 32, Database issue, 2004, pp. D267–D270.

[6] http://www.ilc.cnr.it/EAGLES96/isle/ISLE_Home_Page.htm, accessed June 2005.

[7] http://www.tc37sc4.org/, accessed June 2005.

[8] http://www.tei-c.org/, accessed June 2005.

2

Levels of Natural Language Processing for Text Mining

Udo Hahn and Joachim Wermter

2.1 Introduction

Because of its inherent complexity, the analysis of natural language documents is usually not carried out in a single, large, monolithic step. Rather, (computational) linguists deal with natural languages at several layers of description (processing). A common decomposition distinguishes between the consideration of words (the lexical level), the organization of groups of words in sentences as phrases or clauses (the syntactic level), and the meaning that can be ascribed to these entities at the content layer (the semantic level). Given such a division into various descriptions and processing levels, a need for integration arises. Hence, the architecture of natural language systems becomes a crucial issue. It provides the organizational platform to integrate the outcome of many decomposed subprocesses, all being part of the grand picture, which is the content-oriented analysis of natural language. In this chapter, we look at the processing of natural language documents for text mining from this holistic architectural perspective, as well as from the perspective of its decomposition into the major components at the lexical, syntactic, and semantic level. We are only concerned with the English language, since it serves as the international lingua franca for the field of biology.

We start at the lexical level in Section 2.2. First, we deal with the problem of identifying words from a given document stream, which is the tokenization problem (see Section 2.2.1). Once words are determined, lexical variants (which

are due to inflection and derivation) are unified through morphological analysis, by assigning these variants a canonical base form (see Section 2.2.2). We are then able to relate these canonical forms, via lexicon look-up, to a corresponding lexical entry in a linguistic lexicon (see Section 2.2.3). This is a repository in which morphological, syntactic, or semantic information about the canonical base form of a word is stored. The coverage of linguistic lexicons is usually restricted to lexical items from the general language we use (e.g., the ones we encounter in daily newspapers). For scientific domains, such as biology, we often lack comprehensive lexical resources (see Chapter 3), not only because of the sheer size of scientific terminologies, but also because they are extremely productive (i.e., new terms are continuously being formed), and almost impossible to keep track of. Besides scientific terminology, concrete named entities (e.g., the names of drugs, genes, or proteins) constitute another major lexical analysis task, which reflects domain-specific patterns of sublanguage use (see Chapters 4 and 6).

Having looked at accessing linguistic information from a lexicon, we turn in Section 2.3 to the identification of structural relationships that hold between groups of words at the sentence level, in the realm of syntactic analysis. As a starting point, we may use categorical information attached to each lexical item in terms of its parts of speech (e.g., verb, adjective, noun, and so forth). The first step of syntactic analysis consists of the assignment of part-of-speech (POS) tags to the sequence of lexical items that makes up a sentence, in a process called tagging (see Section 2.3.1). This is a prerequisite for determining groups of words that grammatically belong together, and thus constitute larger syntactic units, which linguists usually refer to as phrases. The procedure for identifying phrases (often only noun phrases) is called chunking (see Section 2.3.2). While chunks just delimit the beginning and the end of a phrase without further considering its internal structure, parsing procedures add exactly this type of information. This is done, for example, by determining the head of a noun phrase (i.e., the dominating noun), or hierarchies of adjectival attribution related to the head noun (see Section 2.3.3).

The focus on phrasal groupings of words is motivated by the observation that phrases are a linguistic means to denote conceptually relevant entities. This brings us, in Section 2.4, to the consideration of content aspects of natural language, the level of semantic representation, and interpretation. Roughly, we may distinguish between entities that denote things (such as drugs, proteins, genes, and so forth), and relations linking these entities (processes such as interaction, blocking, activation, and regression). This is basically what text mining is about—finding relevant and "new" entities for specific relations, in a process called entity mining, or finding relevant and "new" relations between specific entities, via relation mining.

Content ascription to single terms builds on a semantically enhanced lexicon, a terminology (thesaurus), or an ontology (see Chapter 3) where lexical items (or terms) are linked via different semantic relations, such as hyponymy, synonymy, or antonymy[1] (see Section 2.4.1). Linguistic lexicons usually carry language-dependent specifications (e.g., contextual conditions for verbs with respect to the type of noun phrases or prepositions they require), while language-independent, yet domain-specific repositories, such as thesauri or ontologies, entirely focus on relational semantic specifications, but do not usually contain inference rules for domain-specific reasoning, unless rigorously formalized.

While these content repositories assign a semantic reading in terms of concepts to single terms or phrases, the benefits of syntactic analysis open further opportunities for semantic interpretation (see Section 2.4.2). Typically, content relations between concepts are linguistically mediated through verbal (also nominal or adjectival) expressions, such as *X inhibits Y* or *the inhibition of X by Y*. At the semantic level, these relations are lexically represented as predicates with a corresponding argument frame, as in 'Inhibit(X, Y)', where we even might want to semantically constrain "X" to 'Substance' and "Y" to 'Growth Process'.[2] A semantic interpretation of a sentence such as *Interferon inhibits cell growth* might then lead to the proposition 'Inhibit(Interferon, Cell Growth)'. This process is based on structural information from the preceding syntactic analysis; that is, by properly relating the subject (*Interferon*) and direct object (*cell growth*) of an active-voiced clause as the 'Acting Agent' and the 'Affected Object' of an 'Inhibition Activity.' Once we have a fact base of such relations, this, in turn, may feed a semantic inference mechanism, which tries to derive new relations between entirely isolated factual assertions.

With these considerations in mind, linguistic knowledge resources (e.g., the lexicon) and processes making use of them (e.g., morphological analysis, chunking, parsing, and so forth) have to be properly synchronized. In Section 2.5, we deal with the organization of these repositories and computational threads in terms of a comprehensive pipeline architecture for text mining processes for biology, and look at two exemplary prototype systems, GeneWays and Protein Active Site Template Acquisition (PASTA).

1. There is a long debate still going on whether lexicons should just provide a link to a concept identifier, which points into another encapsulated component where conceptual reasoning takes place on the basis of conceptual relations and inference rules (e.g., an ontology or knowledge base); or whether the linguistic lexicon should be extended to cover language-specific semantic, as well as language-independent conceptual relations.
2. Italicized items refer to textual strings and lexical items. Items within single quotation marks, with initial capital letters, refer to concept labels, semantic roles, predicates, and other formal objects.

2.2 The Lexical Level of Natural Language Processing

We now look at the lexical level. At this level, we deal with how words and other tokens can be recognized, analyzed, and formally characterized to enable further processing.

2.2.1 Tokenization

Before any linguistic analysis can take place, be it morphological, syntactic, or semantic, the basic tokens involved (e.g., words, acronyms, abbreviations, numbers, punctuation symbols, and so forth) have to be identified. Tokenization, which is segmenting the input text character stream into linguistically plausible units (i.e., tokens), is an elementary, although often understudied, preprocessing step in any natural language processing system dealing with real-world input. The simplest strategy would be to tokenize text on white spaces and punctuation symbols. Such a simple heuristic, however, easily runs into severe problems.

- *Abbreviations.* Words are not always separated from other tokens by white space. A period also may signal an abbreviation (e.g., *Vol.* or *etc.*), in which case it has to be distinguished from a sentence-delimiting period. The problems become even more severe if such items appear at the end of a sentence (*The study was conducted within the U.S.*), as the full stop is here performing double duty. This may interfere with sentence boundary detection (see below).
- *Apostrophes.* Where a clitic is manifested as an apostrophe plus a sequence of one or more letters, such as the English possessive marker's in *IL-10's cytokine synthesis inhibitory activity*, it needs to be separated from its harboring word (*IL-10*), because it denotes a linguistically meaningful relationship between two entities (here, *inhibitory activity* of *IL-10*).
- *Hyphenation.* It is not always clear whether a tokenizer should return one or more tokens for hyphenated words (*text-based*), especially when authors are not consistent with hyphenation (*co-operate* versus *cooperate*).
- *Multiple Formats.* Numbers may occur in multiple formats containing ambiguous separators (*464,285.23* and *464285.23*). Other entities with multiple formats are dates, phone numbers, addresses, and so forth.
- *Sentence boundary detection.* Sentences usually are demarcated by the typical sentence-delimiting punctuation marks (e.g., period, exclamation mark, or question mark). Sometimes, however, other punctuation

symbols also serve to indicate sentence boundaries (e.g., colon, semicolon, or M-dash).

Typically, the solutions employed for handling tokenization problems rely on finite-state regular expression matching, lexicon-based approaches, or a mixture of both. Grefenstette and Tapanainen [1] propose a regular expression grammar for English to recognize numbers and dates. To address the problem of abbreviations and sentence boundary detection, they construct an ordered filter, which separates abbreviations from punctuation markers by means of a lexicon and the spelling information of the token in question. A more elaborate (but computationally more expensive) method is proposed by Palmer and Hearst [2], which takes into account the POS probabilities of the tokens around a punctuation marker, and feeds them into a neural network.

In the biological domain, tokenization poses additional challenges due to domain-specific terminology, and nonstandard punctuation and orthographic patterns (e.g., *an alpha-galactosyl-1,4-beta-galactosyl-specific adhesin* or *the free cortisol fractions were 4.53 +/− 0.15% and 8.16 +/− 0.23%*). Arens [3] points out that, in various MEDLINE publications, the same protein is referred to as *NF-kappaB*, *NF-kappa B*, and *NF-kappa-B*. Thus, normalizing such tokens to a canonical form becomes an additional task (see Chapters 4 and 5). Arens casts this as a machine learning problem, in which various punctuation and orthographic features are fed into a decision-tree classifier.

As another concrete example, we take the following sentence from a MEDLINE abstract in the Gene Expression Information System for Human Analysis (GENIA) corpus (see Chapter 8 and [4]):

> We here report that interferon inhibits growth and survival of NB4 APL cells in cooperation with RA.

Here, the acronyms *NB4* and *APL* come as two different tokens. A search in Google, however, also yields the two tokens concatenated by a hyphen (i.e., as *NB4-APL*). Both representations make sense from a biological point of view, because *NB4* is a cell line that was derived from *acute promyelocytic leukemia (APL)* cells. From a natural language processing perspective, however, it is essential that both representations, if found in running text, are tokenized in a consistent and unified way. Moreover, this example also highlights the importance of acronym detection (see Chapters 4 and 5), because *APL* is not only an acronym for *acute promyelocytic leukemia*, but also for *antiphospholipid syndrome*, which is an autoimmune disease. Similarly, in this example, the acronym *RA* must be mapped to the full term *retinoic acid*, but other potential biomedical candidates could be *retrograde amnesia*, *refractory anemia*, or *rheumatoid arthritis*.

2.2.2 Morphological Analysis

The purpose of morphological analysis is to link heterogeneous surface variants of a lexical element to its canonical base form. While only few lexical items do not undergo inflection (e.g., prepositions, adjectives, or conjunctions), most content-bearing lexical elements (e.g., verbs and nouns) do. Variation is mainly due to (syntactically required) inflection [*activat-es*, *activat-ed*, *activat-ing* (the gerund reading)], or derivation [*activat-ing* (the adjectival reading), *activat-ion*]. From these examples, one might hypothesize *activat* as a canonical base form for the four morphological variants just listed. Since inflectional processes do not alter the core meaning of the underlying base form, and derivational processes only slightly do so, reducing these variants to a canonical form provides a simple means to unify the content description of a document, just at the lexical level alone.

In order to achieve this goal, two basic approaches have been devised up until now. They can be distinguished according to whether or not they refer to a background lexicon, against which variants are matched. Among the lexicon-free approaches, the Porter algorithm [5] constitutes the most successful and most frequently used approach. Originally developed for information retrieval applications, it builds on a list of inflectional and derivational suffixes (morphologically recurrent character strings such as *es*, *ed*, *ing*, *ion*, *ly*, and so forth), and iteratively matches incoming strings from right to left, based on longest match. For example, if *ion* and *ation* were both available for matching *activation*, then the Porter algorithm would generate *activ* rather than *activat* as a stemming result. However, there are some constraints to which these reduced forms must conform; for example, length or character restrictions, to block, say, the generation of *n* caused by stripping *ation* from the input word *nation*. In essence, then, the Porter algorithm builds on a limited number of character sequences indicative of inflection and derivation, and hypothesizes, after stripping them off, the remainder (i.e., the variant's prefix) to be a canonical morphological form. Despite its simplicity, the Porter algorithm has turned out to be a standard morphological analyzer in the bag-of-words domain (i.e., the information retrieval community that determines relevant documents based on term vector–based similarity measures). Porter's stemmer is not adequate, however, for any linguistically more sophisticated approach that builds on rich(er) grammatical and lexical information.

This is exactly the dividing line between Porter-style lexicon-free approaches to morphological analysis, and those approaches that require a lexicon. Again, two approaches must be distinguished. On the one hand, there are full-form lexicons that explicitly enumerate all morphological variants plus their grammatical features, so that morphological "analysis" boils down to a simple lexicon look-up. On the other hand, there are lexicons composed of canonical

base forms, together with a list of morphologically relevant substrings (e.g., suffixes for inflection and derivation), like those mentioned above. Various forms of morphological analyzers then combine both resources, such that, once a morphologically relevant substring is found and removed from token input, the remainder is checked to see whether it exists in the lexicon as a canonical entry. If not, alternative segmentations are tried, until an entry is identified in the lexicon, or segmentation is precluded. A very simple, though unwarranted, reason may be that the canonical form searched for is not listed in the lexicon.

This field is full of a large variety of tools that can be picked up off-the-shelf. The methodologically most advanced approach is constituted by Two-level Morphology [6], a finite-state–based approach that applies a series of transducers, all operating in parallel, to input tokens. Each of these transducers accounts for a single morphological phenomenon (e.g., the change from *y* to *ies*, as with *fl-y* and *fl-ies*). However, this approach has, up until now, not received much attention for text mining in the field of biology.

2.2.3 Linguistic Lexicons

A (computational) lexicon contains the lexical elements (either as full forms or as canonical base forms), together with additional linguistic information about them, which is required for further morphological, syntactic, and semantic processing. Lexicons are not fully standardized with regard to the additional information they contain, since this is often task- (and theory-) dependent. However, we here enumerate the most prominent types of linguistic information located in such resources (see also Chapter 3).

A primary type of information is the POS category that can be assigned to a lexical entry. This requires that the form of the canonical entry coincides with common divisions into these categories. For example, the above-mentioned base form *activat* would be unreasonable, since it cannot be assigned a standard part-of-speech category, while *activate* (with POS tag Verb), *activation* (with POS tag Noun), or *active* (with POS tag Adjective) fit well into this categorization scheme. Note that POS information is crucial for any sort of syntactic processing, such as chunking and parsing (see Section 2.3). If no such processing is anticipated, this information need not be supplied.

Depending on the various POS types, more refined morphological information may be specified. For instance, specifications for nouns might contain information about grammatical number (e.g., irregular nouns, like singular: *nucleus*, plural: *nuclei,* or singular: *sheep*, plural: *sheep*), or grammatical case (e.g., irregular cases, like genitive-singular: *nucleus'*, or genitive-plural: *nuclei's*). While nouns are associated with case, such a category cannot be assigned to verbs, which instead share additional information about grammatical tense (e.g.,

irregular tenses, like past tense for *take*: *took*, or past participle of *go*: *gone*). Note that the categories we mention for each POS category are not meant to be complete.

While this kind of information is still strictly lexically oriented, additional information about the canonical form relates to descriptions of syntactically relevant (valid) contexts in which the morphologically modified canonical form may occur, or in which a grammatically correct phrase or clause is required to be formed. Depending on the theoretical framework of the grammar chosen, this may lead to specifications as to whether some verb is transitive (i.e., always requiring a direct object, such as *to inhibit [something]* or *to transcribe [something]*), or intransitive (i.e., not always, or not at all, requiring a direct object, such as *to compute ([something])* or *to occur*). These specifications are intended to express the syntactic constraint, whether or not a noun phrase following the particular verb is mandatory.

Even more sophisticated (and more theory-dependent) are subcategorization frames, which specify the kinds of syntactic satellites around a verb (such as '[NP __ NP (on)to NP]' for the verb *to map* (the position of which is indicated by the underscore symbol). This frame says that, not only is the verb transitive (indicated by the first NP symbol following the underscore), but it also puts a special lexical constraint on the preposition preceding the final noun phrase).

As an example, the UMLS Specialist Lexicon [7] contains such subcategorization information. Nevertheless, for text mining applications in biology, it faces two problems. First, although its coverage of the medical domain is impressive (approximately 257,000 basic entries with POS, morphosyntactic, and syntactic information), it does not adequately cover terms from the biology domain. Second, all of its verbal (and nominal) subcategorization frames are constructed manually, and thus lack the coverage that is required to fully capture biomedical sublanguage properties, which are different from general-language properties (see Section 2.4.2 on verb frames). Thus, a comprehensive computational lexicon covering major portions of relevant words and terminology from the biology domain is still a desideratum for text mining systems.

The third major type of information that can be linked to lexical specifications is semantic information, which we will introduce in Section 2.4.1. Note also that much domain-specific lexical material that is not part of the standard terminology in the field (e.g., names of proteins, genes, and companies) hardly can be captured by a linguistic lexicon (or by terminologies), because of its diversity, instability, and speed of change. These entities are usually continuously tracked by named entity recognition systems (see Chapter 6).

2.3 The Syntactic Level of Natural Language Processing

Sequences of words from a single sentence may be grouped syntactically into larger units, such as phrases or clauses. While it may well be possible to compute the syntactic structure of a phrase, it is often the case that syntactic analysis cannot completely be extended to the entire sentence to achieve a full parse, because of the lack of lexical or grammatical specifications. Hence, partial parsing is what we most likely can achieve under requirements for robust and efficient parsing. The knowledge behind the assignment of syntactic structure resides in two, usually complementary, resources—grammars and treebanks.

Grammars are explicit linguistic descriptions, usually in the form of rules or constraints, which characterize well-formedness conditions of (morpho)-syntactic categories (e.g., POS tags, features), or of nonterminal grammar categories, such as noun phrases (NP), prepositional phrases (PP), and so forth. Current language engineering efforts tend to avoid the use of highly expressive and computationally expensive grammar systems (e.g., those based on unification), in favor of far less expressive though tractable formalisms (e.g., those based on finite-state technology). However, full-grammar approaches aiming at a complete syntactic analysis are pursued in the field of biology [8–11], since the richer information they provide may be necessary to deliver the type of results required.

In contrast, many text mining applications in biology were (and some still are) characterized by the approximation of grammatical regularities via ad hoc pattern matching rules [12–15]. Such approaches, while they rapidly achieve limited benefits, usually fail to scale up for really large and diverse document collections. The main reason is that their rule specifications tend to become increasingly complex and harder to control and maintain.

The second type of resource consists of treebanks. These are corpora of plain texts, for which human annotators have supplied syntactic annotations at the sentence level (usually with POS tags and syntactic analysis structures, as described in Chapter 8). However, these annotations require a clear commitment to some sort of grammar theory (either constituent-based or dependency-based), and, particularly important for the biology domain, the capability of the annotators to understand the contents of the underlying documents. An advantage of this work is that grammar rules need not be explicitly spelled out, but can be learned automatically from these positive examples. However, large volumes of text have to be annotated by several (at least three) human annotators, in order to determine their annotation quality in terms of mutual consistency and quality. For the general newspaper domain, the Penn TreeBank [16] contains a reasonable quantity of POS and syntactic structure annotations for this learning task (typically, 1.2 million POS-tag-annotated tokens are used for POS training and testing, and 300,000 syntactically annotated tokens are used

for syntactic training and testing). For the biology domain, the GENIA TreeBank [4] (see Chapter 8) provides a sufficient number of POS annotations for learning purposes (500,000 tags), while its syntactic structure annotations are still under development.

2.3.1 Part-of-Speech Tagging

A large number of current NLP systems use POS taggers for the purpose of categorical syntactic disambiguation. This kind of preprocessing step is essential to cope with the various forms of lexico-syntactic ambiguities of words. For example, the word *report* can be either a common noun or a verb, depending on its syntactic context. In our example sentence (see Table 2.1), it is given the POS tag 'VBP' (verbal base form in the present tense), which, however, cannot be decided by simply doing morphological analysis with a (usually incomplete) computational lexicon (see Section 2.2.3). Rather, the tagging process assigns a POS tag to each token/word in the input, depending on its local syntactic context. Thus, most taggers are representatives of supervised and data-driven approaches, which crucially rely on training data from syntactically (here, POS) annotated corpora (see Chapter 8). There are two basic methods for POS tagging.

1. *Rule-based taggers.* For instance, given a small set of lexical and contextual rule templates, Brill's tagger [17] learns linguistic rules based on a transformation-based error-driven search algorithm. By this we mean that the learner has access to both a set of lexical and contextual rules, and an annotated corpus. After an initial seed tagging, learning proceeds by iteratively proposing rules, comparing the results of this rule application to the annotated POS corpus, assessing the mistakes still made, transforming the rules to avoid the encountered mistakes, and, eventually, ranking them in an application order.
2. *Statistical taggers.* The currently prevailing approaches to statistical tagging are n-gram [18], maximum entropy [19], and support vector machine (SVM) [20] models. TnT [18] implements the Viterbi algorithm for second-order Markov n-gram models, in which states of the model represent tags, and the output represents words. The best POS tag for a given word is determined by the highest probability that it occurs with n previous tags. Tags for unknown words are assigned by a probabilistic suffix analysis, and then smoothing is done by linear interpolation.

Until recently (see, however, Chapter 8), the development of annotated corpora has almost exclusively focused on the general-language newspaper domain, with the Penn TreeBank [16] being the most prominent example.

Thus, all standard POS taggers are parameterized according to the newspaper language domain. Porting these to sublanguage domains, such as biological text data, is accompanied by a noticeable loss in performance [21]. Retraining such taggers on annotated biomedical language corpora, in contrast, yields a performance boost beyond state-of-the-art figures [21]. Because subsequent syntactic processing steps (see the following sections) typically depend on the tagger's output, high performance at this level of processing is crucial for success in later stages. The second column of Table 2.1 shows our example sentence from above, with POS tags from the GENIA corpus.

2.3.2 Chunking

Chunkers exploit both lexical and POS sequence information to identify special phrasal units, such as noun, preposition, verb, or adjective phrases. Research in this area has mostly focused on two types of chunking:

1. *Base NP chunking* [22] (see also the third column in Table 2.1), in which "Base NPs" are defined as nonrecursive noun phrases, which end after their nominal head, and exclude any type of postmodification (e.g., prepositional phrases, genitives). For instance, in the annotated utterance *[[the synthesis]$_{NP\text{-}base}$ of [long enhancer transcripts]$_{NP\text{-}base}$]$_{NP\text{-}complex}$* two base NPs are linked by the preposition *of*, and thus form one complex NP.
2. *Text chunking* [23] is a useful preprocessing step for parsing (see Section 2.3.3). It consists of dividing a text into phrases, in such a way that syntactically related words become members of the same (nonoverlapping) phrases. Besides NPs, verbal phrases (VPs) and predicate adjectival phrases (ADJP-PRD) are recognized, as well as PPs and subordinate clause markers (SBAR) (see the fourth column in Table 2.1 for examples).

Both types of chunking rely on the availability of corpora annotated with chunks for use as training material. In such annotated corpora, phrasal entities are typically marked using the standard Inside/Outside/Begin (IOB) chunk representation (i.e., I = current token is inside a chunk; O = current token is outside of any chunk; B = current token is the beginning of a chunk, immediately following another chunk). Extracting the relevant phrasal units from syntactically annotated treebanks is usually done automatically.

There are several statistical and machine learning–based chunkers that exploit this type of annotated information. The techniques range from rule-based learners [22], to hidden Markov models (HMMs) [24], to kernel-based SVMs [25]. As at the POS tagging level, however, these tools are only

Table 2.1
Standard POS Tag Notation and IOB Chunk Tag Notation for NP and Text Chunking, for Our Example Sentence

Tokens	POS Tags	Base NP Chunks	Text Chunks
We	PRP	I-NP	B-NP
here	RB	O	O
report	VBP	O	B-VP
that	IN	O	B-SBAR
interferon	NN	I-NP	B-NP
inhibits	VBZ	O	B-VP
growth	NN	I-NP	B-NP
and	CC	O	O
survival	NN	I-NP	B-NP
of	IN	I-NP	B-PP
NB4	NN	I-NP	B-NP
ctlparAPL	NN	O	I-NP
cells	NNS	I-NP	I-NP
in	IN	O	B-PP
cooperation	NN	I-NP	B-NP
with	IN	O	B-PP
RA	NN	I-NP	B-NP
.	.	O	O

trained/parameterized on general language, and to exploit their full potential, retraining on annotated biological corpora may be essential (see Chapter 8).

Syntactic processing at the NP chunking level has already proven beneficial for some crucial tasks in biological text mining, such as terminology management (see Chapter 4) and named entity recognition (see Chapter 6), since most terms and named entities are contained within noun or prepositional phrases. The output of text chunking also is useful for the relation mining task. Not only can noun phrases be identified as entity markers, but also verbal and adjectival predicates (e.g., *activates*, *inhibited*, and so forth), acting as explicit relation markers, can be identified within verbal and adjectival phrases.

2.3.3 Parsing

Another useful building block for syntactic analysis is the identification of clauses (i.e., word sequences that contain a subject and a predicate). For example, the following clauses can be obtained from our example sentence. Here, "(S" marks the beginning, and ")" the end of a clausal unit (such as an embedded "that-clause" here):

(S *We here report*

(S-BAR *that interferon inhibits growth and survival of NB4 APL cells in cooperation with RA*
)

.)

Again, there are various machine learning–based methods, such as the ones mentioned above for chunking, which can be employed for the clause identification step (see [26] for detailed descriptions of the tasks and techniques currently used).

Finally, the most elaborated syntactic analysis is full sentence parsing. Ideally, the previously partially analyzed syntactic chunks and clauses are brought into high-level syntactic relationships with one another, both across and within phrase boundaries. Although desirable from a linguistic point of view, full parsing has had a rather limited benefit for large-scale text mining applications, up until recently. This is due to the complexity of the parsing task, caused by the inherent ambiguity of natural language. However, there are studies that make use of parsers to analyze biological text corpora, such as in [10], which extracts gene pathway relations using the Arizona Relation Parser, a parser that uses a rule set geared toward the biological domain (see Chapter 7). Another full parser made available through the GENIA Consortium is Enju [9, 27], a probabilistic Head-driven Phrase Structure Grammar (HPSG) parser, which, however, is trained on general-language newspaper text (i.e., the Penn TreeBank [16]).

2.4 The Semantic Level of Natural Language Processing

2.4.1 Lexical Semantic Interpretation

In Section 2.2.3, we introduced two major types of information that are usually related to lexical entries (i.e., morphological and syntactic information). We turn here to a third type of information that can be linked to lexical specifications (i.e., semantic information) (see also Chapter 3). Semantic information may come in two forms. Reconsidering the types of syntactic contextual information we have

already introduced in Section 2.2.3, we might, for example, wish not just to specify *to inhibit [something]* or *to transcribe [something]* (indicating that both are transitive verbs), but rather we might wish to specify *to inhibit [Process]* or *to transcribe [Nucleic Acid]*. In other words, we refine the direct object from a semantic (i.e., content-oriented) perspective. From a processing point of view, such refined constraints prevent a syntactically admissible, though semantically invalid, analysis, such as *to inhibit amino acids* or *to transcribe cell growth*, from being computed. Syntactically, both are admitted as direct objects. These kinds of constraints are usually referred to as "selectional restrictions."

Finally, even more semantic considerations crop up with lexico-semantic relations. Consider the taxonomic generalization relation between 'Cell' and 'Biological Substance' or the part-whole relation between 'Cell Nucleus' and 'Cell'. This type of information is concerned with the lexical meaning of canonical entries, and has no dependence at all on any syntactic considerations.

The consideration of such relations, together with their algebraic properties (e.g., symmetry, transitivity), means that we are crossing a borderline. This borderline separates, on the one hand, language-specific knowledge (e.g., morphosyntactic and POS categories, subcategorization frames, and selectional restrictions), usually assigned to linguistic lexicons (see Section 2.2.3), from, on the other hand, language-neutral, yet domain-specific, concept systems. If these relational considerations are left implicit (mostly carried out by humans using such a concept system), then we call them "terminologies." If, however, they are formalized, so that taxonomic or partonomic inferences can be automatically drawn, then we call them "ontologies." As an example of taxonomic reasoning, we may transitively conclude from:

'Cell' Is-A 'Biological Substance', and
'Biological Substance' Is-A 'Matter', that
'Cell' Is-A 'Matter'.

Similarly (though definitely not in the same way), we may deal with partonomic reasoning, assuming:

'Cell Nucleus' Part-of 'Cell', and
'Cell' Part-of 'Human Body', in order to derive
'Cell Nucleus' Part-of 'Human Body'.

The UMLS Metathesaurus (see also Chapter 3) provides substantial coverage of (mostly) medical English terminology, and contains approximately 1 million entries (canonical forms plus variants) from almost 100 source terminologies whose concepts are semantically linked via the UMLS Semantic Network [28]. Although already a viable resource for biologists, it poses two problems for

large-scale text mining applications in biology. First, there is no explicit link between the linguistic information in the UMLS Specialist Lexicon [7] and the semantic information in the UMLS Metathesaurus. This means that one cannot use the Specialist Lexicon to look up a word in the lexicon and then get a link from the lexicon to the concept in the UMLS Metathesaurus.

Second, although the UMLS terminologies excel on lexico-semantic information, by having a large variety of shallow as well as deep semantic relations, their coverage of biology (as opposed to medicine) is still quite limited. This is despite the fact that the biologically relevant Gene Ontology (GO) [29] recently has been incorporated. The same observation holds for the Obo framework (http://obo.sourceforge.net), a collection of terminological resources exclusively from the field of biology, which is highly fragmentary compared to the remarkable coverage (for medicine) provided by the UMLS. We conclude that text mining in biology is characterized by a lack of interlinked, high-coverage, and domain-specific lexical resources. In contrast, while partial (i.e., subdomain-specific) biological terminologies with lexico-semantic specifications are available, they all lack morphological, morpho-syntactic, or purely syntactic information, which crucially limits their usability for text mining applications.

Concept systems of biology undergo considerable changes, so that mechanisms for automatic lexical acquisition, and concept learning, are desperately needed. Another main challenge related to the enormous rate of lexical dynamism in biology arises with shortcuts (i.e., abbreviations and acronyms) (see Chapter 5). The vast majority of approaches to this challenge focus on various ways to relate short forms to long forms, when they occur in close adjacency within sentences or paragraphs, within the same document. However, even more challenging unification problems arise when the same protein or gene is given different names in entirely different, totally unrelated documents, and no links indicating their sameness (or very close similarity) are explicitly or implicitly stated.

2.4.2 Semantic Interpretation of Utterances

Semantic interpretation of utterances deals with the analysis of the composite meaning of phrases, clauses, and sentences. At this level, semantic roles or arguments are associated with predicates. An instantiated predicate, taken together with its semantic roles, is usually referred to as a proposition. For example, the minimum set of semantic roles for a verb, such as *to inhibit* (with 'Inhibit' as its associated predicate), consists of an 'Agent' (an entity that inhibits) and a 'Patient' (an entity that undergoes inhibition). While these two are the most frequent semantic roles (see [30] for the quantitative distribution of different semantic roles), several others are discussed in the literature ('Experiencer', 'Instrument', 'Source', 'Goal', 'Location', or 'Time').

Just as with syntax, there are two basic approaches to identifying the semantic roles of a predicate in a sentence. The more traditional way is to use the output of a full sentence parser (see Section 2.3.3), and to apply a set of (manually created) mapping rules [31]. These rules specify certain syntactic relations, available from syntactic analysis plus lexical information about the mapping of lexical items to terms. If such a syntactic pattern is matched, then the rules map the semantic correlates to a predicate-argument frame that reflects syntactic dependencies at the semantic or conceptual level. Often, semantic interpretation is collapsed with the text mining step [32, 33], thus omitting any sort of assessment to filter out irrelevant (known, self-evident) propositions.

However, with the growing importance of more shallow methods for syntactic analysis (see Section 2.3.2), semantic role labeling has recently resorted to methods that only consider partial syntactic output [30].[3] Besides partial syntactic analysis, the second major training input to the methods reported in [30] comes in the form of verb argument structure annotation, as specified for general English in the PropBank annotation project [34]. Here, semantic corpus annotation is provided, both as standoff semantic role annotation to the syntactically annotated Penn TreeBank [16], and as a linked lexical resource in frame files for the various verbally expressed predicates found in this corpus. An example for the verbal predicate 'Inhibit' (for general-language English) is provided in Figure 2.1.

As can be seen, for general-language English, the predicate 'Inhibit' contains one role set;[4] that is, its meaning ('Restrain', or 'Suppress') is defined by two semantic roles labeled

n="1" ('Agent'), and
n="2" ('Entity Inhibited').

There are many verbal predicates with more than one meaning, and hence with more than one role set, such as the verbs *to call* or *to draw*. The meaning of this 'Inhibit' predicate is also described by a name gloss:

name="to restrain, suppress".

3. These are similar statistical and machine learning–based methods, as described in Section 2.3.2, such as transformation-based learning, memory-based learning, or support vector machines.

4. Role sets for verbs are determined on purely empirical grounds (i.e., through extensive corpus analysis). Moreover, nominal and adjectival predicates are not covered at the current stage of the PropBank project, although there is complementary ongoing work on nominal predicates at New York University (the NomBank project).

```
<!DOCTYPE frameset (View Source for full doctype...)>
−<frameset>
  −<predicate lemma="inhibit">
     <note>Frames file for 'inhibit' based on survey of sentences in the WSJ
       corpus.</notes>
    −<roleset id="inhibit.01" name="to restrain, suppress" vncls="—">
      </roles>
         <role n="0" descr="agent" />
         <role n="1" descr="entity inhibited" />
      </roles>
    −<example name="transitive">
       <inflection person="ns" tense="present" aspect="ns" voice="active"
         form="full" />
       <text>Corporate lawyers said [0] the new fees wouldn't inhibit
         many mergers or other transactions.</text>
       <arg n="0">the new fees</arg>
       <arg n="M" f="MOD">would</arg>
       <arg n="M" f="NEG">n't</arg>
       <rel>inhibit</rel>
       <arg n="1">many mergers or other transactions</arg>
      </example>
    </roleset>
  </predicate>
  <note>frames created by Olga</note>
</frameset>
```

Figure 2.1 Role set for the verbal predicate 'Inhibit' in the frame file of PropBank.

While this is correct from a general-language point of view, it is not applicable from a domain-specific biological point of view.

Furthermore, simply transferring these general-language verb frames to the biological domain is also problematic, because predicates may exhibit additional, or even different, role set properties in different domains. For example, a search in the GENIA corpus [4] (see also Chapter 8) for the predicate 'Inhibit' yields additional roles (marked here in bold):

- (*TCDD*)$_{\text{Agent}}$ *inhibits* (*murine and human B lymphocyte immunoglobulin production*)$_{\text{Entity-inhibited}}$ (*through an unknown mechanism*)$_{\textbf{Instrumental}}$.
- (*RA*)$_{\text{Agent}}$ *directly inhibits* (*the erythroid differentiation program*)$_{\text{Entity-inhibited}}$ (*at the level of early adult HPC*)$_{\textbf{Stage}}$.

Hence, a domain-specific redefinition of verb frames is necessary to capture the appropriate semantic roles for predicates in the biological domain.

In both approaches, it is crucial to map the output of syntactic analysis to a predicate-argument frame that reflects the relational dependencies between entities at the content level. This is especially important, because natural language

syntax provides several ways to express the same proposition. For example, our active-voice example sentence, from Table 2.1, could be easily expressed in the passive voice, without changing its propositional content:

> Growth and survival of NB4 APL cells is inhibited by interferon in cooperation with RA.

Nevertheless, both active and passive sentences should be mapped to the same predicate argument structure. Given our verb frame 'Inhibit(X, Y)', a simple proposition mapping may look like this:

> Inhibit (Interferon, (Grow (NB4 APL cells)))

It should be noted that this is not the entire propositional content of the sentence. In order to capture more of the propositional content, greater in-depth syntactic processing would have to be employed, in order to recognize the coordinated noun phrases (*growth and survival*) plus the cooperation of 'RA' in this 'Inhibit' relation. For this purpose, the meaning of the prepositional phrase (PP) *in cooperation with* would have to be mapped to 'Together-with', and the attachment site for the PP would have to be correctly disambiguated, so that the full propositional information can be derived:

> Inhibit (Together-with (Interferon, RA), (And (Survive (NB4 APL cells) Grow (NB4 APL cells)))

Whereas the information on 'Survive' is already quite desirable from a biological point of view, the information on the cooperation of 'Interferon' with 'RA' is fundamental, because without it, the contents of the sentence would not be biologically correct. This also can be inferred from the follow-up sentence:

> Interferons alone have minimal maturation effect on NB4 cells.

Here, a biologist knows that cell maturation leads to apoptosis (cell death). A minimal maturation effect cannot cause apoptosis, and interferons alone cannot inhibit the survival of NB4 APL cells. One can already imagine what kind of biological background knowledge a text mining system would need to make these types of inferences.

The predicate-argument frames described so far only list the entities involved, without abstracting the semantic type to which an entity or term belongs. Given the availability of a bio-ontology, taxonomic reasoning could infer that

interferon Is-a 'Protein',
RA Is-a 'Hormone',
NB4 APL cells Is-a 'Cell Line', and
growth and survival Is-a 'Process'.

Thus, the same predicate-argument frame with a much coarser granularity would look like the following. Note that abstraction levels should depend on the requirements of the subsequent processing steps, and the application in general.

Inhibit (Together-with (Protein, Hormone), (Process (Cell Line)))

If such semantic resources are provided, text mining applications in biology can produce some valuable results; for example, a list of relations between conceptual entities that are derived through the various levels of NLP analysis. In a relevance filtering step, such applications can turn to what text mining is actually all about—finding new relations from a set of already known relations [35].

2.5 Natural Language System Architecture for Text Mining

In the previous sections, we dealt with the decomposition of linguistic knowledge into various processing levels pertinent to text mining for biology. In this section, we pull the different strings together by looking at the architectural organization of single natural language processing components (see Figure 2.2). We then relate this general architecture to two representative systems, from the range of information extraction and text mining systems in the field of biology.

2.5.1 General Architecture

The starting point for analysis, for text mining systems in biology, is a collection of raw biology documents. Very often, these are abstracts taken from MEDLINE. Due to copyright restrictions, full texts, which contain much more detailed information than abstracts, are rarely processed [36]. However, this might fundamentally change in the future, and may have immediate effects on the architecture that we propose, particularly discourse-level textual phenomena.

These documents first undergo a formal cleansing process, in order to get rid of text formatting code and diverse annotation expressions (e.g., RTF, HTML, PDF, and so forth). The first linguistic processing level is concerned with lexical analysis. Here, the document is split into single tokens by the tokenizer. At the core of this level of analysis is the POS tagger, which assigns POS tags to text tokens. POS taggers usually have a built-in component for

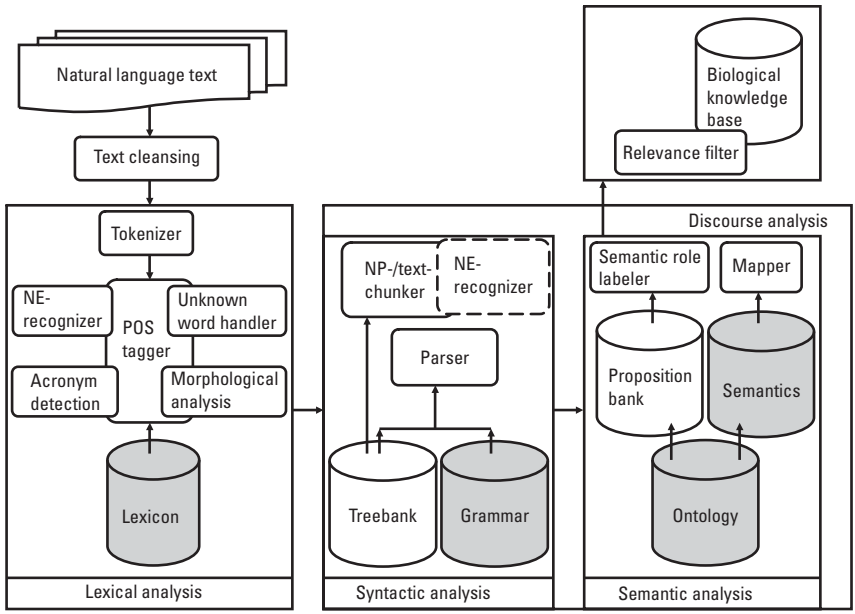


Figure 2.2 Natural language system architecture for text mining for biology.

some morphological analysis, to facilitate access to the corresponding information in a computational lexicon [i.e., the POS tag(s) for a specific token whose base form is a lexicon entry]. If there is a need at this stage to recognize special forms of names relevant to biology (e.g., abbreviations or domain-specific types of text tokens, such as numerical strings, measure units, biological names for proteins or genes that are composed of alphanumerical character combinations, and so forth), then a named entity recognizer (see Chapter 6) and possibly a module for acronym detection (see Chapter 5) are interleaved with the POS tagger. If the POS tagger still fails to provide an assignment of POS categories, another subcomponent is invoked, the "unknown word handler." Usually based on statistical considerations, a suffix analysis is carried out, and the most likely POS tag for the unknown word is guessed. The output of lexical analysis then consists of tuples, which are formed by text tokens and their associated POS (or even named entity) tag(s), together with a segmentation of the document into its constituent sentences.

Next, each sentence is submitted to syntactic analysis. For each sentence, the sequence of POS tags is grouped into linguistically plausible composite units, the phrasal chunks. Typically, a chunker splits a complex sentence into sequences of phrases. The phrases can be simple (e.g., containing base NPs), or increasingly complex (e.g., including verbal and prepositional phrases). Once

sets of chunks have been identified, a (partial) parser either may relate these (only sequentially ordered) chunks according to grammatical criteria (e.g., the distinction between the subject and the object of a sentence), or it may assign additional internal syntactic structures to the chunks. Chunking may precede parsing, although parsers are not necessarily dependent on previous chunking. It also may be the case that parsing is omitted, and only chunking results are made available to subsequent processes. Biological named entity recognition also can be performed at this stage by taking into account the results of partial syntactic analysis (e.g., NP chunking), since most biological named entities are linguistically expressed within noun phrases.

There are two basic resources one may envisage for chunking and parsing. The traditional approach to parsing builds on grammars that are usually manually supplied as the main linguistic resource for syntactic knowledge. This information is then used by standard parsing algorithms (such as Earley or Cocke/Kasami/Younger), in order to derive a syntactic representation of the sentence. Alternatively, we also may learn syntactic bracketing patterns from a gold standard corpora that is already annotated, and assign syntactic structures as learned from the gold standard to new text input. In this case, the grammar is essentially expressed in the annotated data. Note that human annotators must be trained in a particular grammar theory in order to produce proper annotations, and, at the same time, be competent in the biological domain to understand the documents they have to annotate. The use of annotations for training purposes is the technique that currently prevails in the field of human language technology, but due to the current lack of annotated resources (see Chapter 8), the former approach still dominates chunking/parsing applications within text mining settings for biology.

Once syntactic information is available in terms of sentence-level chunks or parses, semantic analysis starts. Whatever the semantic representation of choice may look like (e.g., a logic-style representation in terms of predicate-argument structures, or graph-style representation in the form of semantic networks, frames, or conceptual graphs), the task of semantic analysis consists of linking semantic entities (e.g., biological terms or concepts), in order to form some sort of proposition. The lexical and syntactic representational threads now may be combined.

First, lexical items at the document word level are mapped to a conceptual term level. That is, a lexical item is checked to see whether it might be a predicate or the argument of a predicate. This information usually is made available by lexicon look-up. Next, given a predicate and its n-ary argument structure, syntactic evidence is consulted, in order to decide which chunk or parse unit denotes a particular argument of the predicate. A semantic role labeler carries out this task. This may rely on either a semantics resource (i.e., semantic interpretation or mapping rules similar to human-made grammars for syntactic

analysis), or on the results of semantic learning processes, through which mappings from syntactic structures to propositions are acquired, based on propositions that are already annotated (i.e., based on human coding efforts, just as with treebanks for parsing). The division into rule-based systems and machine learning–based systems is the same as that encountered for syntactic analysis.

The delineation of an extra semantic level—in particular, the methodological level between predicates and their arguments—is often not so strictly drawn. Lexical items may be reorganized in predicate-style list form, to make them appear as if they constitute representational entities on their own. For example, *interferon* may not be checked to see if it actually belongs to the conceptual class 'Protein'. Whether the semantic level has a representational status on its own often is dependent on the requirements of subsequent processing, in particular at the conceptual level. Depending on the availability of such a representation system and a corresponding ontology of (parts of) the biology domain, the propositions derived from text analysis may not only be recorded in a biological knowledge base as a list of predicate argument propositions, but also may be subject to further reasoning processes. For example, such processes may check the implications of that new knowledge to derive additional biological knowledge, or they may test for biological consistency of the newly entered data. To the best of our knowledge, no biology text mining system currently incorporates such an advanced reasoning functionality.

In our architecture scheme, we have deliberately omitted a self-contained component for discourse analysis. This decision simply reflects the fact that almost no efforts previously have been devoted to this topic for biology text mining applications. This is true despite the observation that, in biology abstracts (and even more so in full texts), relations between sentences are expressed in terms of referential relations between entities and propositions, which are encoded by pronouns and definite noun phrases [37]. These relations increasingly take conceptual knowledge (e.g., knowledge about taxonomies, or parts and wholes) into account [38]. While referential relations are a major class of text cohesion phenomena, text coherence phenomena relate to the overall organization of a text in terms of rhetorical structures [39]. Although often disregarded in the text mining literature, the consideration of these text phenomena is required for complete and correct descriptions of the content that can be derived from a natural language document [38].

Finally, the semantic or conceptual structures that were derived from natural language documents need to be assessed by a "biological relevance filter," because not every proposition acquired from a text is interesting or new for biologists. Unfortunately, this final text mining step is often omitted (i.e., all results from linguistic analysis are directly passed to the prospective user), although it should be crucial for any text mining system to avoid overloading the bio-user with irrelevant propositions.

One should always bear in mind that the architecture that we have just sketched might be altered in many ways, and tuned according to the desired functionality. Nevertheless, the picture we have proposed combines current linguistic methods and techniques in a way required to build adequate text mining workflows for biology.

2.5.2 Two Concrete System Architectures

We now focus on the description of two exemplary prototypes for information extraction and text mining in the field of biology, GeneWays and PASTA (see also Chapter 7), in order to examine how the design of concrete systems reflects the architectural considerations we have introduced so far.

The GeneWays system [40, 41] is targeted at the identification of molecular interactions pertinent to signal-transduction pathways from full-text documents (rather than abstracts). First, it provides the standard battery of preprocessing routines (e.g., breaking up texts into sentences, and tokenizing sentences into single words and atomic multiword phrases), and lexical processing (e.g., POS tagging, named entity recognition and interpretation, morphological recognition, lexically oriented semantic disambiguation of homonyms and synonyms). A part of the resources used is an elaborated ontology of semantic types, by which the lexicon (which covers, among other categories, approximately 125 different verbs relevant for the field) is organized according to biological principles.

The syntactic processing module of GeneWays is based on a full grammar approach, rather than on the empirical chunking and parsing methodologies that were the subject of Section 2.3. Basically, a manually constructed and maintained Definite Clause Grammar (DCG) is applied, in order to derive a complete syntactic parse, if possible, from POS-tagged and named-entity-annotated tokens for each sentence. If only fragments are encountered due to incomplete grammar or lexical specifications, then the system switches into an error recovery repair mode, yielding a partial or shallow parse. The grammar of GeneWays is capable of dealing with a wide range of quite complicated nested utterances. The parser outputs semantically interpreted trees, in a special information format, which are imported into the system's knowledge base. GeneWays already has produced 3 million biology-related propositions from a total of 150,000 processed journal articles. However, this huge amount of biological knowledge is still redundant and conflicting, because no (automatic) filtering and curation component has been previously available. The developers of GeneWays also have taken a lot of care to provide visualization tools for accessing this vast amount of data.

The PASTA [42] system was developed and evaluated on a corpus of 1,500 MEDLINE abstracts (rather than full texts), and the system extracts information related to the roles of specific amino-acid residues in protein

molecules. After textual input has passed through a text preprocessor for section filtering (eliminating irrelevant parts of MEDLINE abstracts), lexical processing proceeds with sentence splitting, text tokenization, morphological analysis, lexicon look-up, and rule-based named entity recognition and interpretation. Syntactic processing is carried out by POS tagging and parsing with a manually built phrase structure grammar. The semantic analysis of parse structures leads to a propositional encoding, in terms of a predicate-argument representation that is similar to predicate logic. In the information extraction stage, much effort is devoted to determining textual coreference links. These are intended to account for reference relations between entities that occur in different sentences, but denote the same information items. Access to the biological knowledge extracted by PASTA is made available via displaying templates, using alphabetically ordered indices, color-coding by semantic classes, and hyperlinking to other documents containing the same term.

We have chosen both systems as representatives of the current state of the art in the field. Their architecture reveals that the methods underlying lexical analysis are hardly controversial. As far as syntactic processing is concerned, both systems have been developed using the explicit full grammar approach. Grammatical structures are interpreted, and then mapped to information extraction templates on the basis of manually designed and maintained rules. We assume that this approach increasingly will be replaced by learning-based approaches that rely on syntactically and semantically annotated corpora. PASTA, unlike GeneWays, already takes care of discourse phenomena, which will constitute another major avenue of research in the field as full texts become more accessible in the future. Such systems only are targeted to very specific subdomains of biology, such as the protein structure literature in the case of the PASTA system. One desideratum is to make these architectures more comprehensive, especially concerning their lexico-semantic coverage, in order to include bigger knowledge portions from the vast field of molecular biology. The area in need of most improvement may be the field of information curation [i.e., (automatically) filtering out redundant and conflicting information], as well as the determination of the really new and relevant information contained in the source documents.

2.6 Conclusions and Outlook

In this chapter, we distinguish three main levels of linguistic analysis—lexical, syntactic, and semantic—all of which have to be properly orchestrated to set up a linguistically based text mining system for biology. Despite the progress that has been made, the combination of the different modules of natural language text analysis is still more an art than an engineering discipline. The workflow we

present in Figure 2.2 provides an idealized view of system building for text mining in the field of biology, which has to be complemented by many heuristic solutions.

Text mining is still difficult, due to the inherent properties of the structure of language and methodologies that are not fully adequate. From a structural perspective, language analysis struggles with the enormous potential for ambiguity at all levels of analysis. The problem of scaling-up text mining and natural language systems (i.e., managing the growth of linguistic knowledge resources so that the quality of the functionality also increases), only can be tackled through a clear commitment towards acquiring further linguistic knowledge on the basis of machine learning methods. This applies to all levels of analysis—lexical, syntactic, and semantic—as well as to the learning of information extraction and text mining rules [43, 44].

Accordingly, a major methodological requirement for further progress inevitably will be the provision of sufficiently-sized, quality-checked annotated corpora. While the GENIA corpus (see Chapter 8) fulfills this requirement, as far as POS and named entity annotations are concerned, only limited resources are provided as far as syntactic and, in particular, propositional encodings are concerned. See recent progress on propositional annotation of MEDLINE abstracts reported by [45]. A serious account of discourse-level annotations is also lacking, both at the level of anaphoric reference relations (but see recent work on coreference relations [37]), and at the level of text macrostructures (but see recent work on zoning [39]).

The issue of propositional knowledge for biology is further complicated by the lack of high-coverage terminological and ontological resources. These are important, because any propositional annotation (even seemingly simple named entity assignments) makes implicit use of some form of terminological and ontological commitments.

Only few efforts are currently directed at filtering the results from text analysis. Notions such as interestingness, newsworthiness, and originality, are still awaiting concise and rigid formalization. If such metrics were available, then users in biology domains would have access to relevance-approved text mining results.

Further progress is most likely to come from international competition series [similar to the Text Retrieval Conference (TREC) for document retrieval, and the MUC for information extraction research], hosted by well-established biological institutions (e.g., EMBL, EBI, and so forth) This process has already started [e.g., the task definitions at Knowledge Discovery and Data Mining (KDD) Cup 2002, BioCreAtIvE, and the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications (JNLPBA) Symposium]. Such events will further strengthen the capabilities of text mining systems, due to their strict adherence to evaluation methods and standards (see

Chapter 9). The increasing recognition of text mining in biology as a competitive edge technology is reflected by the creation of the U.K. National Centre for Text Mining, and by similar efforts at the National Library of Medicine (NLM) in the United States and in Japan), and will further stimulate future efforts in the field.

References

[1] Grefenstette, G., and P. Tapanainen, "What Is a Word, What Is a Sentence? Problems of Tokenization," *Proc. 3rd Int. Conf. on Computational Lexicography*, Budapest, Hungary, July 7–10, 1994, pp. 79–87.

[2] Palmer, D. D., and M. A. Hearst, "Adaptive Multilingual Sentence Boundary Disambiguation," *Computational Linguistics*, Vol. 23, No. 2, 1997, pp. 241–267.

[3] Arens, R., "A Preliminary Look into the Use of Named Entity Information for Bioscience Text Tokenization," *Proc. HLT-NAACL Student Research Workshop*, Boston, MA, May 2, 2004, pp. 37–42.

[4] Ohta, T., Y. Tateisi, and J.-D. Kim, "The GENIA Corpus: An Annotated Research Abstract Corpus in Molecular Biology Domain," *Proc. 2nd Int. Conf. on Human Language Technology Research (HLT 2002)*, San Diego, CA, March 24–27, 2002, pp. 82–86.

[5] Porter, M. F., "An Algorithm for Suffix Stripping," *Program*, Vol. 14, No. 3, 1980, pp. 130–137.

[6] Sproat, R., *Morphology and Computation*, Cambridge, MA: MIT Press, 1992.

[7] Browne, A. C., et al., *Lexical Systems: A Report to the Board of Scientific Counselors*, Bethesda, MD: U.S. National Library of Medicine, Lister Hill National Center for Biomedical Communications, September 2003 (LHNCBC-TR-2003-003).

[8] Park, J. C., "Using Combinatory Categorial Grammar to Extract Biomedical Information," *IEEE Intelligent Systems*, Vol. 16, No. 1, 2001, pp. 62–67.

[9] Yakushiji, A., et al., "Event Extraction from Biomedical Papers Using a Full Parser," *Proc. Pacific Symp. on Biocomputing (PSB 2001)*, Kauai, HI, January 3–7, 2001, pp. 408–419.

[10] McDonald, D. M., et al., "Extracting Gene Pathway Relations Using a Hybrid Grammar: The Arizona Relation Parser," *Bioinformatics*, Vol. 20, No. 18, 2004, pp. 3370–3378.

[11] Daraselia, N., et al., "Extracting Human Protein Interactions from MEDLINE Using a Full-Sentence Parser," *Bioinformatics*, Vol. 20, No. 5, 2004, pp. 604–611.

[12] Blaschke C., J. C. Oliveros, and A. Valencia, "Mining Functional Information Associated to Expression Arrays," *Functional and Integrative Genomics*, Vol. 1, No. 4, 2001, pp. 256–268.

[13] Horn, F., A. L. Lau, and F. E. Cohen, "Automated Extraction of Mutation Data from the Literature: Application of MuteXt to G Protein-Coupled Receptors and Nuclear Hormone Receptors," *Bioinformatics*, Vol. 20, No. 4, 2004, pp. 557–568.

[14] Leroy, G., H. Chen, and J. D. Martinez, "A Shallow Parser based on Closed-Class Words to Capture Relations in Biomedical Text," *Journal of Biomedical Informatics*, Vol. 36, No. 3, 2003, pp. 145–158.

[15] Huang, M., et al., "Discovering Patterns to Extract Protein-Protein Interactions from Full Texts," *Bioinformatics*, Vol. 20, No. 18, pp. 3604–3612.

[16] Marcus, M. P., B. Santorini, and M. A. Marcinkiewicz, "Building a Large Annotated Corpus of English: The Penn TreeBank," *Computational Linguistics*, Vol. 19, No. 2, 1993, pp. 313–330.

[17] Brill, E., "Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part-of-Speech Tagging," *Computational Linguistics*, Vol. 21, No. 4, 1995, pp. 543–565.

[18] Brants, T., "TnT: A Statistical Part-of-Speech Tagger," *Proc. 6th Conf. on Applied Natural Language Processing (ANLP 2000)*, Seattle, WA, 2000, pp. 224–231.

[19] Ratnaparkhi, A., "A Maximum Entropy Part-of-Speech Tagger," *Proc. 1st Conf. on Empirical Methods in Natural Language Processing (EMNLP'96)*, Philadelphia, PA, May 17–18, 1996, pp. 133–141.

[20] Giménez, J., and L. Márquez, "Fast and Accurate Part-of-Speech Tagging," *Proc. Int. Conf. on Recent Advances in Natural Language Processing (RANLP 2003)*, Borovets, Bulgaria, 2003, pp.158–165.

[21] Hahn, U., and J. Wermter, "High-Performance Tagging on Medical Texts," *Proc. 20th Intl. Conf. on Computational Linguistics (COLING 2004)*, Geneva, Switzerland, August 23–27, 2004, pp. 973–979.

[22] Ramshaw, R., and M. P. Marcus, "Text Chunking Using Transformation-Based Learning," *Proc. 3rd ACL Workshop on Very Large Corpora*, Cambridge, MA, June 30, 1995, pp. 82–94.

[23] Sang, E. F. T. K., and S. Buchholz, "Introduction to the CoNLL-2000 Shared Task: Chunking," *Proc. CoNLL-2000 and LLL-2000*, Lisbon, Portugal, 2000, pp. 53–57.

[24] Molina, A., and F. Pla, "Shallow Parsing Using Specialized HMMs," *Journal of Machine Learning Research*, Vol. 2, No. 4, 2002, pp. 595–613.

[25] Kudo, T., and Y. Matsumoto, "Chunking with Support Vector Machines," *Proc. 2nd Meeting of the North American Chapter of the Association for Computational Linguistics (NACCL'01)*, Pittsburgh, PA, 2001, pp. 192–199.

[26] Sang, E. F. T. K, and H. Déjean, "Introduction to the CoNLL-2001 Shared Task: Clause Identification," *Proc. CoNLL-2001*, Toulouse, France, 2001, pp. 53–57.

[27] Miyao, Y., T. Ninomiya, and J. Tsujii, "Corpus-Oriented Grammar Development for Acquiring a Head-Driven Phrase Structure Grammar from the Penn Treebank," *Natural Language Processing*, K.-Y. Su, et al., (eds.), Lecture Notes in Computer Science, Vol. 3248, Berlin: Springer-Verlag, 2005, pp. 684–693.

[28] McCray, A. T., and O. Bodenreider, "A Conceptual Framework for the Biomedical Domain," *The Semantics of Relationships: An Interdisciplinary Perspective*, R. Green, C. A. Bean, and S. H. Myaeng, (eds.), Dordrecht: Kluwer, 2002, pp. 181–198.

[29] Gene Ontology Consortium, "Creating the Gene Ontology Resource: Design and Implementation," *Genome Research*, Vol. 11, No. 8, 2001, pp. 1425–1433.

[30] Carreras, X., and L. Márquez, "Introduction to the CoNLL-2004 Shared Task: Semantic Role Labeling," *Proc. 8th Conf. on Computational Natural Language Learning (CoNLL-2004) at HLT-NAACL 2004*, Boston, MA, May 6–7, 2004, pp. 89–97.

[31] Gomez, F., R. Hull, and C. Segami, "Acquiring Knowledge from Encyclopedic Texts," *Proc. 4th Conf. on Applied Natural Language Processing*, Stuttgart, Germany, October 13–15, 1994, pp. 84–90.

[32] Ono, T., et al., "Automated Extraction of Information on Protein-Protein Interactions from the Biological Literature," *Bioinformatics*, Vol. 17, No. 2, 2001, pp. 155–161.

[33] Ding, J., et al., "Extracting Biochemical Interactions from MEDLINE Using a Link Grammar Parser, " *Proc. 15th IEEE Int. Conf. on Tools with Artificial Intelligence (ICTAI 2003)*, Sacramento, CA, November 3–5, 2003, pp. 467–473.

[34] Palmer, M., D. Gildea, and P. Kingsbury, "The Proposition Bank: An Annotated Corpus of Semantic Roles," *Computational Linguistics*, Vol. 31, No. 1, 2005, pp. 71–105.

[35] Srinivasan, P., "Text Mining: Generating Hypotheses from MEDLINE," *Journal of the American Society for Information Science*, Vol. 55, No. 5, 2004, pp. 396–413.

[36] Schuemie, M. J., et al., "Distribution of Information in Biomedical Abstracts and Full-Text Publications," *Bioinformatics*, Vol. 20, No. 16, 2004, pp. 2597–2604.

[37] Castaño, J., J. Zhang, and J. Pustejovsky, "Anaphora Resolution in Biomedical Literature," *Proc. Int. Symp. on Reference Resolution*, Alicante, Spain, June 3–4, 2002.

[38] Hahn, U., M. Romacker, and S. Schulz, "Discourse Structures in Medical Reports—Watch Out! The Generation of Referentially Coherent and Valid Text Knowledge Bases in the MEDSYNDIKATE System," *International Journal of Medical Informatics*, Vol. 53, No. 1, 1999, pp. 1–28.

[39] Mizuta, Y., and N. Collier, "Zone Identification in Biology Articles as a Basis for Information Extraction," *Proc. Int. Workshop on Natural Language Processing in Biomedicine and its Applications (JNLPBA 2004)*, Geneva, Switzerland, August 28–29, 2004, pp. 29–35.

[40] Friedman, C., et al., "GENIES: A Natural-Language Processing System for the Extraction of Molecular Pathways from Journal Articles," *Bioinformatics*, Vol. 17, Suppl. 1, 2001, pp. S74–S82.

[41] Rzhetsky, A., et al., "GeneWays: A System for Extracting, Analyzing, Visualizing, and Integrating Molecular Pathway Data," *Journal of Biomedical Informatics*, Vol. 37, No. 1, 2004, pp. 43–53.

[42] Gaizauskas, R., et al., "Protein Structures and Information Extraction from Biological Texts: The PASTA System," *Bioinformatics*, Vol. 19, No. 1, 2003, pp. 135–143.

[43] Craven, M., and J. Kumlien, "Constructing Biological Knowledge Bases by Extracting Information from Text Sources," *Proc. 7th Int. Conf. on Intelligent Systems for Molecular Biology (ISMB'99)*, Heidelberg, Germany, August 6–10, 1999, pp. 77–86.

[44] Phuong, T. M., D. Lee, and H. Lee-Kwang, "Learning Rules to Extract Protein Interactions from Biomedical Text," *Advances in Knowledge Discovery and Data Mining*, K.-Y. Whang, et al., (eds.), Heidelberg, Germany: Springer, 2003, pp. 148–158.

[45] Kulick, S., et al., "Integrated Annotation for Biomedical Information Extraction," *Proc. NAACL/HLT Workshop on Linking Biological Literature, Ontologies and Databases: Tools for Users (BioLink 2004)*, Boston, MA, May 2–7, 2004, pp. 61–68.

3

Lexical, Terminological, and Ontological Resources for Biological Text Mining

Olivier Bodenreider

3.1 Introduction

Biomedical terminologies and ontologies are frequently described as enabling resources in text mining systems [1–3]. These resources are used to support tasks such as entity recognition (i.e., the identification of biomedical entities in text), and relation extraction (i.e., the identification of relationships among biomedical entities). Although a significant part of current text mining efforts focuses on the analysis of documents related to molecular biology, the use of lexical, terminological, and ontological resources is mentioned in research systems developed for the analysis of clinical narratives (e.g., MedSyndikate [4]), or the biological literature (e.g., BioRAT [5], GeneScene [6], EMPathIE [7], and PASTA [7]). Of note, some systems initially developed for extracting clinical information later have been adapted to extract relations among biological entities (e.g., MedLEE [8], GENIES [9], and SemRep/SemGen [10]). Commercial systems such as TeSSI, from Language & Computing (http://www.landcglobal.com/) also make use of such resources.

Entity recognition often draws on lists of entity names collected in lexicons, gazetteers, and, more generally, terminology resources. For example, lists of disease names can be easily extracted from disease resources, such as the International Classification of Diseases (ICD); from the disease component of general resources, such as the Medical Subject Headings (MeSH); and from specialized resources, such as the Online Multiple Congenital Anomaly/Mental

Retardation (MCA/MR) Syndromes. In contrast, relation extraction may benefit from the relationships represented among terms in terminologies (e.g., 'Parkinson's disease' *child of* 'Neurodegenerative diseases' in MeSH), and in ontologies [e.g., 'Basal ganglia' *finding site of* 'Parkinson's disease' in the Systematized Nomenclature of Medicine Clinical Terms (SNOMED CT)].

Biomedical lexicons, such as the UMLS Specialist lexicon, collect lexical items (i.e., words and multiword expressions) frequently observed in biomedical text corpora, and record information about them, including parts of speech (e.g., noun or adjective), inflectional variants (e.g., singular or plural), and spelling variants (e.g., American versus British English). This information is useful not only to NLP tools, such as part-of-speech taggers and parsers, but also to entity recognition systems, since it can help identify variants of entity names in text [11].

The purpose of biomedical terminology is to collect the names of entities employed in the biomedical domain. Most biomedical terminologies record synonymous terms (e.g., *Parkinson's disease* and *Paralysis agitans*), and have some kind of hierarchical organization, often treelike or graphlike [12]. Terminology-driven approaches to text mining have been explored in [13].

In contrast, biomedical ontology aims to study entities (i.e., substances, qualities, and processes) of biomedical significance, and the relations among them. Examples of such entities include substances such as the mitral valve and glucose, qualities such as the diameter of the left ventricle and the catalytic function of enzymes, and processes such as blood circulation and secreting hormones. Fundamental relations in biomedical ontologies include not only *is a* and *part of*, but also *instance of*, *adjacent to*, *derives from*, and so forth [14].

In practice, the distinction between lexicons, terminologies, and ontologies is not always sharp. On the one hand, although ontologies mostly focus on relations among entities, some of them also record the names by which entities are referred. On the other hand, although terminologies essentially collect the names of entities, their hierarchical organization also reflects relations among such entities. Finally, the very names of these resources can be misleading. For example, despite its name, the GO defines itself as a controlled vocabulary (i.e., a terminological resource), but like ontologies, its terms are linked by relationships such as *is a* and *part of*. However, the definition and use of such relations is not consistent throughout GO [15], as would be expected from ontologies.

The objective of this chapter is to present some of the resources (lexicons, terminologies, and ontologies) of interest for entity recognition and relation extraction tasks. Providing an exhaustive list of these resources is beyond the scope of this chapter. Many of these resources are highly specialized, and therefore would be of little interest to most readers. Instead, we have selected general,

publicly available resources that have been shown to be useful for biomedical text mining. This review is purposely limited to resources in English.

We start by presenting an extended example, illustrating biomedical terms in two pieces of text. We then give a brief description of the major resources available, with a particular emphasis on the UMLS [16]. Finally, we discuss some issues related to biomedical terms and biomedical relations. The reader is referred to Chapters 6 and 7 for a detailed presentation of the tasks of entity recognition and relation extraction.

3.2 Extended Example

In this example, we consider two short pieces of text related to the genetic disease neurofibromatosis 2. This is an autosomal dominant disease, characterized by tumors called schwannomas that involve the acoustic nerve, as well as other features [17]. The disorder is caused by mutations of the *NF2* gene, resulting in absence or inactivation of the protein product. The protein product of *NF2* is commonly called merlin (but also neurofibromin 2 and schwannomin), and functions as a tumor suppressor. The first fragment of text (3.1) is extracted from the abstract of an article [18]. The second is the definition of neurofibromatosis 2 in the MeSH vocabulary (http://www.nlm.nih.gov/mesh/).

- (3.1) Neurofibromatosis type 2 (NF2) is often not recognized as a distinct entity from peripheral neurofibromatosis. NF2 is a predominantly [intracranial condition] whose hallmark is [bilateral vestibular schwannomas]. NF2 results from a mutation in the gene named merlin, located on chromosome 22.
- (3.2) Neurofibromatosis type 2: An [autosomal dominant disorder] characterized by a high incidence of [bilateral acoustic neuromas] as well as schwannomas of other [cranial and peripheral nerves], and other [benign intracranial tumors] including meningiomas, ependymomas, spinal neurofibromas, and gliomas. The disease has been linked to mutations of the NF2 gene on chromosome 22 (22q12) and usually presents clinically in the first or second decade of life.

3.2.1 Entity Recognition

Many biomedical entities can be identified in these two fragments. Underlined expressions correspond to terms present in the UMLS Metathesaurus. This is the case, for example, of the disease *neurofibromatosis 2* and the protein *merlin*. Interestingly, *vestibular schwannomas* in (3.1) and *acoustic neuromas* in (3.2), although lexically distinct, name the same tumor. While a lexicon is useful to

identify these disease names, a terminology (or ontology) is required to identify them as synonymous. These two terms are names for the same disease concept in the UMLS Metathesaurus (C0027859). The list of UMLS concepts that can be identified in the two text fragments is given in Table 3.1.

Many expressions extracted from the two text fragments can be mapped to the UMLS Metathesaurus through a simple match (i.e., exact match or after normalization). Except for *merlin*, which maps to both a protein and a bird, the mapping is unambiguous. In contrast, expressions of (3.1) and (3.2) enclosed in brackets also correspond to biomedical entities, but the name found in the text cannot be mapped directly to a UMLS concept. Expressions, such as *intracranial condition* in (3.1), are vague, compared to the corresponding concept names in the UMLS (e.g., 'central nervous system diseases'). Complex phrases, such as *cranial and peripheral nerves* in (3.2), refer to two concepts (i.e., 'cranial nerves' and 'peripheral nerves') present in the Metathesaurus. Conversely, some expressions in the text convey more precision than the corresponding concepts found in biomedical terminologies [e.g., *bilateral vestibular schwannomas* in (3.1) versus 'vestibular schwannomas' and *benign intracranial tumors* in (3.2) versus 'intracranial tumors']. In these cases, while terminological resources are useful for identifying entities in text, they may not be sufficient for capturing all nuances present in the text. Term variation and management issues are discussed extensively in Chapter 4.

3.2.2 Relation Extraction

Once entities have been identified in text fragments, the next step consists of identifying the relationships among them, such as *vestibular schwannomas* ***manifestation of*** *neurofibromatosis 2* and *NF2 gene* ***located on*** *chromosome 22*. Such relations may be explicitly represented in biomedical ontologies. For example, the relation 'schwannomas' *associated morphology of* 'neurofibromatosis 2' is asserted in SNOMED CT. However, ontologies do not necessarily contain such fine-grained assertions, but may rather represent higher-level facts such as 'gene' *located on* 'chromosome'. A relation extraction system would first identify *NF2 gene* as a kind of gene and *chromosome 22* as a kind of chromosome, before inferring that a particular gene (*NF2 gene*) is located on a particular chromosome (*chromosome 22*).

The use of ontologies to support relation extraction often requires the system to identify in the text not only entities, but also potential relationships. Clues for identifying relationships include lexical items (e.g., the preposition 'on' for the relationship *located on*), and syntactic structures (e.g., *intracranial tumors including meningiomas* for *meningiomas* ***is a*** *intracranial tumors*), as well as statistical and pattern-based clues (not presented here). Relations may span several sentences, and their identification often requires advanced linguistic

Table 3.1
UMLS Concepts: Identifiers, Preferred Names, and Semantic Types, Identifiable in Text Fragments (3.1) and (3.2)

Source	String in Text	M	CUI	Preferred Name	S. Types
(1) (2)	Neurofibromatosis type 2	s	C0027832	Neurofibromatosis 2	neop
(1)	NF2	s	C0085114	Neurofibromatosis 2 genes	gngm
(1)	peripheral neurofibromatosis	s	C0027831	Neurofibromatosis 1	neop
(1)	{intracranial condition}	—	C0007682	Central Nervous System Diseases	dsyn
(1)	[bilateral] vestibular schwannomas	a	C0027859	Neuroma, Acoustic	neop
(1) (2)	mutation/mutations	s	C0026882	Mutation	genf
(1)	gene	s	C0017337	Genes	gngm
(1)	merlin	m	C0254123	Neurofibromin 2	aapp, bacs
(1) (2)	chromosome 22	s	C0008665	Chromosomes, Human, Pair 22	celc
(2)	autosomal dominant disorder	a	C0265385	Autosomal dominant hereditary disorder	dsyn
(2)	bilateral acoustic neuromas	s	C1136042	Neuroma, Acoustic, Bilateral	neop
(2)	schwannomas	s	C0027809	Neurilemmoma	neop
(2)	cranial and peripheral nerves	—	C0010268	Cranial Nerves	bpoc
			C0031119	Peripheral Nerves	bdsy
(2)	[benign] intracranial tumors	a	C0750978	Neoplasms, Intracranial	neop
(2)	meningiomas	s	C0025286	Meningioma	neop
(2)	ependymomas	s	C0014474	Ependymoma	neop
(2)	neurofibromas	s	C0027830	Neurofibroma	neop
(2)	gliomas	s	C0017638	Glioma	neop
(2)	disease	s	C0012634	Disease	dsyn
(2)	NF2 gene	s	C0085114	Neurofibromatosis 2 genes	gngm

CUI: identifier (see Table 3.4 for the full names).

Column 'M' indicates the type of match (s: single simple match, m: multiple simple matches, a: approximate match, -: no direct match).

techniques, such as anaphora and coreference resolution. For example, from the last sentence of (3.2), the relation *disease* ***associated with*** *mutation* can be extracted. While accurate, this relation is incomplete in this context, because *disease* actually refers not to any disease, but to *neurofibromatosis 2* (anaphoric relation). Similarly, *mutations of the NF2 gene* (not mutations in general) is the entity associated with the disease. Therefore, the complete relation to be extracted is *neurofibromatosis 2 associated with mutations of the NF2 gene*. The potential relations extracted from the text then can be validated against the relations explicitly represented in the ontology, or inferred from it.

3.3 Lexical Resources

The resources presented under this category provide the lexical and lexico-syntactic information needed for parsing text. The major resource for biomedical text is the Specialist lexicon. Additionally, specialized resources can be useful for analyzing subdomains of biomedicine (e.g., lists of gene names for molecular biology corpora). Conversely, general resources such as WordNet also can help analyze the literature written for less-specialized audiences (e.g., patients).

3.3.1 WordNet

WordNet is an electronic lexical database developed at Princeton University, which serves as a resource for applications in natural language processing and information retrieval [19]. The core structure in WordNet is a set of synonyms (synset) that represents one underlying concept. For example, the synset representing 'hemoglobin' also contains the lexical entries 'haemoglobin' (British-English spelling) and 'Hb' (abbreviation). A definition is provided for the synset: "a hemoprotein composed of globin and heme that gives red blood cells their characteristic color; function primarily to transport oxygen from the lungs to the body tissues." There are separate structures for each linguistic category covered: nouns, verbs, adjectives, and adverbs. For example, the adjective 'renal' and the noun 'kidney', although similar in meaning, belong to two distinct structures, and a specific relationship (pertainymy) relates the two forms. The current version of WordNet (2.0) contains over 114,000 noun synsets. In addition to being a lexical resource, WordNet has some of the features of an ontology. For example, each synset in the noun hierarchy belongs to at least one *is a* tree (e.g., 'hemoglobin' *is a* 'protein'), and may additionally belong to several *part of*-like trees ('hemoglobin' *substance of* 'red blood cell'). Because of its modest coverage of the biomedical domain [20, 21], WordNet has been used only in a limited number of projects in biomedicine [22], where resources such as the UMLS usually play a more prominent role. WordNet is available free of charge

from http://wordnet.princeton.edu/. Application programming interfaces (API) have been developed for the major programming languages, making it relatively easy for developers to integrate WordNet into their applications.

3.3.2 UMLS Specialist Lexicon

The Specialist lexicon is one of three knowledge sources developed by the NLM, as part of the UMLS project. It provides the lexical information needed for processing natural language in the biomedical domain [23]. The lexicon entry for each word or multiword term records syntactic (part-of-speech, allowable complementation patterns), morphological (base form, inflectional variants), and orthographic (spelling variants) information. It is, in fact, a general English lexicon that includes many biomedical terms. Lexical items are selected from a variety of sources, including lexical items from MEDLINE/PubMed citation records, the UMLS Metathesaurus, and a large set of lexical items from medical and general English dictionaries. Contrary to WordNet, the Specialist lexicon does not include any information about synonymy or semantic relations among its entries. However, this information is present in the Metathesaurus, another component of the UMLS (see Section 3.4.3). The record for 'hemoglobin' in the Specialist lexicon, shown in Figure 3.1, indicates the base form, one spelling variant, and two inflectional classes, since hemoglobin is used as both a mass noun (e.g., in *Hemoglobin concentration is reported as grams of hemoglobin per deciliter of blood*), and as a countable (e.g., in *the study of hemoglobins, both normal and mutant*). Additionally, the abbreviation 'Hb' and the acronym 'Hgb' are cross-referenced to 'hemoglobin'. The Specialist lexicon is distributed as part of the UMLS, and can be queried through application programming interfaces for Java and XML. It is also available as an open source resource, as part of the Specialist NLP tools (http://SPECIALIST.nlm.nih.gov).

3.3.3 Other Specialized Resources

While general resources such as WordNet and the Specialist lexicon provide a good coverage of the general biomedical language, they (purposely) fail to cover

```
{
  base=hemoglobin            (base form)
  spelling_variant=haemoglobin
  entry=E0031208             (identifier)
  cat=noun                   (part of speech)
  variants=uncount           (no plural)
  variants=reg               (plural: hemoglobins, haemoglobins )
}
```

Figure 3.1 Representation of 'hemoglobin' in the Specialist lexicon.

in detail specialized subdomains, such as gene and protein names, or chemical and drug names. Therefore, the syntactic analyzers and parsers relying on these resources may give suboptimal results when analyzing specialized corpora (e.g., molecular biology abstracts). One approach to solving this problem is to use machine learning techniques to identify the names of specialized entities. Alternatively, or in conjunction with these techniques, resources, such as lists of gene, protein, chemical, and drug names, can be exploited [24]. In molecular biology, for example, the Human Genome Organization (HUGO) has established, through its Gene Nomenclature Committee (HGNC), a list of over 20,000 approved gene names and symbols, called Genew [25]. Recorded in this database are the symbol 'NF2' and the name 'neurofibromin 2 (bilateral acoustic neuroma)' for the gene merlin, whose mutation causes the disease neurofibromatosis 2. More generally, lists of names for specialized entities can be extracted from specialized resources. Examples of publicly available specialized resources for genes, proteins, chemical entities, and drugs are given in Table 3.2. Finally, acronyms and abbreviations harvested from the biomedical literature [26, 27] and collected in databases [28] also can benefit entity recognition applications. This issue is discussed extensively in Chapter 6.

3.4 Terminological Resources

The purpose of terminology is to collect the names of entities employed in the biomedical domain [29]. Terminologies typically provide lists of synonyms for the entities in a given subdomain and for a given purpose. As such, they play an important role in entity recognition. Additionally, most terminologies have some kind of hierarchical organization that can be exploited for relation extraction purposes. Many terminologies consist of a tree, where nodes are terms, and links represent parent-to-child or more-general-to-more-specific relationships. Some terminologies allow multiple inheritance, and have the structure of a directed acyclic graph. The Gene Ontology and MeSH provide examples of terminological systems created to support different tasks. Because it integrates a large number of terminologies, the UMLS Metathesaurus is the terminological system most frequently used in the analysis of biomedical texts.

3.4.1 Gene Ontology

The Gene Ontology (GO) is a controlled vocabulary, developed by the Gene Ontology Consortium, for the annotation of gene products in model organisms. GO is organized in three separate hierarchies—for molecular functions (6,933 terms), biological processes (9,053 terms), and cellular components (1,414 terms), as of February 1, 2005 [30]. For example, annotations for the

Table 3.2
Examples of Publicly Available Specialized Resources for Genes, Proteins, Chemical Entities, and Drugs

Domain	Resources	URL
Genes and proteins	Genew	http://www.gene.ucl.ac.uk/nomenclature/
	Entrez Gene (formerly LocusLink)	http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=gene
	UniProt	http://www.ebi.uniprot.org/index.shtml
Chemical entities	PubChem	http://pubchem.ncbi.nlm.nih.gov/
	ChemIDplus	http://chem.sis.nlm.nih.gov/chemidplus/chemidlite.jsp
	ChEBI	http://www.ebi.ac.uk/chebi/
Drugs	RxNorm	http://www.nlm.nih.gov/research/umls/rxnorm_main.html
	National Drug Code	http://www.fda.gov/cder/ndc/

gene *NF2* in the Gene Ontology Annotation (GOA) database (http://www.ebi.ac.uk/GOA/) include the molecular function term 'cytoskeletal protein binding', the biological process term 'negative regulation of cell proliferation', and the cellular component terms 'plasma membrane' and 'cytoskeleton'. Each of the three hierarchies is organized in a directed acyclic graph, in which the nodes are GO terms, and the edges represent the GO relationships *is a* and *part of*. For example, as illustrated in Figure 3.2, the relations of the cellular component 'cytoskeleton' to its parent terms include 'cytoskeleton' *is a* 'intracellular nonmembrane-bound organelle' and 'cytoskeleton' *part of* 'intracellular'. GO terms may have synonyms (e.g., synonyms for 'plasma membrane' include 'cytoplasmic membrane' and 'plasmalemma'). Most terms have a textual definition (e.g., for 'plasma membrane': "The membrane surrounding a cell that separates the cell from its external environment. It consists of a phospholipid bilayer and associated proteins.").

Both the names and the relations comprised in the Gene Ontology can benefit text mining applications. The names of molecular functions, biological processes, and cellular components are frequently used in the biomedical literature [31]. For example, the biological process 'activation of MAPK' and the cellular component 'adherens junction' can be identified in the title *Erbin regulates MAP kinase activation and MAP kinase-dependent interactions between merlin and adherens junction protein complexes in Schwann cells*. As illustrated in the

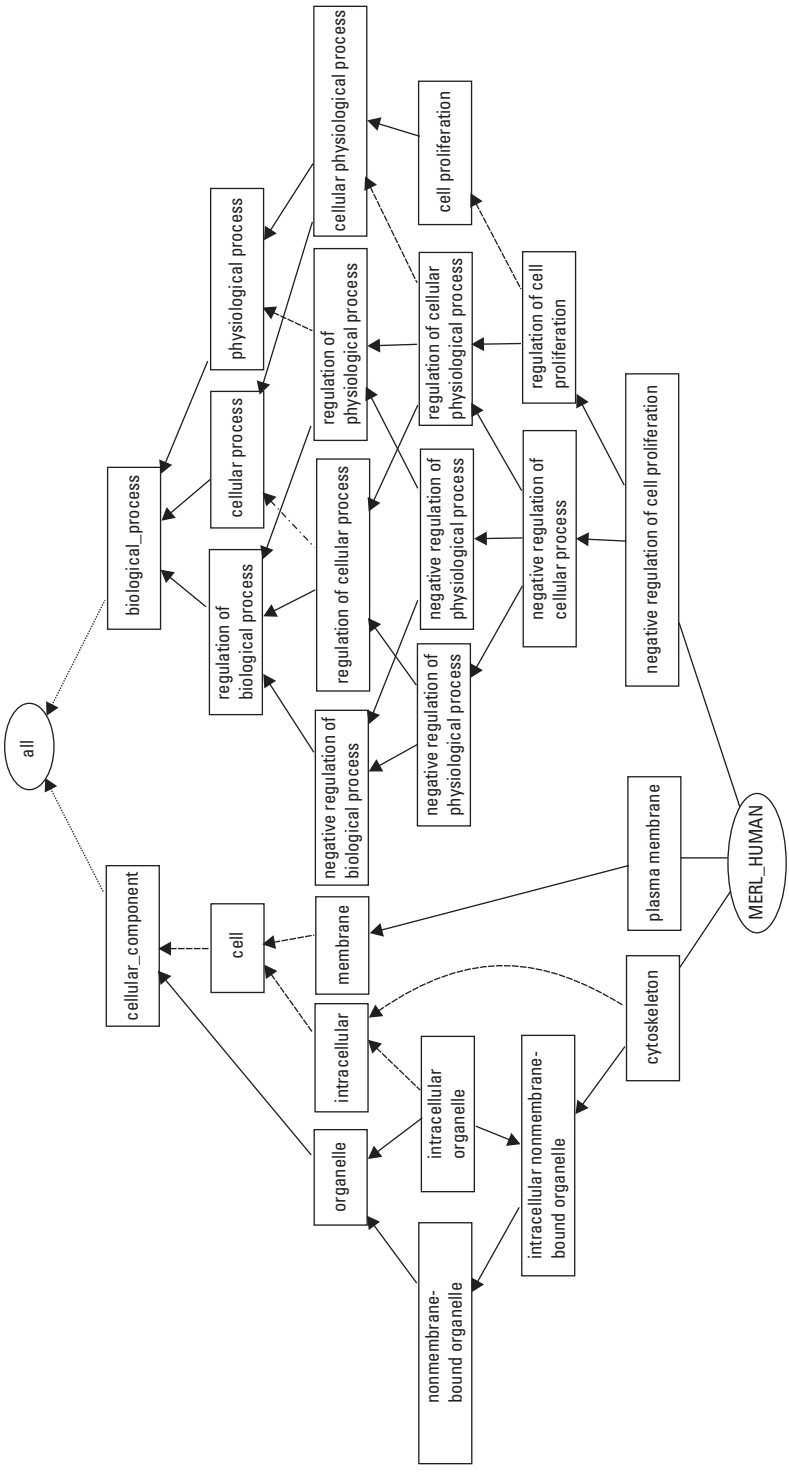


Figure 3.2 Representation of the gene product *merlin* ('MERL_HUMAN'). Solid lines represent some of its annotations in the Gene Ontology. Solid and dashed arrows represent the Gene Ontology relationships *is a* and *part of*, respectively.

following text fragment (3.3), hierarchical relations can help resolve anaphora and interpret associative relations.

- (3.3) The organization of the actin cytoskeleton in prefusion aligning myoblasts is likely to be important for their shape and interaction. We investigated actin filament organization and polarity by transmission electron microscopy (TEM) in these cells.

The terms *actin cytoskeleton* and *actin filament* identified in the first two sentences of (3.3) are present in GO. Moreover, a relation between them is explicitly recorded in GO ('actin filament' *part of* 'actin cytoskeleton'), which helps link together the two sentences. However, many concepts and relations are not represented in GO, or other biomedical terminologies. For example, a relation between *myoblasts* and *these cells*—namely, 'myoblast' *is a* 'cell'—is needed to resolve the anaphoric relation between the two terms in (3.3). Such a relation cannot be found in GO, where the term *myoblast* is not even represented.

Finally, GO terms constitute an entry point to annotation databases, providing a wealth of relations between gene products and the molecular functions, biological processes, and cellular components with which they are associated (e.g., 'NF2' *has biological process* 'negative regulation of cell proliferation'). GO is available from http://geneontology.org/, and is distributed in various formats, including XML and database formats. Perl and Java application programming interfaces are also available. GO is one of the source vocabularies included in the UMLS Metathesaurus. GO is a member of a family of controlled vocabularies, called Open Biomedical Ontologies (OBO). These resources can be useful in text mining applications as a source of specialized vocabulary (e.g., for chemicals or experimental conditions). OBO resources are available at http://obo.sourceforge.net.

3.4.2 Medical Subject Headings

The Medical Subject Headings (MeSH) thesaurus is a controlled vocabulary, produced by the National Library of Medicine, and used for indexing, cataloging, and searching for biomedical and health-related information and documents [32]. It consists of 22,995 descriptors (main headings), organized in 15 hierarchies. Additionally, a set of approximately 150,000 "supplementary concept records" provides a finer-grained representation of biomedical entities, including chemicals and proteins. A list of entry terms (synonyms or closely related terms) is given for each descriptor. Entry terms for the disease 'Neurofibromatosis 2' include *Neurofibromatosis Type II*, *Bilateral Acoustic Neurofibromatosis*, *Bilateral Acoustic Schwannoma*, and *Familial Acoustic Neuromas*. A scope note often provides a definition of the descriptor. In the

MeSH thesaurus, descriptors are related by parent/child relations; each descriptor has at least one parent, and may have several. For example, 'Neurofibromatoses' and 'Neuroma, Acoustic' are the two parents of the descriptor 'Neurofibromatosis 2'. The arrangement of MeSH descriptors in hierarchies is intended to serve the purpose of indexing and information retrieval, and does not always follow strict classifications. In addition to hierarchical relations, cross references may link a descriptor to descriptors from other hierarchies. For example, the disease 'Neurofibromatosis 2' is linked to the protein 'Neurofibromin 2' and to the gene 'Genes, Neurofibromatosis 2'. The MeSH thesaurus is used by the NLM for indexing articles from 4,600 biomedical journals for the MEDLINE/PubMed database. Like GO, MeSH can be used in text mining applications, due to the many names and relations it provides. Its scope is broader than that of GO, but its granularity is coarser. MeSH is available from http://www.nlm.nih.gov/mesh/ in various formats, including XML. MeSH is one of the source vocabularies included in the UMLS Metathesaurus.

3.4.3 UMLS Metathesaurus

The UMLS Metathesaurus is one of three knowledge sources developed and distributed by the NLM, as part of the UMLS project [16]. Version 2005AA of the Metathesaurus contains over 1 million biomedical concepts and 5 million concept names, from more than 100 controlled vocabularies and classifications (some in multiple languages) used in patient records, administrative health data, bibliographic and full-text databases, and expert systems. The Metathesaurus also records over 16 million relations among these concepts, either inherited from the source vocabularies or specifically generated. While the Metathesaurus preserves the names, meanings, hierarchical contexts, attributes, and interterm relationships present in its source vocabularies, it also integrates existing terminologies into a common semantic space. As in WordNet, synonymous names are clustered together to form a concept. The Metathesaurus also assigns a unique identifier to each concept, and establishes new relations between terms from different source vocabularies, as appropriate. Each concept is also categorized with at least one semantic type from the UMLS Semantic Network (see Section 3.5.2), independent of its hierarchical position in the source vocabularies. The scope of the Metathesaurus is determined by the combined scope of its source vocabularies, including Gene Ontology and MeSH, disease vocabularies (e.g., International Classification of Diseases), clinical vocabularies (e.g., SNOMED CT), nomenclatures of drugs and medical devices, as well as the vocabularies of many subdomains of biomedicine (e.g., nursing, psychiatry, and gastrointestinal endoscopy).

Examples of Metathesaurus concepts are given in Table 3.1. C0254123 identifies the protein 'neurofibromin 2', whose synonyms include *merlin*, *NF2*

protein, and *schwannomin*. Its semantic types are 'Amino Acid, Peptide, or Protein' and 'Biologically Active Substance'. The following source vocabularies contributed names to this concept: MeSH, SNOMED CT, and the NCI Thesaurus. Once integrated in the Metathesaurus, 'neurofibromin 2' has multiple parents including 'membrane proteins' (from MeSH), 'tumor suppressor proteins' (from both MeSH and SNOMED CT), and 'signaling protein' (from the NCI Thesaurus). Its only descendant is 'merlin, Drosophila' (from MeSH). Beside hierarchical relations, associative relations link the protein 'neurofibromin 2' to the gene 'neurofibromatosis 2 genes' and to the disease 'neurofibromatosis 2'. The frequencies of co-occurrence of MeSH descriptors in MEDLINE/PubMed citations are also recorded in the Metathesaurus. For example, during the last 10 years, the descriptors 'Neurofibromin 2' and 'Neurofibromatosis 2' occurred together 13 times as major descriptors. The descriptors 'Membrane Proteins' (8 times), 'Phosphoproteins' and 'NF2 gene' (7 times), and 'Cell Transformation, Neoplastic' (5 times) frequently co-occur with 'Neurofibromin 2'.

Section 3.2.1 illustrated how the Metathesaurus can be used in entity recognition and relation extraction tasks. Used in many biomedical entity recognition studies, the MetaMap (MMTx) program has been specially designed to take advantage of the features of the UMLS Metathesaurus and Specialist lexicon [33]. MMTx is available from http://mmtx.nlm.nih.gov/. Besides text mining, the Metathesaurus is used in a wide range of applications, including linking between different clinical or biomedical vocabularies, information retrieval and indexing, and biomedical language processing. The Metathesaurus is available from http://umlsks.nlm.nih.gov/ (or on DVD) in relational database format. Users must complete the License Agreement for the Use of UMLS Metathesaurus. Java and XML application programming interfaces are available for the Metathesaurus.

3.5 Ontological Resources

Biomedical ontology aims to study the kinds of entities (i.e., substances, qualities, and processes) of biomedical significance. Unlike biomedical terminology, biomedical ontology is not primarily concerned with names, but with the principled definition of biological classes and their interrelations. In practice, since most terminologies have some degree of organization, and many ontologies also collect names for their entities, the distinction between ontological and terminological resources is somewhat arbitrary. See Chapter 4 for further discussion of this issue. Because they share many characteristics with ontologies, we will list under this rubric two broad resources—SNOMED CT and the UMLS Semantic Network. Other ontologies will be briefly discussed.

3.5.1 SNOMED CT

The Systematized Nomenclature of Medicine (SNOMED) Clinical Terms (SNOMED CT), developed by the College of American Pathologists, was formed by the convergence of SNOMED RT and Clinical Terms Version 3 (formerly known as the Read Codes). SNOMED CT is the most comprehensive biomedical terminology recently developed in native description logic formalism.[1] The version described here is dated January 31, 2004, and contains some 270,000 concepts, named by over 400,000 names. SNOMED CT consists of 18 independent hierarchies reflecting, in part, the organization of previous versions of SNOMED into "axes," such as 'Diseases', 'Drugs', 'Living organisms', 'Procedures', and 'Topography'. Each SNOMED CT concept is described by a variable number of elements. For example, the concept 'Neurofibromatosis, type 2' has a unique identifier (92503002), several names (*Bilateral acoustic neurofibromatosis*, *BANF - Bilateral acoustic neurofibromatosis*, *Neurofibromatosis, type 2*, and *Neurofibromatosis type 2*), and has multiple *is a* parents, including 'Congenital anomaly of inner ear', 'Neoplasm of uncertain behavior of cranial nerve', and 'Acoustic neuroma'. 'Neurofibromatosis, type 2' also participates in a complex network of associative relations to other concepts. The relations (called roles), shown in Table 3.3, indicate that the lesions encountered in 'Neurofibromatosis, type 2' include neurofibromatosis of the vestibulocochlear nerve (Group 1) and neurilemoma of the vestibular nerve (Group 3). SNOMED CT is available as part of the UMLS (from http://umlsks.nlm.nih.gov/), at no charge for UMLS licensees in the United States. The structure of the UMLS Metathesaurus has been modified to accommodate the level of detail provided by ontological resources like SNOMED CT. Because SNOMED CT has become available through the UMLS only since 2004, the number of studies reporting its uses is still limited.

3.5.2 UMLS Semantic Network

The UMLS Semantic Network is one of three knowledge sources developed and distributed by the NLM, as part of the UMLS project. It was created in an effort to provide a semantic framework for the UMLS and its constituent vocabularies [34]. Unlike the Metathesaurus, the Semantic Network is a small structure, composed of 135 high-level categories called semantic types. It is organized in two single-inheritance hierarchies: one for 'Entity' and one for 'Event'. In addition to *is a*, 53 kinds of relationships are defined in the Semantic Network, which are used to represent over 6,700 relations—hierarchical and associative—among semantic types. Semantic types from the Semantic Network are

1. Although not distributed in any description logic (DL) language, SNOMED CT has been developed in native DL formalism, thus ensuring the consistency of its hierarchical relations.

Table 3.3
Some of the Roles Present in the Definition of 'Neurofibromatosis, type 2'

Group	Role	Value
1	Associated morphology	'Neurofibromatosis'
	Finding site	'Skin structure'
	Finding site	'Vestibulocochlear nerve structure'
3	Associated morphology	'Neurilemoma'
	Finding site	'Vestibular nerve structure'

linked to Metathesaurus concepts by the categorization link established by the Metathesaurus editors. Each concept is categorized with at least one semantic type from the Semantic Network, independently of its hierarchical position in the source vocabularies. Fifteen collections of semantic types, called semantic groups, have been defined in order to partition Metathesaurus concepts into a smaller number of semantically consistent groups [35].

Semantic types for the Metathesaurus concepts listed in Table 3.1 are presented in Table 3.4, along with the corresponding semantic groups. For example, the concept 'Neurofibromatosis 2' is categorized as 'Neoplastic Process', a semantic type from the semantic group 'Disorders'. In addition to 'mutation,' Metathesaurus concepts categorized with 'Genetic Function' include 'alternative splicing', 'loss of heterozygosity', and 'ribonuclease activity'. Examples of

Table 3.4
Semantic Types and Semantic Groups for the Metathesaurus Concepts Listed in Table 3.1

ST Abbreviation	ST Name	Semantic Group
aapp	Amino Acid, Peptide, or Protein	Chemicals & Drugs
bacs	Biologically Active Substance	Chemicals & Drugs
bdsy	Body System	Anatomy
bpoc	Body Part, Organ, or Organ Component	Anatomy
celc	Cell Component	Anatomy
dsyn	Disease or Syndrome	Disorders
genf	Genetic Function	Physiology
gngm	Gene or Genome	Genes & Molecular Sequences
neop	Neoplastic Process	Disorders

relations among semantic types include 'Body Part, Organ, or Organ Component' *location of* 'Neoplastic Process;' 'Pharmacologic Substance' *treats* 'Neoplastic Process;' and 'Neoplastic Process' *manifestation of* 'Genetic Function.' A relationship between two semantic types indicates a *possible* link between the concepts categorized with these semantic types. In natural language processing and text mining applications, Semantic Network relations are typically used as supporting evidence for the candidate predicates (i.e., <$concept_1$, relationship, $concept_2$> structures) extracted from the text [36]. For example, in *schwannomas of cranial nerves*, after identifying the concepts 'neurilemmoma' (from *schwannoma*) as a 'Neoplastic Process' and *cranial nerves* as a 'Body Part, Organ, or Organ Component,' the preposition *of* can be interpreted as indicating the location of the neoplastic process to the body part. This candidate predicate is supported by the Semantic Network relation 'Body Part, Organ, or Organ Component' location of 'Neoplastic Process.' Many relation extraction systems rely on correspondences established between semantic relations and linguistic phenomena [37]. Semantic Network relations also can be exploited in conjunction with relations among concepts in the Metathesaurus [38]. The Semantic Network is distributed as part of the UMLS, and is available from http://umlsks.nlm.nih.gov/. Like the other UMLS knowledge sources, it can be queried through application programming interfaces for Java and XML.

3.5.3 Other Ontological Resources

In addition to SNOMED CT and the UMLS Semantic Network, several ontological resources can be used to support text mining. The Foundational Model of Anatomy (FMA) (http://fma.biostr.washington.edu/) is a large reference ontology of anatomy, developed at the University of Washington [39]. In addition to NLP applications [40], the FMA has been used in entity recognition tasks [41], as well as relation extraction tasks [42]. Ontologies, such as OpenGALEN (http://www.opengalen.org/), have been developed to support terminological services [43], and may be less useful for text mining applications. For example, unlike terminologies, OpenGALEN does not record lists of synonyms for biomedical entities. For more information about biomedical ontologies, we refer the interested reader to [44].

3.6 Issues Related to Entity Recognition

The biomedical domain has a long tradition of collecting and organizing terms, as well as building classifications, dating back to the seventeenth century. The dozens of terminological resources resulting from this effort now benefit entity

recognition tasks. Moreover, UMLS has contributed to make existing terminologies both easier to use, by providing a common format and distribution mechanism, and more useful, by identifying synonymy and other semantic relations across them. As part of this effort, the NLM also developed the lexical resources (lexicon and lexical programs) used to detect lexical similarity among biomedical terms, and, more generally, to process biomedical text. This is the reason why the UMLS is used in a large number of text mining systems in biomedicine.

The properties of biomedical terms have been studied. For example, [45, 46] found matches for 10% to 34% of the UMLS strings in MEDLINE/PubMed (depending on the matching criteria used), and [45] developed a model for identifying the UMLS terms useful in NLP applications. In the domain of molecular biology, researchers have investigated the lexical properties of the GO: 35% of GO terms have been found in the biomedical literature [31], and 66% of GO terms are composed of other GO terms [47]. A model of compositionality in GO has even been proposed [48]. These studies have confirmed the interest of using existing terminological resources in entity recognition tasks.

However, there are some remaining challenges in biomedical entity recognition, including the limited coverage of terminological resources and ambiguity in biomedical names.

3.6.1 Limited Coverage

First, some subdomains remain only partially covered by existing resources. One example is given by genes and proteins, and, more generally, chemical entities. Names for such entities have proved difficult to exhaustively compile in terminologies. Vocabularies extracted from specialized databases may complement traditional terminologies. Moreover, while variant formation has been studied and effectively modeled for clinical terms [49], normalization techniques for the less regular names of entities employed in genomics have been only recently researched [50]. For these reasons, entity recognition techniques in this subdomain often include machine learning approaches, rather than the rule-based approach traditionally employed in biomedical NLP. Many gene name identification systems have been developed in the last 5 years [51–54]. Entity recognition systems in molecular biology texts may include algorithms rather than (or in addition to) static resources [24]. However, the product of some of these algorithms is made available to the research community by their authors. For example, [55] shares the lexicon of over 1 million gene and protein names extracted from the biomedical literature. Coverage issues have been explored in clinical terminologies as well [56], and techniques have been developed to extend the coverage of terminologies to specialized subdomains [57], or from specific corpora [58]. More generally, relation

extraction also may benefit from term extraction techniques resulting from research in terminology [59].

3.6.2 Ambiguity

The second issue is the ambiguity of many names in biology. This phenomenon is common in natural language, but poses specific challenges to biomedical entity recognition. Polysemy (several meanings for the same name) is illustrated by *NF2*, which simultaneously names the gene, the protein it produces, and the disease resulting from its mutation. While polysemy usually does not pose problems for domain experts, it makes it difficult for entity recognition systems to select the appropriate meaning. The ambiguity resulting from polysemous gene names has been quantified by Chen et al. [60]. These authors found modest ambiguities with general English words (0.57%) and medical terms (1.01%), but high ambiguity across species (14.20%). Ambiguity across species may be difficult to resolve; for example, when only capitalization conventions differentiate between gene names in various model organisms (e.g., *NF2* in Homo sapiens versus *Nf2* in Mus musculus). Various disambiguation strategies have been applied to biomedical language processing [61, 62]. However, further research is needed to develop strategies adapted to the specificity of molecular biology (e.g., ambiguity across species). Moreover, the limited availability of annotated resources, such as the GENIA corpus [63], hinders the development of unsupervised disambiguation techniques.

3.7 Issues Related to Relation Extraction

We round off our discussion of resources with a brief look at issues concerning relation extraction, to further emphasize the core enabling role of resources with respect to text mining.

3.7.1 Terminological Versus Ontological Relations

Not only do terminologies contain a large number of names for biomedical entities useful for entity recognition tasks, but they also represent a similarly considerable number of relations. For example, over 16 million relations are recorded in the UMLS Metathesaurus. While not all of them represent well-defined predicates or assertions, as would be expected from ontologies, these relations are essentially beneficial to applications such as relation extraction, especially when used in combination with lexico-syntactic clues and additional ontological relations.

The relations found in the most recent terminologies—often developed using knowledge representation techniques such as description logics—are generally better specified and principled, and therefore more directly useful for relation extraction. However, a careful inspection of these and other ontological resources through the prism of formal ontology reveals some limitations, especially in terms of consistency [15, 64, 65]. Applying formal ontological principles to biomedical ontologies results in a clarification of the relations [66], which, in turn, is expected to result in more consistent ontologies and more accurate inferences.

Recent experiments in reengineering terminologies have shown both the benefit and the cost (in terms of human resources) of such efforts [67, 68]. However, improving ontologies is likely to benefit relation extraction, as the candidate assertions extracted from text must be checked, not necessarily against relations explicitly represented in ontologies, but most often against inferred relations.

3.7.2 Interactions Between Text Mining and Terminological Resources

This chapter deliberately looks at ontologies and other resources as enabling resources for text mining, and relation extraction in particular. Conversely, it is worth mentioning that the relations extracted from text corpora and other knowledge sources (e.g., annotation databases) can help identify additional ontological relations. For example, lexico-syntactic patterns have been used to extract hypernymy relations from text corpora [69], and statistical methods have helped identify associative relations among GO terms [70]. In other words, the relations between text mining techniques and terminological resources are not unilateral. There is a virtual cycle, in which applications and resources benefit from one another. Studying this symbiotic relation is beyond the scope of this chapter. Various existing resources can be combined in order to create new resources. For example, semantic lexicons have been derived from lexicons, terminologies, and text corpora [71, 72].

3.8 Conclusion

This chapter presents the various kinds of enabling resources used in biomedical text mining applications. Lexicons support basic natural language processing tasks, such as parsing. Along with terminologies, lexicons also provide lists of names (including variants) for biological entities, thus supporting entity recognition tasks. Finally, the relations represented in ontologies and terminologies often serve as a reference for relation extraction algorithms.

Because it integrates these three kinds of resources, the UMLS plays a central role in biomedical text mining. Its three components (Specialist lexicon, Metathesaurus, and Semantic Network) are illustrated in this chapter, showing their use in entity recognition and relation extraction tasks. The role of other resources, either more specialized or more general, is also discussed.

Despite the existence of these resources, there remain many challenges to entity recognition and relation extraction in biology. Existing biomedical lexicons and terminologies fail to provide adequate coverage of specialized subdomains (e.g., genes and proteins for the various model organisms). Approaches to normalizing the names of genomic entities and to resolving the ambiguity introduced by some of them need to be further researched. Finally, the development of large, consistent, principled sources of biomedical knowledge—namely ontologies—will benefit not only text mining applications, but, more generally, the wide range of tasks relying upon biomedical knowledge (e.g., database interoperability, decision support, and so forth).

Acknowledgments

The author would like to thank Tom Rindflesch, Pierre Zweigenbaum, and Karin Verspoor for useful comments on a previous version of this manuscript.

References

[1] de Bruijn, B., and J. Martin, "Getting to the (C)Ore of Knowledge: Mining Biomedical Literature," *Int. J. Med. Inform.*, Vol. 67, No. 1–3, 2002, pp. 7–18.

[2] Shatkay, H., and R. Feldman, "Mining the Biomedical Literature in the Genomic Era: An Overview," *J. Comput. Biol.*, Vol. 10, No. 6, 2003, pp. 821–855.

[3] Yandell, M. D., and W. H. Majoros, "Genomics and Natural Language Processing," *Nat. Rev. Genet.*, Vol. 3, No. 8, 2002, pp. 601–610.

[4] Hahn, U., M. Romacker, and S. Schulz, "Medsyndikate—A Natural Language System for the Extraction of Medical Information from Findings Reports," *Int. J. Med. Inform.*, Vol. 67, No. 1–3, 2002, pp. 63–74.

[5] Corney, D. P., et al.,"BioRAT: Extracting Biological Information from Full-Length Papers," *Bioinformatics*, Vol. 20, No. 17, 2004, pp. 3206–3213.

[6] Leroy, G., H. Chen, and J. D. Martinez, "A Shallow Parser Based on Closed-Class Words to Capture Relations in Biomedical Text," *J. Biomed. Inform.*, Vol. 36, No. 3, 2003, pp. 145–158.

[7] Humphreys, K., G. Demetriou, and R. Gaizauskas, "Two Applications of Information Extraction to Biological Science Journal Articles: Enzyme Interactions and Protein Structures," *Proc. Pacific Symp. on Biocomputing*, 2000, pp. 505–516.

[8] Friedman, C., et al., "A General Natural-Language Text Processor for Clinical Radiology," *J. Am. Med. Inform. Assoc.*, Vol. 1, No. 2, 1994, pp. 161–74.

[9] Friedman, C., et al., "GENIES: A Natural-Language Processing System for the Extraction of Molecular Pathways from Journal Articles," *Bioinformatics*, Vol. 17, Suppl. 1, 2001, pp. S74–82.

[10] Rindflesch, T. C., M. Fiszman, and B. Libbus, "Semantic Interpretation for the Biomedical Literature," *Medical Informatics: Advances in Knowledge Management and Data Mining in Biomedicine*, H. Chen, et al., (eds.), Berlin: Springer-Verlag, 2005, pp. 399–422.

[11] Grabar, N., et al., "Matching Controlled Vocabulary Words," *Stud. Health Technol. Inform.*, Vol. 95, 2003, pp. 445–450.

[12] Bodenreider, O., and C. A. Bean, "Relationships among Knowledge Structures: Vocabulary Integration Within a Subject Domain," *Relationships in the Organization of Knowledge*, C. A. Bean, and R. Green, (eds.), Amsterdam: Kluwer, 2001, pp. 81–98.

[13] Nenadic, G., I. Spasic, and S. Ananiadou, "Terminology-Driven Mining of Biomedical Literature," *Bioinformatics*, Vol. 19, No. 8, 2003, pp. 938–943.

[14] Smith, B., et al., "Relations in Biomedical Ontologies," *Genome Biol.*, Vol. 6, No. 5, 2005, p. R46.

[15] Smith, B., J. Williams, and S. Schulze-Kremer, "The Ontology of the Gene Ontology," *Proc. AMIA Annual Symp.*, 2003, pp. 609–613.

[16] Bodenreider, O., "The Unified Medical Language System (UMLS): Integrating Biomedical Terminology," *Nucleic Acids Res.*, Vol. 32, Database issue, 2004, pp. D267–D270.

[17] Baser, M. E., D. G. Evans, and D. H. Gutmann, "Neurofibromatosis 2," *Curr. Opin. Neurol.*, Vol. 16, No. 1, 2003, pp. 27–33.

[18] Uppal, S., and A. P. Coatesworth, "Neurofibromatosis Type 2," *Int. J. Clin. Pract.*, Vol. 57, No. 8, 2003, pp. 698–703.

[19] Fellbaum, C., *WordNet: An Electronic Lexical Database*, *Language, Speech, and Communication*. Cambridge, MA: MIT Press, 1998.

[20] Bodenreider, O., A. Burgun, and J. A. Mitchell, "Evaluation of WordNet as a Source of Lay Knowledge for Molecular Biology and Genetic Diseases: A Feasibility Study," *Stud. Health Technol. Inform.*, Vol. 95, 2003, pp. 379–384.

[21] Burgun, A., and O. Bodenreider, "Comparing Terms, Concepts and Semantic Classes in WordNet and the Unified Medical Language System," *Proc. NAACL2001 Workshop, WordNet and Other Lexical Resources: Applications, Extensions and Customizations*, 2001, pp. 77–82.

[22] Leroy, G., and H. Chen, "Meeting Medical Terminology Needs—The Ontology-Enhanced Medical Concept Mapper," *IEEE Trans. on Inf. Technol. Biomed.*, Vol. 5, No. 4, 2001, pp. 261–270.

[23] Browne, A. C., et al., "UMLS Language and Vocabulary Tools," *Proc. AMIA Annual Symp.*, 2003, p. 798.

[24] Krauthammer, M., and G. Nenadic, "Term Identification in the Biomedical Literature," *J. Biomed. Inform.*, Vol. 37, No. 6, 2004, pp. 512–526.

[25] Wain, H. M., et al., "Genew: The Human Gene Nomenclature Database, 2004 Updates," *Nucleic Acids Res.*, Vol. 32, Database issue, 2004, pp. D255–D257.

[26] Pustejovsky, J., et al., "Automatic Extraction of Acronym-Meaning Pairs from MEDLINE Databases," *Medinfo*, Vol. 10, No. 1, 2001, pp. 371–375.

[27] Schwartz, A. S., and M. A. Hearst, "A Simple Algorithm for Identifying Abbreviation Definitions in Biomedical Text," *Proc. Pacific Symp. on Biocomputing*, 2003, pp. 451–462.

[28] Wren, J. D., et al., "Biomedical Term Mapping Databases," *Nucleic Acids Res.*, Vol. 33, Database Issue, 2005, pp. D289–D293.

[29] Chute, C. G., "Clinical Classification and Terminology: Some History and Current Observations," *J. Am. Med. Inform. Assoc.*, Vol. 7, No. 3, 2000, pp. 298–303.

[30] Ashburner, M., et al., "Gene Ontology: Tool for the Unification of Biology: The Gene Ontology Consortium," *Nat. Genet.*, Vol. 25, No. 1, 2000, pp. 25–29.

[31] McCray, A. T., A. C. Browne, and O. Bodenreider, "The Lexical Properties of the Gene Ontology," *Proc. AMIA Annual Symp.*, 2002, pp. 504–508.

[32] Nelson, S. J., D. Johnston, and B. L. Humphreys, "Relationships in Medical Subject Headings," *Relationships in the Organization of Knowledge*, C. A. Bean and R. Green, (eds.), Amsterdam: Kluwer, 2001, pp. 171–184.

[33] Aronson, A. R., "Effective Mapping of Biomedical Text to the UMLS Metathesaurus: The Metamap Program," *Proc. AMIA Annual Symp.*, 2001, pp. 17–21.

[34] McCray, A. T., "An Upper-Level Ontology for the Biomedical Domain," *Comp. Funct. Genom.*, No. 4, 2003, pp. 80–84.

[35] Bodenreider, O., and A. T. McCray, "Exploring Semantic Groups Through Visual Approaches," *J. Biomed. Inform.*, Vol. 36, No. 6, 2003, pp. 414–432.

[36] Rindflesch, T. C., and M. Fiszman, "The Interaction of Domain Knowledge and Linguistic Structure in Natural Language Processing: Interpreting Hypernymic Propositions in Biomedical Text," *J. Biomed. Inform.*, Vol. 36, No. 6, 2003, pp. 462–477.

[37] Libbus, B., et al., "Using Natural Language Processing, LocusLink, and the Gene Ontology to Compare OMIM to MEDLINE," *Proc. HLT-NAACL Workshop on Linking the Biological Literature, Ontologies and Databases: Tools for Users*, 2004, pp. 69–76.

[38] McCray, A. T., and O. Bodenreider, "A Conceptual Framework for the Biomedical Domain," *The Semantics of Relationships: An Interdisciplinary Perspective*, R. Green, C. A. Bean, and S. H. Myaeng, (eds.), Boston, MA: Kluwer Academic Publishers, 2002, pp. 181–198.

[39] Rosse, C., and J. L. Mejino, Jr., "A Reference Ontology for Biomedical Informatics: The Foundational Model of Anatomy," *J. Biomed. Inform.*, Vol. 36, No. 6, 2003, pp. 478–500.

[40] Distelhorst, G., et al., "A Prototype Natural Language Interface to a Large Complex Knowledge Base, the Foundational Model of Anatomy," *Proc. AMIA Annual Symp.*, 2003, pp. 200–204.

[41] Sneiderman, C. A., T. C. Rindflesch, and C. A. Bean, "Identification of Anatomical Terminology in Medical Text," *Proc. AMIA Annual Symp.*, 1998, pp. 428–432.

[42] Bean, C. A., T. C. Rindflesch, and C. A. Sneiderman, "Automatic Semantic Interpretation of Anatomic Spatial Relationships in Clinical Text," *Proc. AMIA Annual Symp.*, 1998, pp. 897–901.

[43] Nowlan, W. A., et al., "From Terminology to Terminology Services," *Proc. Annual Symp. on Computer Applications in Medical Care*, 1994, pp. 150–154.

[44] Bodenreider, O., and A. Burgun, "Biomedical Ontologies," *Medical Informatics: Advances in Knowledge Management and Data Mining in Biomedicine*, H. Chen, et al., (eds.), Berlin: Springer-Verlag, 2005, pp. 211–236.

[45] McCray, A. T., et al., "Evaluating UMLS Strings for Natural Language Processing," *Proc. AMIA Annual Symp.*, 2001, pp. 448–452.

[46] Srinivasan, S., et al., "Finding UMLS Metathesaurus Concepts in MEDLINE," *Proc. AMIA Annual Symp.*, 2002, pp. 727–731.

[47] Ogren, P. V., et al., "The Compositional Structure of Gene Ontology Terms," *Proc. Pacific Symp. on Biocomputing*, 2004, pp. 214–525.

[48] Mungall, C., "Obol: Integrating Language and Meaning in Bio-Ontologies," *Comparative and Functional Genomics*, Vol. 5, No. 7, 2004, pp. 509–520.

[49] McCray, A. T., S. Srinivasan, and A. C. Browne, "Lexical Methods for Managing Variation in Biomedical Terminologies," *Proc. Annual Symp. on Computer Applications in Medical Care*, 1994, pp. 235–239.

[50] Morgan, A. A., et al., "Gene Name Identification and Normalization Using a Model Organism Database," *J. Biomed. Inform.*, Vol. 37, No. 6, 2004, pp. 396–410.

[51] Collier, N., and K. Takeuchi, "Comparison of Character-Level and Part of Speech Features for Name Recognition in Biomedical Texts," *J. Biomed. Inform.*, Vol. 37, No. 6, 2004, pp. 423–435.

[52] Koike, A., Y. Niwa, and T. Takagi, "Automatic Extraction of Gene/Protein Biological Functions from Biomedical Text," *Bioinformatics*, Vol. 21, No. 7, 2005, pp. 1227–1236.

[53] Proux, D., et al., "Detecting Gene Symbols and Names in Biological Texts: A First Step Toward Pertinent Information Extraction," *Genome Inform. Ser.*, Vol. 9, 1998, pp. 72–80.

[54] Yu, H., et al., "Automatic Extraction of Gene and Protein Synonyms from MEDLINE and Journal Articles," *Proc. AMIA Annual Symp.*, 2002, pp. 919–923.

[55] Tanabe, L., and W. J. Wilbur, "Generation of a Large Gene/Protein Lexicon by Morphological Pattern Analysis," *J. Bioinform. Comput. Biol.*, Vol. 1, No. 4, 2004, pp. 611–26.

[56] Chute, C. G., et al., "The Content Coverage of Clinical Classifications: For the Computer-Based Patient Record Institute's Work Group on Codes & Structures," *J. Am. Med. Inform. Assoc.*, Vol. 3, No. 3, 1996, pp. 224–233.

[57] Harris, M. R., et al., "A Term Extraction Tool for Expanding Content in the Domain of Functioning, Disability, and Health: Proof of Concept," *J. Biomed. Inform.*, Vol. 36, No. 4–5, 2003, pp. 250–259.

[58] Bodenreider, O., T. C. Rindflesch, and A. Burgun, "Unsupervised, Corpus-Based Method for Extending a Biomedical Terminology," *Proc. ACL'2002 Workshop Natural Language Processing in the Biomedical Domain*, 2002, pp. 53–60.

[59] Jacquemin, C., *Spotting and Discovering Terms Through Natural Language Processing*, Cambridge, MA: MIT Press, 2001.

[60] Chen, L., H. Liu, and C. Friedman, "Gene Name Ambiguity of Eukaryotic Nomenclatures," *Bioinformatics*, Vol. 21, No. 2, 2005, pp. 248–256.

[61] Liu, H., Y. A. Lussier, and C. Friedman, "Disambiguating Ambiguous Biomedical Terms in Biomedical Narrative Text: An Unsupervised Method," *J. Biomed. Inform.*, Vol. 34, No. 4, 2001, pp. 249–261.

[62] Liu, H., V. Teller, and C. Friedman, "A Multi-Aspect Comparison Study of Supervised Word Sense Disambiguation," *J. Am. Med. Inform. Assoc.*, Vol. 11, No. 4, 2004, pp. 320–331.

[63] Kim, J. D., et al., "Genia Corpus—Semantically Annotated Corpus for Bio-Textmining," *Bioinformatics*, Vol. 19, Suppl. 1, 2003, pp. i180–i182.

[64] Ceusters, W., et al., "Ontology-Based Error Detection in SNOMED-Ct(R)," *Medinfo*, Vol. 2004, 2004, pp. 482–486.

[65] Smith, B., A. Kumar, and S. Schulze-Kremer, "Revising the UMLS Semantic Network," *Medinfo*, Vol. 2004, No. CD, 2004, p. 1700.

[66] Smith, B., and C. Rosse, "The Role of Foundational Relations in the Alignment of Biomedical Ontologies," *Medinfo*, Vol. 2004, pp. 444–448.

[67] Hahn, U., and S. Schulz, "Towards a Broad-Coverage Biomedical Ontology Based on Description Logics," *Proc. Pacific Symp. on Biocomputing*, 2003, pp. 577–588.

[68] Wroe, C. J., et al., "A Methodology to Migrate the Gene Ontology to a Description Logic Environment Using DAML+OIL," *Proc. Pacific Symp. on Biocomputing*, 2003, pp. 624–635.

[69] Hearst, M. A., "Automatic Acquisition of Hyponyms from Large Text Corpora," *Proc. 15th Int. Conf. on Computational Linguistics (COLING-92)*, 1992, pp. 539–545.

[70] Bodenreider, O., M. Aubry, and A. Burgun, "Non-Lexical Approaches to Identifying Associative Relations in the Gene Ontology," *Proc. Pacific Symp. on Biocomputing*, 2005, pp. 104–115.

[71] Johnson, S. B, "A Semantic Lexicon for Medical Language Processing," *J. Am. Med. Inform. Assoc.*, Vol. 6, No. 3, 1999, pp. 205–218.

[72] Verspoor, K., "Towards a Semantic Lexicon for Biological Language Processing," *Comparative and Functional Genomics*, Vol. 6, No. 1–2, 2005, pp. 61–66.

4

Automatic Terminology Management in Biomedicine

Sophia Ananiadou and Goran Nenadic

4.1 Introduction

The dynamic and rapidly evolving field of biomedicine makes the processing and management of terminology one of the key factors for accessing the information stored in literature, since information across scientific articles is conveyed through terms and their relationships. Without knowledge of the terminology of a domain, the understanding of scientific documents is hindered. In this chapter, we examine how text mining can facilitate terminology management in biomedicine. We examine the basic concepts of term and terminology, the relation between concepts and terms, and issues related with term variation and term ambiguity. We also present approaches for automatically recognizing and structuring terminology from literature, and for discovering associations between terms that can be used for terminology management.

4.1.1 Principles of Terminology

Any domain is organized around a network of concepts and associations between these concepts. We define "term" as the linguistic realization of a specialized concept in a given domain. For example, biomedical terms refer both to concepts, such as gene, protein, disease, interaction, and inhibition; as well as to names, such as gene products, organisms, drugs, chemical compounds, and so forth. The main purpose of terms, as opposed to words, is the classification of

specialized knowledge: "The classification principle […] becomes the chief motivation in designation" [1]. Terms are means of scientific communication and are used by specialists to convey knowledge within a domain. They delineate the knowledge space of a domain.

We refer to the collection of terms belonging to a specific domain as a terminology. A terminology may be structured (i.e., organized in a hierarchy), denoting different types of relationships among concepts, such as specific-generic, whole-part, and so forth. Thus, a terminology is not merely a list of lexical items used in a specific domain, but it also includes a representational structure.

When we construct a terminology, we link concepts and their definitions with terms. The acceptance of a term by a specific community depends on how well this linking is carried out, which ensures communication among specialists. The introduction of a new term assumes the prior existence of a concept that points to a specific area of the domain knowledge space [2–4]. Typically, this process includes a definition that positions a concept (as a discrete entity) within a domain-specific knowledge space, and relates it to other concepts via the use of relations. The establishment of these relations, and the grouping and association of concepts as a result, constitute the classification of a term. Terms are classified according to the properties of their corresponding concepts, and not according to their linguistic or lexical properties. Although terms point to a specific knowledge space, they are merely labels of knowledge. They are not themselves knowledge items, but they refer to the concepts that are the knowledge items. Naming involves the use of domain-specific term formation patterns to label a concept introduced by a definition [5]. This process establishes a designator or a term that is used to refer to the concept in question. Naming is carried out within the boundaries of linguistic, social, and domain norms. In all cases, efforts have to be made to ensure that specialists in the field accept a term.

Term formation patterns are specific to a given domain. In general, new terms are created by:

1. Using existing linguistic or terminological resources that narrow, widen, or adjust the meaning of an existing word-form by using simile (e.g., *window*, *table, mouse*, in computer science). This type of term creation often uses metaphors, explores polysemy and homonymy in language, and causes ambiguity.
2. Modifying existing term resources using transformations, such as affixation (*promyelocyt-ic, leuke-mia*), compounding (*protein kinase C*), abbreviations (*RAR*), and so forth. For example, the vast majority of terms in biomedicine are multiword compounds, typically consisting of nouns, adjectives, and acronyms.

3. Creating new linguistic entities (neologisms). This approach is very common in biomedicine. Neologisms can be completely new inventions or may borrow heavily from other languages, typically Greek and Latin (e.g., a protein named after a Chinese breakfast noodle: *yotiao*). Neologisms are often created by using some modification [e.g., a combination of words and numerals (*annexin II mRNA*), letters, symbols (*Ca_2^+-calmodulin-dependent protein*), and eponyms (*Jurkat T cells*)].

Guidelines for the naming process in the biomedical domain have been produced for many types of concepts. For example, HUGO (http://www.gene.ucl.ac.uk/nomenclature/) has provided guidelines and a nomenclature for human gene names and symbols for each known human gene. Similar nomenclatures exist for other species (e.g., bovine, yeast, farmed animals, and so forth). However, specialists do not always follow them. Thus, not all new terms created necessarily follow the existing term formation patterns and the guidelines promulgated by formal bodies. There have been huge efforts to collect and promote the usage of standardized and shared terminologies (e.g., SNOMED CT, UMLS, HUGO, and so forth). Standardized and unique names are important for domain specialists to communicate and share knowledge. They are also important for text processing, since their use facilitates information retrieval, natural language processing, and integration between different knowledge sources.

Mapping between terms and concepts is not trivial. Ideally, terminology theory defines a term as uniquely designating a concept, and vice versa. In practice, this monoreferential correspondence is far from reality. The relation between terms and concepts is many-to-many: the same term may refer to a number of concepts, and the same concept may be referred to by many terms [3]. The phenomenon when a single concept is linked to several terms (or term forms) is called term variation [6]. An example of term variation is the orthographic variation *oestrogen—estrogen*, or the variation between *cancer* and *carcinoma*. Term variation is a very productive phenomenon, especially in rapidly evolving domains such as biomedicine. Furthermore, in such evolving domains, it is common that a term is introduced and is subsequently found to denote several distinct concepts [7]. The phenomenon when the same term refers to many concepts is known as term ambiguity. An example of term ambiguity is the term *promoter*. In biology the term refers to "a binding site in a DNA chain at which RNA polymerase binds to initiate transcription of messenger RNA by one or more nearby structural genes," while, in chemistry, it refers to "a substance that in very small amounts is able to increase the activity of a catalyst." Another example is the fact that many proteins have dual names that also denote the protein family to which they belong. For instance, the term *cycline-dependent kinase*

inhibitor was first introduced to represent a protein family with only one extension, *p27.* However, it is now used interchangeably with *p27* or *p27kip1* as the name of the individual protein, and not as the name of the protein family [7, 8]. Furthermore, in the case of *NFKB2*, the term is used to denote the name of a family of two individual proteins with separate entries in Swiss Prot. These proteins are homologous, which means that they belong to different species—human and chicken. These examples demonstrate that it is not always easy to establish the link between terms and the concepts that they denote. Furthermore, due to the evolving nature of biomedicine, concepts are often not fully delineated, since they are themselves evolving. This dynamism is particularly notable in biomedicine, and is reflected in the degree of term variation and ambiguity observed. For these reasons, terminological processing remains as one of the main challenges in biomedical text mining [9–11].

In the next section, we provide a brief overview of existing terminological resources in biology (see Chapter 3), and point to their limitations for supporting text mining.

4.2 Terminological Resources in Biomedicine

More than 300 biological information resources containing an abundance of nomenclatures and ontologies are publicly available [12]. Many have been used in attempts to recognize terms in literature in order to support access to biological information and text mining (as discussed in Section 4.4). For example, sources commonly used for this purpose are the UMLS (http://www.nlm.nih.gov/research/umls/) and the Universal Protein Resource (UniProt) (http://www.uniprot.org/). The UMLS merges information from more than 100 biomedical vocabularies, and currently contains over 1 million concepts and 2.8 million terms. The concepts are organized into a hierarchy of 135 classes, and connected by 54 different types of relations. Although the UMLS Metathesaurus mainly focuses on conceptual information, rather than on lexical and terminological data, it additionally includes and cross-links some term variants (mainly abbreviations, inflectional and upper/lower case variation). UniProt contains extensive curated protein information (almost 1.9 million entries), including function, classification, and cross references, as well as alternative (synonymous) protein names. Other important biomedical resources include: SNOMED CT (http://www.snomed.org/snomedct), a clinical reference terminology that contains 357,000 health care concepts annotated by more than 960,000 terms; the GO (http://www.geneontology.org/), which contains approximately 17,000 concept descriptions covering molecular function, biological processes, and cellular components; and the UMLS Specialist Lexicon (http://specialist.nlm.nih.gov/LexiconDescription.html) [13], which contains

mainly lexical information (spelling variants, parts-of-speech, morphological, and syntactic information) on words and terms used in the biomedical domain. These resources are discussed in detail in Chapter 3.

However, many studies have suggested that information available in existing biomedical resources is not sufficient for text mining applications, and that substantial reengineering is needed to tailor the resources for automatic processing. For example, for UMLS, we refer the reader to [14–17]. Friedman et al. [18] combined the UMLS and its Specialist Lexicon to acquire lexical knowledge for medical and clinical processing. They generated lexical entries from the Specialist Lexicon for all represented UMLS concept names from specific categories, but reported that this automatic, straightforward approach of acquiring lexical information did not improve the performance of their system. They also pointed out that term ambiguity (in particular for abbreviations) was responsible for many errors in term identification. Despite a huge number of entities, some clinical terms (or their variants) were not present in the UMLS, and many of the recognized terms do not appear, either because a given term form and/or concept is missing, or available resources do not represent these types of entities (e.g., terms that refer to families or group of proteins [19]). We summarize these issues as they relate to existing biomedical resources, by enumerating problems not only for text mining, but also for manual curation, knowledge discovery, and knowledge integration.

1. *Resources are focused on human users.* Human specialists, who are able to resolve many types of term variations and ambiguities, use the majority of resources for knowledge integration. Such resources are not designed for automated processing. This is especially true for biomedical ontologies. For example, concept descriptions in an ontology are frequently confused with domain terms. Many of the problems reported in the literature arise when attempts are made to consult an ontology by finding concept descriptions in running text. Moreover, some ontology concepts may have no "natural" terms associated with them (e.g., placeholder concepts), but still may be useful for human specialists.
2. *Limited lexical and terminological coverage of biological subdomains.* Many terminological and lexical resources are more focused on medicine than on biology. The full range of terminologies, covering genotype, genes, proteins, protein structure and function, tissue types, species, and disease types, is only partially covered.
3. *Update and curation of resources.* The rate of biological term creation impedes manual update of resources. Even if a terminological resource exists for a given subdomain, many concepts and terms that appear in literature are typically missing. The dynamic nature of data makes the

number of names prone to change. Therefore, for any resource to be useful, it should be semiautomatically updated, and providing sustainable ways to update and curate existing resources is one of the main challenges.

4. *Naming conventions and representation in heterogeneous resources.* Many databases do not use uniformly the official names and symbols provided by, for example, HUGO and the International Protein Index (IPI). When integrating different terminologies, the problem of the same concept being represented in different ways has to be addressed. Although some cross references exist, many problems still remain in relation to communication between and integration of different databases.
5. *Terminological variation and complexity of names.* Biological names are very complex. They include an enormous amount of synonyms and different variant term forms that are used in literature. Most terms are used with synonyms or other variants, such as acronyms, morphological and derivational variations, and so forth. Existing terminological resources do not cover all such variants.
6. *Term ambiguity.* Many biological terms and their variants are ambiguous. They share their lexical representations with either common English words (e.g., gene names and abbreviations, such as *an*, *by*, *can*, and *for*); or with other biomedical terms (as indicated above). Existing resources typically lack information that can support disambiguation of terms. Terminological resources also do not address ambiguities related to finer biological classifications, such as species information. For example, homologue genes have the same name, but belong to different species.

To overcome these problems, techniques for automatic terminology management are needed to meet the demands not only of text mining systems, but also of human specialists. In particular, techniques for automatic recognition of terms in text, and methods for linking concepts with term forms (in text) are necessary. These techniques also are essential for systematic collection and update of terminological data, and for the integration and linking of heterogeneous resources.

4.3 Automatic Terminology Management

Terminology management is concerned with the creation, storage, maintenance, updating, and curation of terminologies. Terminology management

systems may be manual, semiautomatic, or automatic. An automatic terminology management system is viewed as incorporating:

- An automatic term recognition (ATR) module, which recognizes and extracts lexical units from text that corresponds to domain concepts;
- An automatic term structuring (ATS) module, which organizes terms within a terminology by assigning specific relationships among terms, typically by using classification and clustering;
- An intelligent term manager (ITM) module, which stores terminological data in a suitable repository. ITM also may link terminological entries with respective factual databases that provide additional information (e.g., definitions, links to the corresponding documents, or factual databases).

An ATR module typically processes literature to recognize, suggest, validate, and extract individual terms and their variants. An ATS module uses lexical, statistical, grammatical, contextual, or other features to discover possible relationships among terms and then organize the terminology accordingly. Different approaches can be applied, including extraction of specific relationships, term classification, term clustering, automatic thesaurus/ontology construction, term annotation, and so forth. Finally, an ITM module stores and manages all the terminological data, and provides integration of literature and factual databases. In the following sections, we review existing approaches to term recognition (see Section 4.4), term structuring (see Section 4.6), and term management (see Section 4.7).

4.4 Automatic Term Recognition

Identification of terms in a document consists of three tasks [20] (see Figure 4.1). The first step is term recognition, in which the detection of single or multiple adjacent words indicates the presence of domain concepts. In the second step, recognized terms are classified into broader domain classes, such as genes,

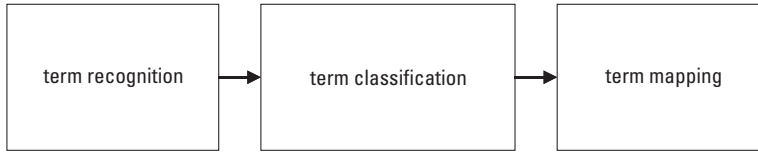


Figure 4.1 Steps in term identification in text.

proteins, or tissues. The final task is the mapping of terms into domain terminologies, which involves the linking of terms and the corresponding concepts. Such concepts are typically represented in a domain terminological resource, and mapping should establish a link from a term occurrence in text to a specific entry in the resource. In this section, we will concentrate on the first of these tasks.

Automatic term recognition denotes the process used to systematically extract pertinent terms and their variants from a collection of documents. Its main aim is to distinguish terms of a subject field from nonterms [21]. However, the majority of approaches in the biomedical domain integrate term recognition and term classification in one single step.

The techniques for ATR include dictionary-based, rule-based, statistical, and machine learning approaches, as well as their combinations. For extensive overviews related specifically to biomedical ATR, we refer the reader to [20, 22].

4.4.1 Dictionary-Based Approaches

Dictionary-based approaches for ATR use existing terminological resources in order to locate term occurrences in text. Each sequence of words in text that matches an entry in a terminological resource is considered as a term occurrence; only such strings are treated as terms. However, given the amount of neologisms and variations, many term occurrences cannot be recognized in text if straightforward dictionary/database look-up is used. This reduces the sensitivity of ATR systems. In contrast, term ambiguity affects the accuracy. For example, when names from FlyBase were used as a terminological source for recognition of gene names in literature, the results have shown an extremely low precision (2% for full articles and 7% for abstracts), with recall in the range from 31% for abstracts, to 84% for full articles [23].

Some ATR approaches combine dictionaries with additional processing to support the term recognition process. For example, edit-distance operations (i.e., substitution, deletion, and insertion of characters and digits) can be used to implement more flexible string matching against a dictionary of protein terms [24, 25]. In order to address the peculiarities of biomedical terms, the cost function for edit operations can be tuned. For example, substitution of a space with a hyphen (or vice versa) is considerably less expensive than substitution of any other two different characters. A similar method was used to recognize gene and protein names and their variations by approximate string comparison based on mapping both protein dictionaries and target documents using the nucleotide code (i.e., a four-letter encoding over the {A, C, G, T} alphabet) [26]. Techniques used for alignment of protein sequences in databases are applied to the input text, in order to identify text character sequences that are similar to

existing gene and protein names, which are also encoded by the corresponding nucleotide codes.

4.4.2 Rule-Based Approaches

Rule-based approaches generally use term formation patterns. The general approach is to develop rules that describe common naming structures for certain term classes, using either orthographic, lexical, or more complex morpho-syntactic features. Consequently, each sequence of words or morphemes in text that can be described by a rule is considered as a term occurrence; only such strings are treated as terms. Dictionaries of typical term constituents (e.g., terminological heads,[1] class-specific adjectives, affixes, or specific acronyms) often are used to assist in term recognition.

A general grammar-based methodology for the recognition of medical terminology was suggested by Ananiadou [27], in which a four-level ordered morphology was proposed to describe formation patterns. The system used a morphological unification grammar, and a lexicon with instances of specific affixes, roots, and Greek/Latin neoclassical combining forms. Gaizauskas et al. [28] used a context-free grammar for the recognition of protein names. They used morphological features (suffixes) and hand-constructed grammar rules for each terminological class of interest (e.g., enzymes and proteins). Fukuda et al. [29] relied mainly on simple lexical rules (patterns) and orthographic features for the recognition of protein names. Their system, PROPER, distinguishes between core and feature terms. Core terms are words that usually bear the core meaning, while feature terms are keywords that describe the function and characteristics of terms (e.g., *protein*, *receptor*, and so forth). For example, in *SAP kinase*, *SAP* is the core term, while *kinase* is the feature term. Nouns and/or adjectives between core and feature terms are concatenated by application of simple extension rules to identify protein names. Many systems have been influenced by PROPER [30, 31].

Rule-based approaches are usually difficult to tune to different domains or classes, since rules are mostly domain-specific. Alternatives are machine learning and statistical approaches.

4.4.3 Machine Learning Approaches

Machine learning (ML) systems are usually designed for a specific class of entities. They use training data to learn features that are useful and relevant for term recognition and classification. Then, depending on the techniques used, each

1. The notion of terminological head refers to the element in a term that confers termhood to the whole term.

sequence of words in text that fulfills fitness criteria (based on learned features) is considered as a term occurrence (of a given term class); only such strings are treated as terms. The main challenge here is to select a set of representative features that can be used for accurate recognition and classification of term instances. Another challenge is detection of term boundaries of multiword terms that are the most difficult to learn. Finally, the existence of reliable training resources is one of the main problems, since these resources are not widely available (see Chapter 8).[2]

Several ML techniques have been used for the identification and classification of terms, including HMMs, naïve Bayesian approaches, SVMs, and decision trees. For example, Collier et al. [32] used HMMs and specific orthographic features (e.g., "consisting of letter and digits," or "having initial capital letter") for identifying and classifying terms; while Kazama et al. [33] used multiclass SVMs with a feature set, including lexical information, POS tags, and affix information. In order to train the classifier, a training corpus was annotated with "B-I-O" tags. B-tags denote words that are at the beginning of a term, I-tags denote words that are inside a term, and O-tags denote words outside a term. The tags were also complemented with the appropriate class information (i.e., a 'B-PROTEIN' tag denotes a word that is at the beginning of a protein name). Yamamoto et al. [34] further combined boundary features (based on morpheme-based tokenization) with morpho-lexical features (e.g., POS tags or stems); biomedical features (e.g., whether a given word exists in a compiled database of biomedical resources); and syntactic features (e.g., head morpheme information). They reported that, individually, biomedical features were crucial for recognition of protein names.

4.4.4 Statistical Approaches

Statistical approaches are based on various statistical distributions of collocations in text. The challenge here is to define adequate measures of termhood of candidate terms (i.e., their likelihood of representing domain-specific concepts). Since the vast majority of terms are noun phrases, the main strategy in many systems is to extract specific noun phrases as term candidates, and then to estimate their termhoods. Often, term candidates are ranked according to their likelihood of being terms. Occurrences of candidate terms whose termhood is above a certain threshold are considered as terms; only such strings are treated as terms. Statistical techniques often are more easily tuned to different domains, since they avoid

2. Few terminologically tagged biomedical corpora are available (e.g., the GENIA corpus, see Chapter 8), since it is very time consuming to produce them manually. Thus, one of the major challenges for ML approaches is the automated creation of tagged corpora.

the problems of designing specific rules, feature selections, and unavailability of training data.

4.4.5 Hybrid Approaches

Hybrid approaches combine several techniques for term recognition. An example of a hybrid method that combines statistics with linguistic preprocessing is the C/NC-value method [35], which has been used to recognize multiword terms in several biomedical subdomains (e.g., nuclear receptors and yeast). Term candidates are suggested by a set of morpho-syntactic filters, while their termhoods are estimated by a corpus-based statistical measure. The measure (called C-value) amalgamates four numerical characteristics of a candidate term: (1) the frequency of occurrence, (2) the frequency of occurrence as a substring of other candidate terms (in order to tackle nested terms), (3) the number of candidate terms containing the given candidate term as a substring, and (4) the number of words contained in the candidate term. This approach facilitates the recognition of nested (embedded) subterms, which is particularly important in the biomedical domain. The selected list of term candidates is further refined by taking into account the context of candidate terms (NC-value).

A second example of a hybrid system is the protein and gene name tagger, ABGENE [36]. The system combines machine learning, transformation rules, and a dictionary-based approach with statistical and probabilistic comparisons. The protein tagger that has been trained on MEDLINE abstracts by adapting a general language POS tagger obtains the initial results. An additional set of transformation rules for the recognition of single-word gene and protein names also is used. Compound names are extracted by a set of regular expressions, which rely on the combinations of frequently occurring components in known multiword gene names. These initial results are then filtered to discard falsely recognized protein names, while attempting to recover missed instances. Filtering is based on an extensive list of precompiled nonbiological terms (obtained by comparing word frequencies in MEDLINE with a general language corpus), and general (i.e., nongene and nonprotein) biomedical terms. In contrast, missed names are targeted with an extensive list of proteins and genes. Context words also are consulted. If a word is in a "good" context that typically appears with proteins and genes, then it is tagged accordingly. A probabilistic algorithm has generated good context words from the training set.

Context clues also can be used as negative knowledge, in order to recognize terms or term boundaries [37]. More precisely, every sequence of words between two boundary words can be considered as a candidate term, where boundary words can be carefully defined by a list containing common English stop words and some domain specific expressions [38].

4.4.6 Conclusion

Performance of ATR methods in the biomedical domain varies. Precision is typically in the 70% to 90% range, while recall is approximately 70%. A recent evaluation exercise of protein name taggers, for example, has shown that the best precision values are just above 80%, with maximum recall at 80% (at lower precision points) [39]. Rule-based approaches typically perform better than other approaches, but suffer from problems of customization to new domains and classes. The performance of ML systems depends on the existence of sufficient training data [36], while the performance of statistically-based systems is limited due to smaller corpora, but can be substantially enhanced by the inclusion of normalization techniques (see Section 4.5).

Finally, the majority of ATR approaches focus only on recognizing isolated, individual occurrences in text, and not on linking equivalent term forms and integrating them with/to existing reference databases. Mapping terms to databases as the final step of the term identification process is essential for further text mining, where acquired knowledge on specific biomedical concepts is aggregated across different data sources [20]. In order for mapping to be successful, resolving terminological variations and ambiguities is necessary. We discuss these issues in the following section.

4.5 Dealing with Term Variation and Ambiguity

Systematic identification of terms in text faces two major problems: the extensive variability of lexical term representations, and the problem of term ambiguity. We discuss these in the following sections.

4.5.1 Term Variations

Along with official and standard names, a range of variants, synonyms, and alternative names is used to refer to biomedical concepts. A particular concept can be denoted by various surface realizations, which are called term variants. For example, *TIF2*, *TIF-2*, *transcription intermediary factor-2*, and *transcriptional intermediate factor 2* are all used to denote the same concept. Usually, one of these term variants is considered as the preferred term for terminology management purposes. Term variation is very frequent: approximately one-third of term occurrences are variants [21]. Since terms may appear in many different forms, term variation is an essential part of term identification and management.

Terminological variation also is related to language diversity, and usually is used to make text more readable by applying stylistic variations. The use of variants is typically "in proportion to the density of terms over a given text

segment" [1]. The introduction of a term variant typically assumes that domain specialists are able to recover the preferred term from a variant by applying variation techniques. Nevertheless, term variation may cause problems not only for automatic term recognition but for human experts as well.

We use a broad definition of variability, where the meaning of the term does not change under variation. This covers simple variations, such as spelling differences, as well as more complex variations (including synonymy).

Extensive studies have been conducted to analyze variability in biology terms. For example, Cohen et al. [40] differentiated between contrastive features, which can be used as clues to distinguish terms that have different meanings, and noncontrastive variability in synonymous gene and protein names. They found case insensitivity and optionality of hyphens and parentheses to be noncontrastive, while "edge effects" (e.g., a number at the last position of a protein name) were contrastive, [i.e., changing the meaning (identity) of a term]. The simplest but most frequent variations are related to punctuation differences (*bmp-4* and *bmp4*), usage of different numerals (*syt4* and *syt iv*), or different transcriptions of Greek letters (*iga* and *ig alpha*) [41, 42].

We consider the following types of term variation: orthographic, morphological, lexical, and structural variations. We also consider acronyms (see Table 4.1).

Orthographic variants include alternative usage of hyphens and slashes (*amino acid* and *amino-acid*), lower and upper cases (*NF-KB* and *NF-kb*), different Latin/Greek transcriptions (*oestrogen* and *estrogen*), other spelling variations (*tumour* and *tumor*), and so forth.

Table 4.1
Term Variation Types and Examples

Type of Variation	Example Variants
Orthographic	*9-CIS-retinoic acid* and *9-cis retinoic acid*
	amyloid beta-protein and *amyloid* β*-protein*
Morphological	*nuclear receptor* and *nuclear receptors*
	Down's syndrome and *Down syndrome*
Lexical	*hepatic leukaemia factor* and *liver leukemia factor*
	human cancer and *human carcinoma*
Structural	*cancer in humans* and *human cancers*
	SMRT and Trip-1 RNAs and *SMRT RNA and Trip-1 RNA*
Acronyms	*RAR alpha, RAR-alpha, RARA, RARa, RA receptor alpha*
	NF-kappaB, NF(kappa)B, kappaB, NFKB factor, NF-KB

Morphological variants are mainly related to simple inflectional phenomena (plural and possessive forms). Derivation can lead to variants in some cases (*cellular gene* and *cell gene*), but not always (*activated factor* versus *activating factor*, which are different terms).

Lexical variants are genuine synonyms, which may be used interchangeably (*carcinoma* and *cancer*, *haemorrhage* and *blood loss*). A special subclass of lexical variants includes the usage of related synonym-like pairs (*hepatic* and *liver*, *cardiac* and *heart*) in multiword terms (*cardiac disease* and *heart disease*).

Structural variants are more complex. They involve transformations in term structure, such as simple permutations (*integrin alpha 4* and *alpha4 integrin*); possessive usage of nouns using prepositions (*clones of humans* and *human clones*); prepositional variants (*cell in blood* and *cell from blood*); term coordination (*adrenal glands and gonads*); and so forth. In the case of structural variation, the term variants involved are multiword units. Special subtypes also may include insertions (*adenovirus 5* and *adenovirus type 5*), but in the majority of cases, insertions typically generate a specialization of a term (*blood mononuclear cell* and *blood cell* [21]).

Acronyms and abbreviations are very frequent term variation phenomena used as synonyms for the corresponding full-form terms. For example, a recent study [43] reported that only 25% of documents relevant to the concept denoted by the term *c-jun N-terminal kinase* could be retrieved by using the full form, since more than 33% of the documents refer to the concept by using its acronym, *JNK*. Discovering acronyms and relating them to their expanded forms is an essential aspect of text mining and terminology management. In general, selecting the first (or first few) letters of the words from the corresponding full form forms the acronym. There are no formal rules or guidelines for the coinage of new acronyms. Therefore, acronyms are equally subjected to variation and ambiguity. Some acronyms are synonymous—the same term may have several acronyms (*NF kappa B* and *NF kB* for *nuclear factor kappa B*). Other acronyms are polysemous (ambiguous)—the same acronym may correspond to different terms (*GR* is used as an acronym for *glucocorticoid receptor* and for *glutathione reductase*). Both phenomena present substantial challenges for terminology management and for text mining (see Chapter 5).

These variation phenomena are frequently combined (e.g., lexical and morphological variation, structural and morphological variation, acronym and morphological variation, and so forth). The process of composing variations is studied in detail in [21].

Since concepts can be represented by a set of term variants, terms can be viewed as equivalence classes of term variants. An equivalence class (called synterm [44]) contains all term forms that are used to denote a given concept. From each equivalence class, we select one term form as the preferred term, which acts as the concept label (see Table 4.2). ATR now can be viewed as the

Table 4.2
Preferred Terms and Associated Term Equivalence Classes (Synterms)

Preferred Terms for Concepts	Term Forms
all trans retinoic acid	all trans retinoic acid, all-trans-retinoic acids, ATRA, at-RA
nuclear receptor	nuclear receptor, nuclear receptors, NR, NRs
9-cis retinoic acid	9-c-RA, 9cRA, 9-cis-retinoic acid, 9-cis retinoic acid
deoxyribonucleic acid	DNA, DNAs, deoxyribonucleic acid
nuclear factor kappa B	NF-KB, NF-kb, nuclear factor kappa B, NF-kappaB
human cancer	human cancers, cancer in humans, human's cancer, human carcinoma

task of recognizing equivalence term classes, rather than individual term forms [44]. For terminology management purposes, it is critical that the preferred term be an attested term form, occurring naturally in text. Normally, a dictionary entry form is chosen as the preferred term.

For example, in order to arrive at the equivalence class for *human cancer* in Table 4.2, we have to deal with not only orthographic, morphological, and lexical variants, but also with structural variations. To do this, we need to carry out linguistic normalization of the individual term variants using a set of transformations [44] (see also Section 4.7).

The simplest approach to handle some types of term variation (e.g., morphological) is based on stemming. If two term forms share a stemmed representation, then they can be considered as belonging to the same equivalence class [45, 46]. However, stemming may result in overstemming (i.e., leading to the conflation of terms that are not real variants), or understemming (i.e., leading to failure to link real term variants).

We can also recognize and link term variants in text by combining terms and their synonyms from existing terminologies with approximate string matching and edit distance techniques (as indicated in Section 4.4). These techniques also can be augmented with probabilistic or machine learning methods to filter false positive variation pairs [25], or to automatically generate acronyms from definitions [47]. Another example is the MetaMap program [48], which maps noun phrases identified by the Specialist minimal commitment parser to UMLS Metathesaurus concepts. Handling terminological variation supports mapping terms from text into corresponding data sources. MetaMap uses a multilevel mapping strategy, which first analyzes a target term to generate a multitude of possible variants, including acronyms, synonyms, and inflectional variants. Each of these variants is then mapped against concept names in the

Metathesaurus. The method compares the strength of the mapping for each term variant, ordering possible mapping candidates.

Other approaches rely essentially on rules that describe variations. For example, FASTR [21] uses several hundred metarules dealing with morphological, syntactic, and semantic term variation. Here, term variation recognition is based on the transformation of basic term structures into variant structures. Syntactic metarules address the transformation of term structures, and do not involve inflectional, morphological, derivational, or semantic relationships. However, they may contain additional constraints (e.g., POS tags) that a sequence has to fulfill in order to be considered as a candidate variation. Some of the variants recognized by FASTR are more conceptual than terminological, since nonterminological units (e.g., verb phrases, extended insertions, and so forth) also are linked to terms in order to improve indexing and retrieval.

Lastly, there has been work towards direct identification (i.e., mining) of term synonyms in documents. Various techniques can be utilized. For example, Yu and Agichtein [49] experimented with unsupervised, partially supervised, and supervised ML approaches, as well as with a rule-based system, for the extraction of gene and protein synonyms that occurred within the same sentence. The unsupervised ML approach was based on the comparison of mutual information of synonym candidates with respect to other words in their neighboring contexts, while the partially supervised, bootstrap method used a set of seed synonym occurrences to learn contexts that indicated occurrence of synonyms. The supervised SVM-based method used the same seed occurrences to learn a classifier that classified the text surrounding a pair of gene/protein names as a synonym or not. Finally, the rule-based system was based on a set of manually defined lexical patterns (e.g., fragments such as *<GENE>, also known as <GENE>*) that indicated typical contexts used to express synonymy.

4.5.2 Term Ambiguity

In addition to variability, terminological ambiguity is a further challenge for automatic term management. Terms are frequently associated with multiple meanings, mainly for reasons of terminological economy. This is typically the case when the same term is used to denote several different views or aspects of a concept (so-called systematic ambiguity or term multidimensionality). For example, an occurrence of the *CAT* protein can be associated with several different protein entries in a protein data source, depending on the species in question. However, some terms (in particular acronyms) can have multiple independent meanings (e.g., *CAT* can be a protein, animal, or medical device). Disambiguation methods typically rely on contextual analysis of a given occurrence, mainly using various machine learning (i.e., classification) strategies to decide which term sense is correct in the given context. For example,

Hatzivassiloglou et al. [50] experimented with a variety of techniques (e.g., naïve Bayesian and decision trees) to disambiguate gene and protein names and their transcripts (mRNA, tRNA). They used contexts of known occurrences to learn weights for contextual elements. When an unknown occurrence is encountered, they apply these weights to the new contextual elements. They experimented with a variety of contextual features, such as morphological, syntactic, distributional, and so forth. Pakhomov [51] used a maximum-entropy classifier by using only two neighboring words left and right to find a correct interpretation of a given ambiguous acronym. He also experimented with features based on document layout, particularly headings (titles) of the sections of clinical reports in which ambiguous acronyms appeared. In many cases, a narrow context may not always be enough to disambiguate a term (e.g., when a protein name is shared between species); a wider context (e.g., a whole article) may need to be analyzed before terms can be mapped. Liu et al. [52] aimed at selecting a correct sense of a term (typically an acronym) that is associated with several entries in the UMLS. Based on other (unambiguous) UMLS concepts that are associated (through the Metathesaurus) with each of the given senses of the target term, the method builds a classifier for each sense of the term.

In addition to disambiguation between various biomedical meanings, a problem is that many biological names share their lexical representation with common English words (e.g., gene names, such as *an*, *by*, *can*, and *for*; see [23]).

4.6 Automatic Term Structuring

Automatic term recognition is one of the components of term management. Once we identify terms and link their variants together, we need to relate them with each other (i.e., to organize them into knowledge structures). As we mentioned in the introduction of this chapter, terms belong to a delimited knowledge space, and in order to situate them within this knowledge space, we need to establish and identify their relationships with other terms. Capturing relationships also is useful for users wishing to acquire, situate, and visualize concepts or groups of concepts [53, 54].

Term structuring typically includes classification and clustering, which are keys to knowledge organization. In many cases, direct links between terms (e.g., semantic networks) are of special interest (e.g., protein-protein interactions, annotation of genes with relevant GO terms, and so forth).

Several methods have been suggested for the extraction of relationships from literature. For detailed overviews in the biomedical domain, we refer the reader to [55, 56]. The most straightforward approach to establishing term links is to measure lexical similarity among the words that constitute terms [57, 58]. For example, a term derived by modifying another term typically indicates

concept specialization (*orphan nuclear receptor* is a kind of *receptor*), or some specific functional relationship (*CREP binding protein* is linked to *CREP* through the binding relationship [59, 60]). Many term classification systems rely on such functional words (*receptor*, *factor*, or *radical*) for assigning term categories [28, 29, 31]. More often than not, terms do not contain any explicit term category information. In such situations, additional lexical processing is needed. For example, functional words can be combined with conditional probabilities of words that are indicative of specific classes [61]. Jacquemin [6] experimented with linking terms associated by specific syntactic variation links that reflect internal term structures. Nenadic et al. [59] generalized these approaches by considering all possible overlapping subterms, with an additional weight given to the similarity between terms if they have common heads. However, lexical similarities are not always straightforward, due to a high rate of neologisms and ad hoc names, which rarely aid the encoding of particular functional properties of the underlying concepts in a systematic manner. For example, experiments have shown that only 5% of the most frequent terms from the GENIA corpus that belong to the same biomedical class have some lexical links [42].

Therefore, term relationships are typically extracted, using analysis of context in which term instances tend to appear. Contexts may be determined as an entire abstract or document, a sentence, or a phrase. Ding et al. [62] investigated the effectiveness of these contexts, based on a term-term cooccurrence measure. They reported that larger units naturally provided better recall, while smaller units (e.g., phrases) typically delivered significantly better precision.

Various methods are used to uncover associations among terms. Typically, some type of similarity is established using the contexts of term occurrences. One of the main problems is selecting the best discriminating textual features that show similarities among terms. A traditional approach is based on the idea that entities that cooccur in text with a given term (e.g., within the same sentence, paragraph, or document) may be indicative of its function, and that terms with similar cooccurrence distribution profiles have related roles or convey related meanings. For example, many approaches rely on simple cooccurrence patterns of words [54, 62] or other terms [38, 63] that appear next to the terms in question. Similarly, Jenssen et al. [64] assigned weights to relationships between genes based on the frequency of their mutual cooccurrence. Other statistical measures that assess how strongly terms are related to each other also are used (e.g., mutual information). However, statistical systems typically do not identify the type of relationship, but rather indicate a connection or association among entities. Furthermore, cooccurrences and statistical distributions within larger text units (e.g., documents) may not reveal significant links for some types of relationships. For example, many studies reported that even 40% of cooccurrence–based relationships were biologically meaningless [64, 65].

Therefore, other features that consider more linguistic and terminological aspects need to be considered.

Rule-based approaches for extracting term relationships rely on predefined patterns whose instances are captured in text. They include simple lexical patterns [66, 67], (semantic) templates [68, 69], and domain-specific grammars that are used to parse sentences [70–72]. For example, hyponymic and taxonomic relationships can be extracted by using general language lexico-syntactic patterns, such as term enumerations (*[...] steroid receptors such as estrogen receptor, glucocorticoid receptor, and progesterone receptors*) [59]. More domain-specific relationships can be extracted by special patterns and rules, which extract a predefined type of relationship (e.g., binding, activation, and so forth). Since relationship-specific rules are handcrafted, this significantly prolongs the construction of a terminology mining system, reduces its adaptability, and excludes term relationships that do not correspond to the predefined patterns and templates. Thus, some approaches are aimed at automatic mining [59, 73] and bootstrapping [74] of contextual patterns, which can be used for estimating the similarities among terms.

To cope efficiently with the complexity of knowledge needed to perform reliable relationship extraction and term classification, many approaches use machine learning techniques to learn features that characterize specific relationships or term classes. In addition to the systems discussed in Section 4.4.3, see [54, 75–81]. For instance, an SVM approach with word features was used to classify genes with respect to their subcellular location [54]. Term sources, both internal (functional words and suffixes) and external (words occurring nearby), can be combined for the classification task [82]. Spasic et al. [83] suggested an alternative ML approach that uses case-based reasoning to classify terms. Classification experience is collected in the form of term contexts indicative of a given class. New terms are then classified by flexible comparison of their contexts against the collection of contexts, and corresponding classes are selected. Finally, several ML methods have been used for the annotation of biomedical entities. For example, a word-based maximum entropy measure [84], as well as SVM classification with terminological features [85], were used for the annotation of genes and proteins with GO codes. The main problems related to ML methods are that they require large amounts of training data and need to be periodically retrained upon the advent of new data. They typically underperform for small (minority) classes, due to data sparseness.

As indicated in the previous examples, various approaches are combined to classify, cluster, or link terms. For example, Koike et al. [86] used cooccurrences, collocation similarities, and rule-based techniques to automatically extract gene and protein biological process functions; while Nenadic et al. [59] combined lexical, syntactic, and contextual similarities to estimate overall similarity among

terms. It has been shown that by combining various aspects of term similarities, accuracy of suggested associations typically improves.

4.7 Examples of Automatic Term Management Systems

Few systems have been designed and developed to facilitate automatic terminology management and integration in the biomedical domain. Various approaches have been suggested to automatically integrate and map between resources (e.g., between GO and UMLS, using flexible string matching [87, 88]). For example, the TERMINO system [11] provides a database schema for storing large-scale and diverse biomedical word lists collected from various resources (including UMLS, GOA, and in-house ontologies). It attempts to establish and maintain links between resources of various types, including ontology concept names, terms, controlled vocabularies, nomenclatures, classification descriptors, and so forth. Its specific focus is to provide storage efficiency and flexibility, and to provide a dictionary-based look-up mechanism. Despite its name, TERMINO is not focused on terminology, since it has no means to discriminate between terms and nonterms. In addition, morphologic and orthographic term variations are not included in the database. Instead, they are treated by an external general processing component, which reduces its ability to deal with particular domain phenomena. These attempts revealed the difficulties inherent in the integration of biological terminologies, which were due mainly to extensive term variation, and to term ambiguity with respect to mapping into a data source. For example, attempts to integrate gene names in UMLS were not successful, since they increased ambiguity.

While these approaches aim at collecting and integrating existing resources, Automatic Term Recognition and Clustering of Terms (ATRACT) is a terminology management workbench in the biomedicine domain that integrates automatic recognition of terms and extraction of their associations [89]. Its main purpose is to help biologists and terminologists gather and structure domain-specific terminology. It automatically retrieves and clusters terms on-the-fly, as well as facilitates integration of different textual sources.

The core ATRACT modules (see Figure 4.2) are: an ATR module (C/NC-value module), an acronym recognition module, and a term clustering module (ClusTerm). Additional external modules (e.g., a POS tagger, shallow parser, and a clustering program) are invoked. Communication among all modules is based on the exchange of XML-tagged documents.

Automatic term recognition in ATRACT is carried out using an enhanced version of the C/NC-value method [35], which incorporates term variation and acronym recognition [44]. The overall ATR process is presented in Figure 4.3. The first step is to extract term candidates using general term formation

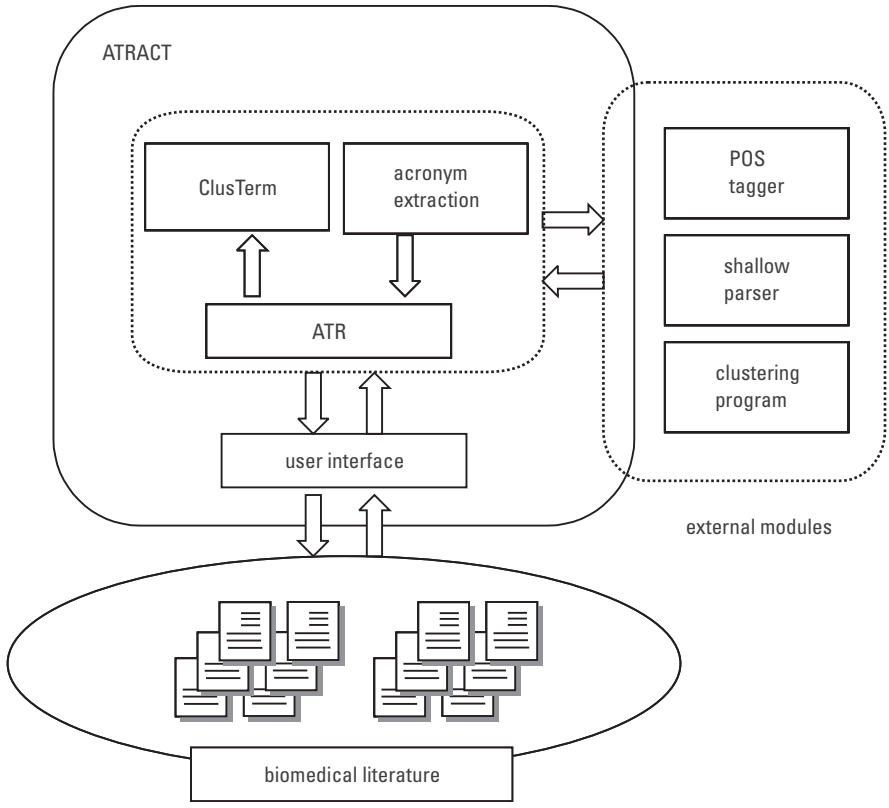


Figure 4.2 The ATRACT architecture.

patterns, mostly using noun phrases, and to transform them into their normalized form, using a set of rules. Normalization facilitates conflation of equivalent surface expressions. In ATRACT, we normalize orthographic, morphological, lexical, prepositional, and coordinated terms. An acronym recognition and conflation module [90] is used to link acronyms and their variants to their respective preferred terms. All equivalent term forms are then grouped into synterms. The estimation of termhoods (i.e., calculation of C/NC-values) for synterms is then carried out for the whole synterm, rather than for individual term candidates. This approach ensures that all term variants are dealt with jointly, thus supporting the fact that they denote the same concept. Finally, synterms are ranked according to their termhoods, and presented to the user. It has been shown that the incorporation of some term variation types is very useful for boosting precision and recall of ATR. For example, acronyms and inflectional unification can increase precision by up to 70%, while recall can improve by up to 25% [44]. However, other term variation types typically have only marginal influence on performance, but can be still important for many text

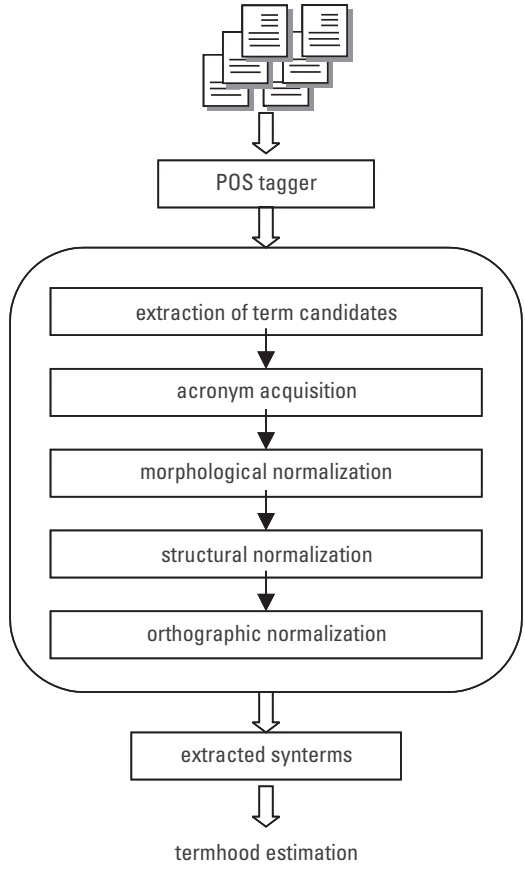


Figure 4.3 The ATR process in ATRACT.

mining tasks, such as information retrieval, information extraction, term and document clustering and classification, as well as for systematic collection of terminology.

Extraction of term relationships in other approaches is typically based either on their mutual co-occurrences in text, or on automatic mining of their similarities. Relationships based on co-occurrences are assessed through standard mutual information [63, 89]. In contrast, ATRACT uses a more linguistically- and terminologically-based approach to discover term similarities, by combining lexical, syntactic, and contextual similarities [59]. Lexical similarity is based on the degree of constituent words shared between terms. It is calculated by considering substrings (i.e., possible nested terms) that are shared among terms, with additional credits given to the similarity if the two terms have a common terminological head. This type of similarity typically captures hyponymy and meronymy relationships, as well as some specific biomedical relationships (e.g.,

the binding relationship between *CREP* and *CREP binding protein*). Syntactic similarities rely on specific lexico-syntactic expressions (such as term enumerations and conjunctions), in which a sequence of terms appears as a single syntactic unit. All terms appearing in such expressions are used in combination with the same verb or preposition, and typically share similar functions. Finally, contextual similarities are based on automatic discovery of relevant contexts shared among terms, in which terms appear individually, not as part of an enumeration or term conjunction. For example, terms such as *receptor*, *estrogen receptor*, *RXR*, and *TR* appear in textual contexts that can be roughly described using the pattern 'TERM VERB:bind CLASS:dna'. By collecting and comparing such contexts, some functional links among these terms can be discovered. Extensive experiments performed for these similarities have shown that lexical and syntactic links have high precision but low recall, while contextual similarities result in significantly higher recall with moderate precision. By combining the three similarity measures, we can improve both precision and recall of extracted term relationships. These relationships are further used in ATRACT to establish clusters of related terms. Some examples are presented in Table 4.3.

A typical way of navigating through the textual resources via ATRACT is that the domain expert, whose interest is expressed by a set of key terms, first retrieves a set of documents. Then, after having the corpus POS tagged and the basic syntactic chunks marked, the core ATRACT modules process it. Terms (including variants and acronyms) are automatically recognized and annotated in the corpus. They are consequently highlighted in the text, and by clicking on them, the user can get all the information (e.g., corresponding synterms and links to their occurrences) extracted from the documents. Furthermore, term

Table 4.3
Example of Term Clusters

Cluster ID	Cluster Elements
1	*basal transcription machinery, basal transcription, transcription activation transcriptional activation, transcriptional interference, transcriptional repressor, transcriptional silence, transcriptional activity*
2	*cell proliferation, HL-60 cell, breast cancer cell line, cancer cell, breast cancer cell, breast cancer patient, breast cancer, primary breast cancer*
3	*hRAR alpha, RXR alpha, RXR homodimers, TR/RXR heterodimers*
4	*RA response element, RA response, response element, responsive element*
5	*glucocorticoid receptor, mutant androgen receptor, receptor complex, mineralocorticoid receptor, progesterone receptor, estrogen receptor, human estrogen receptor, retinoid receptor, retinoid x receptor, nuclear receptor family, nuclear receptor*

similarities are calculated for the extracted terms, and they are clustered and presented to the user. After terminological processing, the user may keep the results in the form of a repository of terms in which they are interested, along with all collected data. The results can be used to generate a controlled vocabulary for indexing or searching. More importantly for text mining purposes, they can be used to update lexical, terminological, and ontological resources utilized by applications such as information extraction (see Chapter 7); and to drive conceptual annotation of corpora (see Chapter 8).

Users can tune the terminology mining process to meet their needs. Processing is guided by a set of parameters that can affect the term extraction and structuring processes. By tuning the parameters, the user can control the retrieval of different types of terms, as well as the amount of noise in the resulting set of terms. This allows the domain specialist to experiment with the results of the system according to their needs. By specifying a list of preexisting or known terms, ATRACT can optionally highlight only new terms, so that the user can concentrate on the acquisition/analysis of only those terms that would potentially expand their knowledge. Thus, ATR becomes an exploration method for bridging the gap between the existing knowledge of the user and available knowledge resources.

4.8 Conclusion

Given the amount of neologisms characterizing biomedical terminology, it is necessary to provide tools that will automatically extract newly coined terms from texts, and link them with databases and other controlled vocabularies. The importance of the topic has triggered significant research, which has resulted in several approaches used to collect, classify, and identify term occurrences in biomedical texts. The main aim of automatic terminology management is to provide tools to bridge the gap between terms as they occur in text and domain concepts, which further facilitates effective mining of scientific literature and integration with databases. This chapter has presented various methods for automatic terminological management: recognizing terms in text; linking various terminological variants; and structuring terms by extracting corresponding features from literature.

Naming conventions and internal characteristics of specific terminological classes, or contextual clues that are used for the recognition of sequences that represent specific domain concepts, are the basis for term recognition and classification methods. Features ranging from orthographic and morphological, to syntactic and contextual, are used. In addition to rules, different statistical measures and machine learning approaches are used for identification and structuring of terms.

There are still several open challenges requiring further research. In particular, in the area of term recognition, challenges include:

- Accurate recognition of term boundaries and further identification of internal term structure, which also can be useful for term classification;
- Treatment of various types of term variation and their integration into term identification;
- Anaphora resolution [91] and linking term coreferences;
- Selection of the most representative terms (and concepts) in a document.

This latter challenge holds out the hope of attaining sophisticated document indexing for improved information retrieval through term-based indexing (rather than using index terms). This is crucial for database curation and other annotation tasks.

Although much of the research is devoted to terms related to protein and gene names (see Chapter 6), the recognition of other classes of terms is vital for successful mining of the biomedical literature.

The recognition and collection of terms that correspond to domain concepts is not the ultimate goal of automatic term management. Structuring terms into classes, establishing their mutual links and associations, and mapping them to databases, is an additional part of term management. Particular challenges are the extraction and mining of specific term relationships from the literature and resolving term ambiguity.

Automatic term management aims to develop and apply efficient and accurate methods, both for building and updating terminological resources, and for supporting on-the-fly recognition of terms in text. Furthermore, leveraging term management (particularly terminology structuring) with experimental data derived by postgenomic techniques (such as expression array and sequence analysis) yields an opportunity to improve the performance of text-based approaches (see Chapter 10). For example, classification and clustering are improved by integrating and complementing features and relationships extracted from text with those from experimental databases [54].

To summarize, the processing and management of terminology is crucial for accessing information stored in literature, since information across scientific articles is conveyed through terms and their relationships. Without knowledge of the terminology, the understanding of documents is hindered. Since biomedical knowledge is expanding dynamically, it is very difficult to rely exclusively on manually developing terminologies. Therefore, automatic term management tools are indispensable for filling gaps in existing knowledge by suggesting possible additions to the biomedical terminology, and for supporting dynamic

literature mining and knowledge integration. Efficient term management is not only essential for the (semi)-automatic and systematic collection and updating of terminological data, but is also crucial for mining biomedical literature, and for the integrating with and linking of heterogeneous resources. Consistently and systematically derived terminological resources are vital for text mining, and substantial efforts should be dedicated to terminology processing, particularly in the biomedical domain. As a final point, we note that the field in general would benefit from a proper treatment of terminology, by which we mean that a clear distinction should be made between terms, ontology concept labels, controlled vocabulary items, index terms, and descriptors.

References

[1] Sager, J. C., *A Practical Course in Terminology Processing*, Amsterdam: John Benjamins, 1990.

[2] Kageura, K., "Multifaceted/Multidimensional Concept Systems," *Handbook of Terminology Management, Vol. 1: Basic Concepts of Terminology Management*, S. E. Wright, and G. Budin, (eds.), Amsterdam: John Benjamins, 1997, pp. 119–132.

[3] Meyer, I., K. Eck, and D. Skuce, "Systematic Concept Analysis Within a Knowledge-Based Approach to Terminology," *Handbook of Terminology Management*, *Vol. 1: Basic Concepts of Terminology Management*, S. E. Wright and G. Budin, (eds.), Amsterdam: John Benjamins, 1997, pp. 98–118.

[4] Sager, J. C., and M. -C. L'Homme, "A Model for the Definition of Concepts: Rules for Analytical Definitions in Terminological Databases," *Int. J. Terminology*, Vol. 1, 1994, pp. 351–374.

[5] Sager, J. C., "Term Formation," *Handbook of Terminology Management, Vol. 1: Basic Concepts of Terminology Management*, S. E. Wright and G. Budin, (eds.), Amsterdam, John Benjamins, 1997, pp. 25–41.

[6] Jacquemin, C., "Syntagmatic and Paradigmatic Representations of Term Variation," *Proc. 37th Annual Meeting of the Association for Computational Linguistics*, 1999, pp. 341–348.

[7] Tsujii, J., and S. Ananiadou, "Thesaurus or Logical Ontology: Which One Do We Need for Text Mining?" *Language Resources and Evaluation*, 2005, (in press).

[8] Morgan, A., et al., "Gene Name Extraction Using FlyBase Resources," *Proc. ACL Workshop, NLP in Biomedicine*, Sapporo, Japan, 2003, pp. 1–8.

[9] Ananiadou, S., "Challenges of Term Extraction in Biomedical Texts," *Proc. BioCreAtIvE Workshop*, 2004, http://www.pdg.cnb.uam.es/BioLink/workshop_BioCreative_04/handout/, accessed June 2004.

[10] Ananiadou, S., C. Friedman, and J. Tsujii, (eds.), "Named Entity Recognition in Biomedicine," *Journal of Biomedical Informatics*, Special Issue, Vol. 37, No. 6, 2004.

[11] Harkema, H., et al., "A Large-Scale Terminology Resource for Biomedical Text Processing," *Proc. BioLINK Workshop*, 2004, pp. 53–60.

[12] Hirschman, L., et al., "Accomplishments and Challenges in Literature Data Mining for Biology," *Bioinformatics*, Vol. 18, No. 12, 2002, pp. 1553–1561.

[13] Browne, A. C., A. T. McCray, and S. Srinivasan, *The Specialist Lexicon*, Bethesda, MD: Nat. Lib. Medicine, 2000.

[14] McCray, A. T., et al., "Evaluating UMLS Strings for Natural Language Processing," *Proc. AMIA Annual Meeting,* 2001, pp. 448–452.

[15] Liu, H., Y. Lussier, and C. Friedman, "A Study of Abbreviations in UMLS," *Proc. AMIA Annual Meeting*, 2001, pp. 393–397.

[16] Bodenreider, O., J. A. Mitchell, and A. T. McCray, "Evaluation of the UMLS as a Terminology and Knowledge Resource for Biomedical Informatics," *Proc. AMIA Annual Meeting*, 2002, pp. 61–65.

[17] Bodenreider, O., "The Unified Medical Language System (UMLS): Integrating Biomedical Terminology," *Nucleic Acids Res.*, Vol. 32, 2004, pp. 267–270.

[18] Friedman, C., et al., "Evaluating UMLS as a Source of Lexical Knowledge for Medical Language Processing," *Proc. AMIA Annual Meeting*, 2001, pp. 189–193.

[19] Blaschke, C., and A. Valencia, "Molecular Biology Nomenclature Thwarts Information-Extraction Progress," *IEEE Intelligent Systems*, Vol. 17, No. 3, 2002, pp. 73–76.

[20] Krauthammer, M., and G. Nenadic, "Term Identification in the Biomedical Literature," *Journal of Biomedical Informatics*, Special Issue on *Named Entity Recognition in Biomedicine*, Vol. 37, No. 6, 2004, pp. 512–526.

[21] Jacquemin, C., *Spotting and Discovering Terms Through NLP*, Cambridge MA: MIT Press, 2001.

[22] Ananiadou, S., "An Overview of Automatic Term Recognition Approaches in Biology," Pacific Symp. on Biocomputing, January 2001, http://www.ccs.neu.edu/home/futrelle/bionlp/psb2001tutorials.html/, accessed June 2005.

[23] Hirschman, L., A. Morgan, and A. S. Yeh, "Rutabaga by Any Other Name: Extracting Biological Names," *Journal of Biomedical Informatics*, Vol. 35, No. 4, 2002, pp. 247–259.

[24] Tsuruoka, Y., and J. Tsujii, "Probabilistic Term Variant Generator for Biomedical Terms," *Proc. 26th Annual ACM SIGIR Conference*, 2003, pp. 167–173.

[25] Tsuruoka, Y., and J. Tsujii, "Improving the Performance of Dictionary-Based Approaches in Protein Name Recognition," *Journal of Biomedical Informatics*, Special Issue on *Named Entity Recognition in Biomedicine*, Vol. 37, No. 6, 2004, pp. 461–470.

[26] Krauthammer, M., et al., "Using BLAST for Identifying Gene and Protein Names in Journal Articles," *Gene*, Vol. 259, No. 1–2, 2001, pp. 245–252.

[27] Ananiadou, S., "A Methodology for Automatic Term Recognition," *Proc. COLING-94,* Kyoto, Japan, 1994, pp. 1034–1038.

[28] Gaizauskas, R., G. Demetriou, and K. Humphreys, "Term Recognition and Classification in Biological Science Journal Articles," *Proc. Workshop on Computational Terminology for Medical and Biological Applications*, Patras, Greece, 2000, pp. 37–44.

[29] Fukuda, K., et al., "Towards Information Extraction: Identifying Protein Names from Biological Papers," *Proc. Pacific Symp. on Biocomputing*, Wailea, HI, 1998, pp. 707–718.

[30] Narayanaswamy, M., K. E. Ravikumar, and K. Vijay-Shanker, "A Biological Named Entity Recognizer," *Proc. Pacific Symp. on Biocomputing*, 2003, pp. 427–438.

[31] Frantzen, K., et al., "Protein Names and How to Find Them," *Int. J. Med. Inf.*, Vol. 67, No. 1–3, 2002, pp. 49–61.

[32] Collier, N., C. Nobata, and J. Tsujii, "Extracting the Names of Genes and Gene Products with a Hidden Markov Model," *Proc. COLING*, Saarbrücken, Germany, 2000, pp. 201–207.

[33] Kazama, J., et al., "Tuning Support Vector Machines for Biomedical Named Entity Recognition," *Proc. ACL Workshop on NLP in the Biomedical Domain*, Philadelphia, PA, 2002, pp. 1–8.

[34] Yamamoto, K., et al., "Protein Name Tagging for Biomedical Annotation in Text," *Proc. of ACL Workshop on NLP in Biomedicine,* Sapporo, Japan, 2003, pp. 65–72.

[35] Frantzi, K., S. Ananiadou, and H. Mima, "Automatic Recognition of Multi-Word Terms," *International Journal of Digital Libraries,* Vol. 3, No. 2, 2000, pp. 117–132.

[36] Tanabe, L., and W. J. Wilbur, "Tagging Gene and Protein Names in Biomedical Text," *Bioinformatics*, Vol. 18, No. 8, 2002, pp. 1124–1132.

[37] Bourigault, D., I. Gonzalez-Mullier, and C. Gross, "LEXTER, a Natural Language Processing Tool for Terminology Extraction," *Proc. EURALEX '96*, 1996, pp. 771–779.

[38] Blake, C., and W. Pratt, "Better Rules, Fewer Features: A Semantic Approach to Selecting Features from Text," *Proc. IEEE Data Mining Conference*, San Jose, CA, 2001, pp. 59–66.

[39] Yeh, A., et al., "BioCreAtIvE Task 1A: Gene Mention Finding Evaluation," *BMC Bioinformatics,* 2005, Vol. 6, Suppl. 1, p. S2.

[40] Cohen, K. B., et al., "Contrast and Variability in Gene Names," *Proc. ACL Workshop on NLP in the Biomedical Domain*, Philadelphia, PA, 2002, pp. 14–20.

[41] Tuason, O., et al., "Biological Nomenclature: A Source of Lexical Knowledge and Ambiguity," *Proc. Pacific Symp. on Biocomputing*, Big Island, HI, 2004, pp. 238–249.

[42] Nenadic, G., I. Spasic, and S. Ananiadou, "Mining Biomedical Abstracts: What's in a Term?" in *Natural Language Processing,* K.-Y. Su, et al., (eds.), *LNCS*, Vol. 3248, Berlin: Springer-Verlag, 2005, pp. 797–806.

[43] Wren, J. D., et al., "Biomedical Term Mapping Databases," *Nucleic Acid Research*, Vol. 33, Database Issue, 2005, pp. D289–D293.

[44] Nenadic, G., S. Ananiadou, and J. McNaught, "Enhancing Automatic Term Recognition Through Recognition of Variation," *Proc. COLING 2004*, Geneva, Switzerland, 2004, pp. 604–610.

[45] Jacquemin, C., and E. Tzoukermann, "NLP for Term Variant Extraction: A Synergy of Morphology, Lexicon and Syntax," *Natural Language Information Retrieval*, I. Strzalkowski, (ed.), Dordrecht: Kluwer, 1999, pp. 25–74.

[46] Ananiadou, S., S. Albert, and D. Schuhmann, "Evaluation of Automatic Term Recognition of Nuclear Receptors from MEDLINE," *Genome Informatics Series*, Vol. 11, 2000, pp. 450–451.

[47] Tsuruoka, Y., S. Ananiadou, and J. Tsujii, "A Machine Learning Approach to Automatic Acronym Generation," *Proc. ISMB BioLINK SIG: Linking Literature, Information, and Knowledge for Biology* and *13th Ann. Intl. Conf. on Intelligent Systems for Molecular Biology (ISMB 2005)*, Detroit, MI, 2005, pp. 25–31.

[48] Aronson, A. R., "Effective Mapping of Biomedical Text to the UMLS Metathesaurus: The MetaMap Program," *Proc. AMIA Annual Meeting*, 2001, pp. 17–21.

[49] Yu, H., and E. Agichtein, "Extracting Synonymous Gene and Protein Terms from Biological Literature," *Bioinformatics*, Vol. 19, Suppl. 1, 2003, pp. I340–I349.

[50] Hatzivassiloglou, V., P. A. Duboue, and A. Rzhetsky, "Disambiguating Proteins, Genes, and RNA in Text: A Machine Language Approach," *Bioinformatics,* Vol. 17, Suppl. 1, 2001, pp. 97–106.

[51] Pakhomov, S., "Semi-Supervised Maximum Entropy Based Approach to Acronym and Abbreviation Normalization in Medical Texts," *Proc. 40th ACL Conference*, 2002, pp. 160–167.

[52] Liu, H., S. B. Johnson, and C. Friedman, "Automatic Resolution of Ambiguous Terms Based on Machine Learning and Conceptual Relations in the UMLS," *J. Am. Med. Inform. Assoc.*, Vol. 9, No. 6, 2002, pp. 621–636.

[53] Mima, H., S. Ananiadou, and K. Matsushima, "Design and Implementation of a Terminology-Based Literature Mining and Knowledge Structuring System," *Proc. 3rd CompuTerm Workshop, Coling*, Geneva, Switzerland, 2004.

[54] Stapley, B. J., L. A. Kelley, and M. J. E. Sternberg, "Predicting the Sub-Cellular Location of Proteins from Text Using Support Vector Machines," *Proc. Pacific Symp. on Biocomputing*, 2002, pp. 374–385.

[55] Mack, R., and M. Hehenberger, "Text-Based Knowledge Discovery: Search and Mining of Life-Sciences Documents," *Drug Discovery Today*, Vol. 7, No. 11 (Suppl.), 2002, pp. S89–S98.

[56] Shatkay, H., and R. Feldman, "Mining the Biomedical Literature in the Genomic Era: An Overview," *Journal of Computational Biology,* Vol. 10, No. 6, 2003, pp. 821–856.

[57] Bourigault, D., and C. Jacquemin, "Term Extraction and Term Clustering: An Integrated Platform for Computer-Aided Terminology," *Proc. 8th Conf. European Association for Computational Lingustics*, Bergen, 1999, pp. 15–22.

[58] Yeganova, L., L. Smith, and W. J. Wilbur, "Identification of Related Gene/Protein Names Based on an HMM of Name Variations," *Comput. Biol. Chem.*, Vol. 28, No. 2, 2004, pp. 97–107.

[59] Nenadic, G., I. Spasic, and S. Ananiadou, "Mining Term Similarities from Corpora," *Terminology,* Vol. 10, No. 1, 2004, pp. 55–80.

[60] Ogren, P., et al., "The Compositional Structure of Gene Ontology Terms," *Proc. Pacific Symp. Biocomputing,* 2004, pp. 214–225.

[61] Nobata, C., N. Collier, and J. Tsujii, "Automatic Term Identification and Classification in Biological Texts," *Proc. Natural Language Pacific Rim Symp*., 1999, pp. 369–374.

[62] Ding, J., et al., "Mining Medline: Abstracts, Sentences, or Phrases?" *Proc. Pacific Symp. on Biocomputing*, 2002, pp. 326–337.

[63] Nenadic, G., et al., "Terminology-Based Literature Mining and Knowledge Acquisition in Biomedicine," *International Journal of Medical Informatics*, Vol. 67, No. 1–3, 2002, pp. 33–48.

[64] Jenssen, T., et al., "A Literature Network of Human Genes for High-Throughput Analysis of Gene Expressions," *Nature Genetics,* Vol. 28, 2001, pp. 21–28.

[65] Tao, Y., and R. Leibel, "Identifying Functional Relationships Among Human Genes by Systematic Analysis of Biological Literature," *BMC Bioinformatics*, 2002, pp. 3–16.

[66] Blaschke, C., et al., "Automatic Extraction of Biological Information from Scientific Text: Protein-Protein Interactions," *Proc. Intelligent Systems for Molecular Biology*, 1999, pp. 60–67.

[67] Ng, S., and M. Wong, "Toward Routine Automatic Pathway Discovery from On-Line Scientific Text Abstracts," *Genome Informatics*, Vol. 10, 1999, pp. 104–112.

[68] Maynard, D., and S. Ananiadou, "Term Extraction Using a Similarity-Based Approach," *Recent Advances in Computational Terminology*, D. Bourigault, C. Jacquemin, and M.-C. L'Homme, (eds.), *Natural Language Processing Series*, Vol. 2, Amsterdam: John Benjamins, 2001, pp. 261–279.

[69] Pustejovsky, J., et al., "Robust Relational Parsing Over Biomedical Literature: Extracting Inhibit Relations," *Proc. Pacific Symp. on Biocomputing*, 2002, pp. 362–373.

[70] Friedman, C., et al., "GENIES: A Natural-Language Processing System for the Extraction of Molecular Pathways from Journal Articles," *Bioinformatics* Vol. 17, No. 1, 2001, pp. S74–S82.

[71] Yakushiji, A., et al., "Event Extraction From Biomedical Papers Using a Full Parser," *Proc. Pacific Symp. on Biocomputing*, 2001, pp. 408–419.

[72] Gaizauskas, R., et al., "Protein Structures and Information Extraction from Biological Texts: The PASTA System," *Bioinformatics,* Vol. 19, No. 1, 2003, pp. 135–143.

[73] Nenadic, G., I. Spasic, and S. Ananiadou, "Terminology-Driven Mining of Biomedical Literature," *Bioinformatics*, Vol. 19, No. 8, 2003, pp. 938–943.

[74] Thelen, M., and E. Riloff, "A Bootstrapping Method for Learning Semantic Lexicons Using Extraction Pattern Contexts," *Proc. Empirical Methods in Natural Language Processing*, 2002.

[75] Craven, M., and J. Kumlien, "Constructing Biological Knowledge Bases by Extracting Information from Text Sources," *Proc. Intelligent Systems for Molecular Biology*, 1999, pp. 77–86.

[76] Marcotte, E., I. Xenarios, and D. Eisenberg, "Mining Literature for Protein-Protein Interactions," *Bioinformatics*, Vol. 17, No. 4, 2001, pp. 359–363.

[77] Donaldson, I., et al., "PreBIND and Textomy—Mining the Biomedical Literature for Protein-Protein Interactions Using a Support Vector Machine," *BMC Informatics*, Vol. 4, No. 11, 2003.

[78] Nenadic, G., et al., "Selecting Text Features for Gene Name Classification: From Documents to Terms," *Proc. ACL Workshop on NLP in Biomedicine*, Sapporo, Japan, 2003, pp. 121–128.

[79] Lee, K. J., et al., "Biomedical Named Entity Recognition Using Two-Phase Model Based on SVMs," *Journal of Biomedical Informatics*, Special Issue, *Named Entity Recognition in Biomedicine*, Vol. 37, No. 6, 2004, pp. 436–447.

[80] Takeuchi, K., and N. Collier, "Bio-Medical Entity Extraction Using Support Vector Machines," *Proc. ACL Workshop NLP in Biomedicine*, Sapporo, Japan, 2003, pp. 57–64.

[81] Spasic, I., and S. Ananiadou, "A Flexible Measure of Contextual Similarity for Biomedical Terms," *Proc. Pacific Symposium on Biocomputing*, Big Island, HI, 2005.

[82] Torii, M., S. Kamboj, and K. Vijay-Shanker, "An Investigation of Various Information Sources for Classifying Biological Names," *Proc. ACL Workshop NLP in Biomedicine*, Sapporo, Japan, 2003, pp. 113–120.

[83] Spasic, I., S. Ananiadou, and J. Tsujii, "MaSTerClass: A Case-Based Reasoning System for the Classification of Biomedical Terms," *Bioinformatics*, Vol. 21, No. 11, 2005, pp. 2748–2758.

[84] Raychaudhuri, S., et al., "Associating Genes with Gene Ontology Codes Using a Maximum Entropy Analysis of Biomedical Literature," *Genome Res*, Vol. 12, No. 1, 2002, pp. 203–214.

[85] Rice, S., G. Nenadic, and B. Stapley, "Mining Protein Function from Text Using Term-Based Support Vector Machines," *BMC Bioinformatics*, 2005, Vol. 6, Suppl. 1, p. S22.

[86] Koike, A., Y. Niwa, and T. Takagi, "Automatic Extraction of Gene/Protein Biological Functions from Biomedical Text," *Bioinformatics*, Vol. 21, No. 7, 2005, pp. 1227–1236.

[87] Cantor, M. N., et al., "An Evaluation of Hybrid Methods for Matching Biomedical Terminologies: Mapping the Gene Ontology to the UMLS," *Stud. Health Technol. Inform.* Vol. 95, 2003, pp. 62–67.

[88] Sarkar, I. N., et al., "Linking Biomedical Language Information and Knowledge Resources: GO and UMLS," *Proc. Pacific Symp. on Biocomputing*, 2003, pp. 427–450.

[89] Mima, H., S. Ananiadou, and G. Nenadic, "The ATRACT Workbench: Automatic Term Recognition and Clustering for Terms," *Text, Speech and Dialogue*, V. Matousek, et al., (eds.), Lecture Notes in Artificial Intelligence 2166, Heidelberg: Springer Verlag, 2001, pp. 126–133.

[90] Nenadic, G., I. Spasic, and S. Ananiadou, "Automatic Acronym Acquisition and Term Variation Management Within Domain-Specific Texts," *Proc. Language Resources and Evaluation Conf.*, 2002, pp. 2155–2162.

[91] Yang, X., et al., "Improving Pronoun Resolution by Incorporating Coreferential Information of Candidates," *Proc. ACL*, 2004, pp. 128–135.

5

Abbreviations in Biomedical Text

Jeff Chang and Hinrich Schütze

5.1 Introduction

Abbreviations, representing shortened forms of names of concepts, including diseases and genes, are prevalent in the biomedical literature. In the age of high-throughput genomic analyses, the rate at which new molecules are discovered and characterized is increasing, which is reflected in the ever-growing numbers of scientific articles. As a result, the number of abbreviations used in the literature also increases (see Figure 5.1). In the biomedical literature, 64,262 new abbreviations were introduced in 2004, and there is an average of one new abbreviation in every 5 to 10 abstracts [1].

Therefore, it is becoming increasingly difficult for researchers to keep up with known abbreviations. When encountering an unfamiliar abbreviation, one possible solution is to use a dictionary of abbreviations, such as the Dictionary of Medical Acronyms & Abbreviations [2]. However, such manual compilations suffer from several problems. First, dictionaries contain from 4,000 to 32,000 abbreviations [3], which is only a fraction of the approximately 800,000 believed to exist [1]. Second, due to the rapid and continuous creation of new abbreviations, "acronym dictionaries are outdated by the time they are published" [4]. Finally, dictionaries do not quantify the popularity of an abbreviation, which would be useful to help distinguish among possible different meanings [3].

Often, an abbreviation can be interpreted as several different definitions. For example, *ACE* can be either *angiotensin converting enzyme* or *affinity capillary*

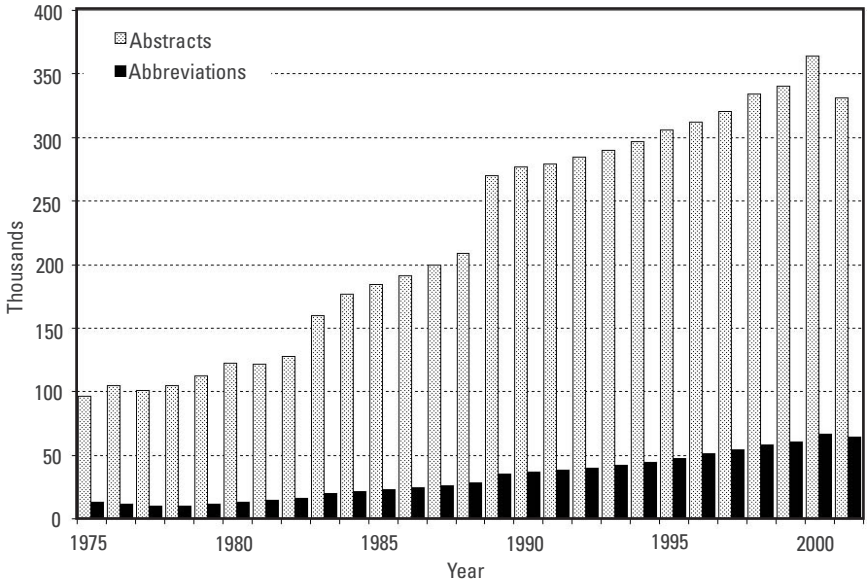


Figure 5.1 The number of new abbreviations and abstracts observed in MEDLINE each year is increasing.

electrophoresis [5]. If the abbreviation were not explicitly defined in the text, then its correct definition would depend on the content of the remainder of the text [6]. Some have observed that the number of definitions is growing faster than the number of abbreviations, indicating that this problem is growing more severe [3].

Despite the possible ambiguity in the meaning, abbreviations are an important component in information retrieval systems. Such systems identify documents from a database relevant to one or more keywords provided by the user. In searches for biomedical articles, how abbreviations are handled in the query clearly impacts the results. Search engines that explicitly exploit abbreviations, rather than full forms, in search terms yield more relevant documents [7].

Abbreviations are also an important component of text mining algorithms. In one study of computational approaches to generate a lexicon for biomedical natural language processing applications, the researchers noted that not handling abbreviations in the text is a major source of error [8]. In text, abbreviations function essentially as synonyms, where the abbreviation and its definition both refer to the same concept. Thus, a robust text processing system should understand and correctly handle them. Failure to handle such phenomena would reduce the amount of information available to the system.

Therefore, in the biomedical text processing community, there is great scientific interest in developing computational methods to identify and define

abbreviations in the literature. Most of the work has been focused on investigating methods to automatically find abbreviations in literature.

Abbreviation is a broad term that describes a shortened form of a word or phrase. The term acronym also is commonly used, and generally means a shortened form created from the initial letters of the words in the phrase. Some people also require acronyms to be pronounceable. In this chapter, we will refer to the most general problem as abbreviation identification, and we will consider an acronym to be a type of abbreviation.

There are two parts to an abbreviation, the short form (e.g., *NAT*) and the long form (e.g., *N-Acetyl Transferase*). The short form is sometimes called the abbreviation, and the long form can be called the definition or expansion. We use these terms interchangeably. Framed succinctly, abbreviation identification is the problem of analyzing unstructured text to extract pairs of <abbreviation, long form> that occur in the text.

There are many potential difficulties in developing methods to find abbreviations. In the simplest case, searching for words whose initial letters match the letters in the acronym can identify that acronym. However, many biomedical abbreviations deviate from this pattern, causing complications in the design of robust automated algorithms (see Table 5.1).

In addition to varied patterns of abbreviation formation, a more subtle and challenging problem stems from the fact that there is disagreement on the actual definition of abbreviation. There has not yet been a systematic attempt to define an abbreviation and to document differing notions of what constitutes an abbreviation. In an experiment where multiple experts were asked to identify the

Table 5.1
Varied Patterns Used to Abbreviate Long Forms

Abbreviation	Definition	Description
VDR	vitamin D receptor	The letters match the initial character of each word.
PTU	propylthiouracil	The letters match at boundaries of syllables.
JNK	c-Jun N-terminal kinase	The letters match at punctuation marks.
IFN	interferon	The letters match some other place.
SULT	sulfotransferase	The abbreviation contains contiguous characters from a word.
ATL	adult T-cell leukemia	The long form contains words not in the abbreviation.
CREB-1	CRE binding protein	Letters in the abbreviation do not occur in the long form.
beta-EP	beta-endorphin	The abbreviation contains complete words.

abbreviations in identical documents, the experts disagreed from 6% to 12% of the time. There were three main areas of disagreements: differing definitions of abbreviations, disagreement on the boundaries of long forms, and overlooked abbreviations (i.e., human error) [6].

One clear disagreement is whether aliases (i.e., two phrases that are not lexically related to each other), are abbreviations. For example, *apoptosis* is an alias for *programmed cell death*. We use the term alias instead of synonym, because alias includes a wider range of relations, such as hyponymy and hyperonymy. Some experts would consider that a type of abbreviation—two terms that have the same meaning, even though they do not share characters. Medstract, a corpus of MEDLINE abstracts where the acronym-meaning pairs were annotated by an expert, includes 13 aliases out of a total of 168 abbreviations [4]. Many would not consider aliases to be abbreviations. Algorithms to identify aliases are fundamentally different than algorithms that can rely on the fact that most abbreviations are shortened versions of the long form.

Another more subtle area of disagreement arises from abbreviations that are not strictly synonyms of their long forms. Instead, their meaning may be either more general or more specific (i.e., hypernyms or hyponyms). For example, in *HOT-SPOT (HOT1)*, the abbreviation indicates a specific variant of the HOT-SPOT gene, and thus HOT1 is a hyponym of the long form. However, many would still consider it an abbreviation because it is a shortened form of the long form.

Finally, there is considerable ambiguity in the boundary of the long form. For example, applying a straightforward letter matching heuristic to *RNA Polymerase I (Pol I)* would yield the long form *Polymerase I*. However, based on biological knowledge, many would consider the complete long form to be *RNA Polymerase I*. Similarly, experts in anatomy would consider the phrase *lateral arcuate nucleus*, rather than just *arcuate* or *arcuate nucleus*, to be the long form for the abbreviation *Arc*. These disagreements are based on expert knowledge of the use and meaning of the word. To correctly identify these long forms, algorithms need to include considerably more sophisticated processing based on knowledge or typical use of the abbreviation within the corpus.

Despite these problems, many algorithms have been developed and applied to the biomedical domain, specifically to literature available from MEDLINE. MEDLINE is a database of citations of scientific literature related to medicine and biology, originating in the 1960s. It is made available by the National Library of Medicine. MEDLINE currently contains more than 15 million citations and is growing at an increasing rate. Although MEDLINE does not contain the full text of the articles, it does contain the title and abstract of the articles, which constitutes the main parts of interest in text analysis.

MEDLINE is made available on the Internet at several different Web sites, with the most prominent site being PubMed at the National Center for

Biotechnology Information (NCBI). Because of the difficulties in obtaining full texts, the entire biomedical corpus for abbreviations has not yet been comprehensively analyzed. Indeed, some have argued that analyzing the full text may be required to find all abbreviations [9].

5.2 Identifying Abbreviations

In this chapter, we concentrate on abbreviation identification methods that are applied directly to the biomedical domain. Although many nonbiomedical abbreviation finders exist [4], the abbreviations in biomedicine deviate sufficiently from nonbiomedical text, so that general algorithms perform poorly and are difficult to adapt to the specialized domain [4, 10].

The problem of identifying abbreviations in free text can be decomposed into two broad steps: (1) finding a list of candidate <abbreviation, long form> pairs based on parentheses; and (2) distinguishing actual abbreviations from other parenthetical statements. Out of all candidate pairs extracted, only approximately 30% actually indicate an abbreviation [3].

The first step is to find candidates. The community generally agrees on the methods for finding them, although there are some differences in the details. Current methods search the text for parentheses that may indicate abbreviations. For further investigation, the methods extract candidates according to one of the two following patterns:

1. candidate long form (candidate short form)
2. candidate short form (candidate long form)

After the text is extracted from the parentheses, algorithms differentiate between these two patterns by assuming that abbreviations consist of a single word. If there is only a single word within the parentheses, then the text fits the first pattern; otherwise, it fits the second. One exception to this arises from the fact that some long forms also may consist of a single word [e.g., *the amount of Ab (antibody)*]. The algorithm thus may need to consider a third possibility that a single word inside the parentheses is a long form of the immediately preceding word [6]. In practice, only 1% of text matches the second pattern [3].

As another complication, the algorithm must determine the amount of text to examine in the first pattern. At one extreme, the algorithm may search for the long form in all the text from the beginning of the sentence. Although this strategy always would find the long form, it also would often find a coincidental combination of words that incorrectly appear to be a long form. Thus, most algorithms either will limit the amount of text that they examine, or will impose a penalty on long forms that include too many words. For example, some

algorithms limit the number of words examined, based on the number of letters in the abbreviation [1, 3, 5].

This general approach to finding candidates is limited, since it only considers abbreviations occurring in parentheses. However, abbreviations may be defined with other patterns, and methods that consider these patterns will be able to identify more abbreviations. Wren and Garner propose a method that examines sentences across multiple abstracts to find common sets of words that may be long forms [3]. Such approaches are still under development. Nevertheless, the extent of the benefits from this more-thorough treatment is unclear. It is possible that across a large corpus, such as MEDLINE or the Internet, common abbreviations will be defined in a way that is already detectable.

After the candidates are identified, the algorithm then must determine whether they are indeed abbreviations. In this step, there is considerable variation in approaches, which mainly differ based on the sources of the information. The three main approaches are Heuristic, Alignment, and NLP. Heuristic, or rule-based, methods attempt to match the letters in the abbreviation to the letters in the long form, using manually constructed patterns. Alignment methods find all possible matches between the letters in the abbreviation and the letters in the long form, and then score the quality of the matches. NLP methods may use heuristics or alignments, but also consider information about the parts-of-speech of the words. These three categories are not cleanly divided, and there is some overlap in their application.

5.2.1 Heuristics

The creation of abbreviations following a limited set of patterns is the main insight of a heuristic method. In the simplest case, concatenating the initial letters of the words in the long form creates an acronym. Similarly, authors also tend to use letters at the beginning of syllables, to ignore conjunctions and prepositions, and to use consecutive letters in a word. Abbreviations can be recognized if the candidate abbreviation matches the candidate long form according to one or more of these patterns.

Heuristic methods differ based on the types of information encoded in their rules. Although all methods favor letters that align to the beginning of words, methods also can give credit to letters that align on syllable boundaries [11], or even boundaries of domain-specific word fragments, such as *acetyl* [12].

For reasons described above, heuristic rules also should penalize words in the long form that do not match letters in the abbreviation. One exception to this rule is that authors often do not include words such as prepositions and conjunctions as part of the abbreviation. The ARGH method includes a list of such words, and does not penalize long forms in which these words are skipped in the abbreviation [3].

When matching letters from the abbreviation to the long form, one source of error is that the letters may match coincidentally. For example, in *argininosuccinate synthetase (AS)*, the letters in the abbreviation *AS* can match the underlined letters in *synthetase*, obviating the need for the first word, even though it clearly belongs in the long form [3]. A much better long form would match the *A* to *argininosuccinate*. Therefore, many algorithms include the rule that the first letter of the abbreviation must match the first letter in the definition. In fact, some have argued that using only this rule is sufficient to accurately identify abbreviations [10].

One can tune the performance of heuristic methods by including or modifying the patterns. Adding more promiscuous patterns would result in algorithms that find more abbreviations, at the cost of also including more incorrect ones. There is still disagreement on the optimal set of patterns, which will depend on the domain of text to be analyzed, and the performance required of the algorithm.

5.2.2 Alignment

In alignment methods, the notion of matching the letters from the abbreviation to those in the long form is made more explicit, since the alignment method shows the concordance between specific letters in the two forms. Having an explicit alignment allows the algorithm to handle gracefully two problems that occur in abbreviations: (1) the abbreviation contains letters that do not occur in the long form, and (2) the long form includes words that do not correspond to letters in the abbreviations.

To find all possible alignments between the candidate abbreviation and long form, these methods use a dynamic programming algorithm. This is a case of the Longest Common Substring (LCS) problem studied in computer science and adapted for biological sequence alignment in bioinformatics [13].

Dynamic programming can find the optimal alignment between two strings X and Y in $O(NM)$ time, where N and M are the lengths of the strings. This algorithm is expressed as a recurrence relation in (5.1).

$$\mathrm{M}[i,j] = \begin{cases} 0 & : \quad i = 0 \text{ or } j = 0 \\ \mathrm{M}[i-1, j-1] + 1 & : \quad i, j > 0 \text{ and } X_i = Y_j \\ \max(\mathrm{M}[i, j-1], \mathrm{M}[i-1, j]) & : \quad i, j > 0 \text{ and } X_i \neq Y_j \end{cases} \tag{5.1}$$

M is a score matrix. $M[i,j]$ contains the total number of characters aligned between the substrings $X_{1...j}$ and $Y_{1...j}$ in an optimal alignment. To recover the alignment, the algorithm keeps track of the aligned characters as it constructs the score matrix.

Once all possible alignments are gathered, the algorithm must choose the best alignment by examining each one and scoring it based on characteristics, such as whether the letters in the abbreviation match the initial letters of words in the long form, or whether there are many words in the long form not included in the abbreviation. The challenge is in finding optimal weights for each heuristic [14]. The weights can be specified manually [5] or determined with an automated machine learning algorithm [1].

Alignment methods share with heuristic methods many of the same advantages and disadvantages. The main difference is that the letter matches are delineated explicitly, which provides an obvious method to score the likelihood of the candidate, based on the characteristics of the alignment. However, generating the alignments can be computationally costly. Nevertheless, this method has been successfully applied to the entire MEDLINE database [1, 5].

5.2.3 Natural Language Processing

One of the limitations of the previous approaches is that they do not explicitly consider the parts-of-speech of the words, which can be useful to identify abbreviations. For example, in *Office of Nuclear Waste Isolation (ONWI)*, the *O* in the abbreviation can be matched with the initial letter of either *Office* or *of* [15]. The alignment that minimizes the length of the long form would align the *O* with *of* and result in an incorrect long form.

Information in the parts of speech may be useful in refining the extent of the long form. Pustejovsky et al. [4] noticed that the long forms of abbreviations often include complete noun phrases. Thus, in their algorithm, they apply a shallow parser to identify the noun phrases in the candidate long form, and use this information to constrain the search, so that the long form does not contain words extraneous to the actual definition of the abbreviation.

Using part-of-speech information also allows a more principled method to develop rules. While other heuristic systems rely on lists of noise words to exclude, NLP systems can instead exclude words based on part of speech. Such an approach is more complete than using manually constructed lists. It can also handle ambiguities in the noise words (e.g., if a word can serve both as a function word and a noun). Furthermore, using the part-of-speech allows systems to set a more natural boundary for the long form, based on the noun phrase closest to the target acronym.

5.2.4 Stanford Biomedical Abbreviation Method

In this section, we describe in detail the algorithm used to construct the Stanford Biomedical Abbreviation Database. We developed a method based on alignments, because it suggested a natural method to score the confidence of

abbreviations. In addition, allowing the users to choose score cutoffs will let them choose a suitable performance of the algorithm.

We decomposed the abbreviation-finding problem into four steps: (1) scanning text for occurrences of candidate abbreviations and long forms, (2) aligning the candidates to the preceding text, (3) describing the characteristics of the alignments as a feature vector, and (4) scoring the feature vector using a statistical machine learning algorithm.

We gathered candidates for abbreviations and their definitions. For every pair of parentheses, we retrieved the words up to a comma or semicolon. We rejected candidate abbreviations longer than two words, those without any letters, and those that exactly matched the words in the preceding text.

For each abbreviation candidate, we saved the words before the open parenthesis, so that we could search these words for the long form of the abbreviation. Although we could have included every word from the beginning of the sentence, as a computational optimization, we only used $3N$ words, where N was the number of letters in the abbreviation. We chose this limit conservatively, based on the informal observation that we always found long forms within $3N$ words.

We aligned the candidate abbreviation with its long form using the LCS algorithm, as described earlier.

Next, we calculated feature vectors that quantitatively described each candidate abbreviation and its alignment to its candidate long form. For the abbreviation recognition task, we used nine features described in Table 5.2. Each feature constituted one dimension of a nine-dimension feature vector, which describes the relevant characteristics of the alignment.

Finally, to score the features from the alignments, we used a supervised machine learning algorithm. To train this algorithm, we created a training set of 1,000 randomly chosen candidates identified from a set of MEDLINE abstracts pertaining to human genes, which we had compiled for another purpose. For the 93 real abbreviations, we annotated by hand the alignment between the abbreviation and long form.

Next, we generated all possible alignments between the abbreviations and long forms in our set of 1,000. This yielded our complete training set, which consisted of (1) alignments of incorrect abbreviations, (2) correct alignments of correct abbreviations, and (3) incorrect alignments of correct abbreviations. We converted these alignments into feature vectors.

Using these feature vectors, we trained a binary logistic regression classifier [17]. We chose this classifier based on its lack of assumptions on the data model, ability to handle continuous data, speed in classification, and probabilistically interpretable scores. To alleviate singularity problems, we removed all the duplicate vectors from the training set.

Table 5.2
Features, Descriptions, and Weights Used to Calculate the Score of an Alignment Using Logistic Regression

Feature	Description	Weight
Describes the abbreviation		
LowerAbbrev	Percent of letters in abbreviation in lowercase.	−1.21
Describes where the letters are aligned		
WordBegin	Percent of letters aligned at the beginning of a word.	5.54
WordEnd	Percent of letters aligned at the end of a word.	−1.40
SyllableBoundary	Percent of letters aligned on a syllable boundary.	2.08
HasNeighbor	Percent of letters aligned immediately after another letter.	1.50
Describes the alignment		
Aligned	Percent of letters in the abbreviation that are aligned.	3.67
UnusedWords	Number of words in the long form not aligned to the abbreviation.	−5.82
AlignsPerWord	Average number of aligned characters per word.	0.70
Miscellaneous		
CONSTANT	Normalization constant for logistic regression.	−9.70

Notes: Syllable boundaries identified using the algorithm in TEX [16]. The right-most column indicates the weight given to each feature. The sign of the weight indicates whether or not that feature is favorably associated with real abbreviations.

Finally, the score of an alignment is the probability calculated from the logistic regression. The score of an abbreviation is the maximum score of all the alignments.

We implemented the code in Python 2.2 [18] and C with the Biopython 1.00a4 and mxTextTools 2.0.3 libraries. We also created a Web site that implements this algorithm. The Web site was built with RedHat Linux 7.2, MySQL 3.23.46, and Zope 2.5.0, on a Dell workstation with a 1.5GHz Pentium IV and 512 MB of RAM. It is available at http://abbreviation.stanford.edu/ (see Figure 5.2).

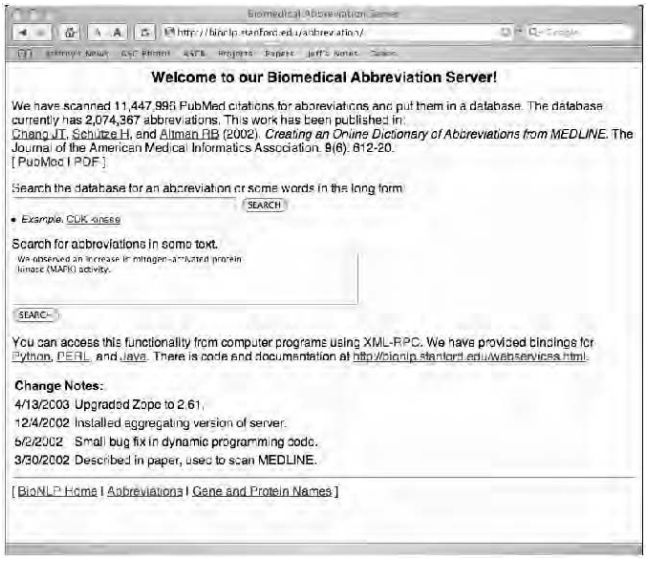


Figure 5.2 The Stanford Biomedical Abbreviation server allows users to (1) query for the definitions of abbreviations, (2) search for definitions that contain a keyword, and (3) extract abbreviations that occur in free text.

5.2.5 Evaluating Abbreviation Identification Methods

Evaluating the performance of abbreviation identification algorithms requires a suitable gold standard, in which the abbreviations and their definitions have been annotated manually by human readers. Issues in creating such a gold standard include its size, breadth, accuracy, and acceptance.

The size of a gold standard is important in obtaining an accurate measure of the performance of an abbreviation identification algorithm. A small set of documents may not include a wide range of the various types of abbreviations, which would not yield a reliable estimate of the performance of the algorithm. An algorithm may perform well on a small test set, but not generalize well to a larger corpus.

Similarly, the breadth of the domain covered is important. Many have noted that the patterns of abbreviations vary among domains, and that an algorithm developed for one domain may not perform as effectively in another. Notably, algorithms developed from the general literature have generally performed poorly on biomedical literature [4]. Similarly, an algorithm optimized to perform well on a subset of biomedical literature (e.g., gene names) may not necessarily perform as well on all biomedical literature. Therefore, the domain of

literature covered by a gold standard must be noted when evaluating the performance of an algorithm.

The third issue in abbreviation gold standards concerns their accuracy. Because gold standards are developed using human annotation, errors can occur. These errors are particularly problematic, because they penalize complete algorithms more than others do. That is, an algorithm that can detect 100% of the abbreviations may still receive a low score because many of the correct abbreviations would be counted as incorrect, since they are missing from the gold standard. Conversely, an algorithm that overlooks the same abbreviations that are missing in the gold standard may achieve 100% accuracy, despite incompletely finding the abbreviations. This may occur if the algorithm and the human expert shared biases against certain types of abbreviations, such as the hypernyms discussed at the beginning of this chapter. Ideally, a gold standard should be the product of multiple experts [6].

Finally, a gold standard must be generally accepted and used by the community in a standard manner. It should be publicly available and easily accessible. Because many methods are currently evaluated on disparate and internally developed gold standards, it is difficult to compare the performance of various approaches.

Unfortunately, the amount of human manual annotation required to create such a gold standard makes the endeavor difficult. One carefully constructed gold standard was used to evaluate the AbbRE algorithm [6]. The authors selected articles from both biology and medical journals, and asked three experts from each field to identify the abbreviations in abstracts. They resolved disagreements based on majority vote. However, the total number of abstracts annotated was small, consisting of only 10 articles.

Another popular gold standard is the Medstract acronym gold standard [4]. It consists of two annotated corpora. The development corpus is a set of 86 MEDLINE abstracts randomly collected from 1997 to 1998. A biologist manually annotated it, and found 155 pairs of abbreviations. Medstract broadly defines abbreviation to also include aliases (e.g., *apoptosis* for *programmed cell death*). The test corpus contains 100 abstracts, randomly selected from results of the search for *gene* in a diverse but small group of high-impact biomedical journals. This includes 173 alias pairs. The gold standard is publicly available as XML files at http://www.medstract.org/gold-standards.html.

Although Medstract is commonly used within the community, it contains some errors. Each algorithm that has used Medstract has corrected the errors in different ways. Although all of these algorithms have used the same gold standard, the differences in the corrections impede efforts to compare their performances [10].

Nevertheless, we evaluated our algorithm against Medstract, because of its general availability, after correcting six typographical errors in the XML file. We

applied our algorithm on the documents in Medstract and predicted a list of abbreviations, long forms, and their scores. We ranked the results in decreasing score, and calculated the recall and precision at every rank in the list to generate a curve showing the tradeoff between the two. Recall

$$\frac{\#\,\text{correct abbreviations}}{\text{all correct abbreviations}} \tag{5.2}$$

measures how thoroughly the method finds all the abbreviations. Precision

$$\frac{\#\,\text{correct abbreviations}}{\text{all predictions}} \tag{5.3}$$

indicates the number of errors produced. Near the top of the list, the recall is low, because few of the abbreviations are as yet identified, but precision is high, because the abbreviations that are identified are correct. However, as we include lower scoring abbreviations, the recall increases while the precision decreases.

We counted an <abbreviation, long form> pair as correct if it also occurred in the gold standard. To be consistent with the evaluation of the Acromed algorithm, we allowed mismatches in 10 cases, where the long form contained words not indicated in the abbreviation. For example, we accepted *protein kinase A* for *PKA*, and did not require the full *cAMP-dependent protein kinase A* indicated in the gold standard.

We ran our algorithm against the Medstract gold standard, and calculated the recall and precision at various score cutoffs (see Figure 5.3). Identifying 140 out of 168 correctly, it obtained a maximum recall of 83% at 80% precision. Our algorithm failed to find 25 total abbreviations in the Medstract gold standard. Table 5.3 categorizes the types of abbreviations and the number of each type missed. The recall/precision curve plateaued at two levels of precision, 97% at 22% recall (score = 0.88), and 95% at 75% recall (score = 0.14).

At a score cutoff of 0.14, the algorithm made eight errors, seven of which were abbreviations missing from the gold standard: *primary ethylene response element (PERE)*, *basic helix-loop-helix (bHLH)*, *intermediate neuroblasts defective (ind)*, *Ca2+-sensing receptor (CaSR)*, *GABA(B) receptor (GABA(B)R1)*, *polymerase II (Pol II)*, and *GABAB receptor (GABA(B)R2)*. The final error occurred when the algorithm assigned an incorrect long form because of a coincidental sequence of words (matching letters underlined): *Eas and Fas ligand (FasL)*. The seven missing abbreviations decreased the precision of the algorithm. Disregarding these cases yields a precision of 99% at 75% recall.

Furthermore, the majority of the errors on this dataset (see Table 5.3) occurred because the gold standard included synonyms (i.e., words and phrases

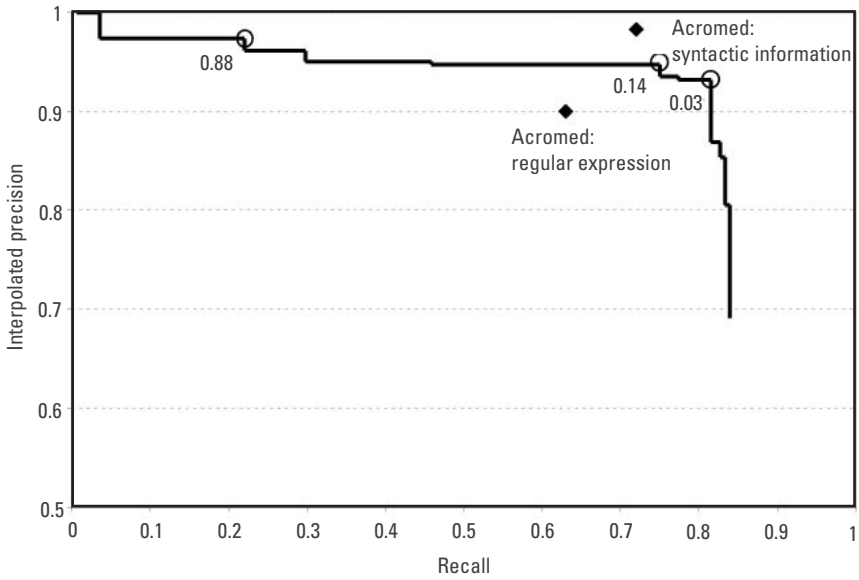


Figure 5.3 Algorithm applied to identify the abbreviations in the Medstract Gold Standard. The recall and precision at every score cutoff are calculated and plotted in the resulting curve. The scores at pivotal points are marked on the curve. The performance of the Acromed system is shown for comparison.

Table 5.3
Type of Abbreviations and the Number Missed

#	Description	Example	
12	Abbreviation and long form are synonyms.	apoptosis	programmed cell death
7	Abbreviation is outside parentheses.		
3	Best alignment score yields incorrect long form.	FasL	Fas and Fas ligand
3	Letters in abbreviation are out of order.	ATN	anterior thalamus
25	TOTAL		

with identical meanings) in addition to abbreviations. In these cases, the algorithm could not find the correspondences between letters, indicating a fundamental limitation of letter matching techniques.

5.3 Normalizing Abbreviations

After abbreviations are extracted from the text, abbreviations with identical meanings can still vary in minor ways. For example, *JNK* can be an abbreviation

for 155 different long forms, including *Jun N-kinase*, *Jun N-kinases*, and *Jun NH2 kinase*. Common abbreviations intuitively tend to have more textual variants than rare ones [19]. These variations complicate many problems, such as counting abbreviations accurately, mapping abbreviations to proper long forms, or viewing distinct abbreviations in a database. Therefore, there must be methods to normalize abbreviations and their definitions into a canonical form.

Abbreviations can vary in many ways. Orthographical variations include different capitalizations or spellings, such as *estrogen* versus *oestrogen*. An abbreviation for *retinoic acid receptor alpha* may be *RAR alpha*, *RAR-alpha*, *RARA*, or *RARa*. In addition, there are morphological variations due to different forms of words, such as *nuclear receptor* versus *nuclear receptors*, and *TR* or *TRs* for *thyroid hormone receptor(s)*. Furthermore, there are syntactic variations, such as *human cancer* versus *cancer in humans*. Finally, there are lexico-semantic variations, such as *carcinoma* versus *cancer* [12].

Orthographical and morphological variations involve only small changes in the letters. Therefore, algorithms can handle these variations using simple heuristics, such as (1) removing plural endings, (2) ignoring differences in white space or punctuation, and (3) converting long forms to lower case [3, 20]. Pustejovsky et al. also consider two long forms to be equivalent if 80% of the longer form also is present in the shorter form. One more sophisticated approach breaks each long form into a sequence of trigrams (three consecutive letters), and then clusters all the long forms based on the number of shared trigrams [5].

Although more difficult to handle, there have been some approaches to normalize lexico-semantic variations. Generally, these methods will map abbreviations to a list of semantic categories, and then group abbreviations that map to the same categories. The EXPGrouper program maps the long forms to the UMLS Specialist Lexicon [19]. The SaRAD program groups abbreviations from MEDLINE citations based on the MeSH terms[1] assigned to the citations, and clusters the abbreviations based on the similarity of those terms.

In our work, we used a heuristic approach to handle orthographical and morphological variations and cluster similar abbreviations. The algorithm is based on the notion that the long forms with small variations can be safely aggregated, if their abbreviations are the same. For each abbreviation found in MEDLINE, we first sort the long forms in alphabetical order. We then consider each <abbreviation, long form> pair sequentially, and aggregate the pair with a previous pair if they meet two conditions:

1. Medical Subject Headings (MeSH) is a hierarchical vocabulary that describes the contents of biomedical documents. Human annotators at the NCBI assign MeSH terms based on reading the full text of the article.

1. The abbreviations are the same, or differ by no more than an *s* (or *es*) appended to the end.
2. The alphanumeric characters (ignoring spaces and punctuation) in the long forms differ by at most one character.

Following these criteria, we group together all pairs of <abbreviation, long form> that are similar. We apply these rules transitively. That is, if abbreviation A is similar to abbreviation B, which in turn is similar to abbreviation C, then we create a cluster with all three abbreviations. As a result, the final cluster may contain abbreviations whose long forms differ by more than one character.

We also experimented with alternate, more lenient, conditions that allow more mismatches, depending on the length of the long form. When we allowed one mismatch per *N* characters, more stringent mismatch requirements (greater *N*) intuitively led to increased numbers of clusters, as shown in Table 5.4. The abbreviations are clustered, allowing *N* mismatches per alphanumeric character in the long form. The first row, X mismatches, is the clustering obtained when only one mismatch is allowed, regardless of the length of the long form. The Edit Distance is the maximum edit distance between two long forms in the same cluster. Fewer <abbreviation, long form> pairs could be clustered together.

Although the number of clusters varies, the method appears robust and clusters together similar long forms. For example, when using the most lenient strategy, allowing a mismatch every other character, the two most distant long forms were:

```
1 colony-forming units, erythroid burst-forming units, and
  |||||||||||||| ||||
2 colony-forming unit-
```

Table 5.4
Clusters of Abbreviations

N	# Clusters	Edit Distance
X	609,162	14
2	461,822	83
3	510,955	81
5	555,813	47
7	576,524	33
10	592,912	23
15	609,105	21

```
1 granulocyte erythrocyte macrophage             megakaryocyte
  ||||||||||| ||||||||||  |||||||||||||
2 granulocyte,            macrophage, erythroid, megakaryocyte
1 colony-forming units
2
```

Requiring the abbreviations to be the same (except for a possible *s* at the end) constrains the long forms that may be clustered together. However, such a simple heuristic can cluster long forms with similar letters but different meaning. One example is:

```
1 androgen      receptor
  | ||  ||      ||||||||
2 a dr  energic receptor
```

However, the frequency or significance of such errors for the user is unclear.

Ultimately, we used the computationally cheapest strategy, and allowed one mismatch between any two long forms. When applied to abbreviations identified from all of MEDLINE, this heuristic reduced 1,948,246 abbreviations into 609,162 clusters.

5.4 Defining Abbreviations in Text

Although many approaches can identify abbreviations and their long forms in text, it is also necessary to define abbreviations whose long forms do not explicitly appear in the same document. In one study in MEDLINE abstracts, 22% of the total number of abbreviations used were not defined [3]. Comprehensive text mining algorithms need to recognize abbreviations and correctly define them.

The main difficulty in defining abbreviations stems from the fact that an abbreviation can have multiple long forms with different meanings. For example, *AR* can have many forms, including *autosomal recessive*, *androgen receptor*, *amphiregulin*, *aortic regurgitation*, *aldose reductase*, and so forth. A total of 36% of abbreviations in MEDLINE have more than one definition, and 10% of definitions have more than one abbreviation. In fact, the number of new definitions is growing at four times the rate as that of new abbreviations [3]. It is clear that many definitions are being assigned to previously existing abbreviations. Therefore, algorithms often must disambiguate among multiple definitions.

One method to choose among alternate possible definitions relies on the prevalence of the acronyms [3]. If one definition of an abbreviation was usually dominant, then the algorithm usually would be correct if it always chose the most prevalent one. Clearly, there are limitations to this method, and it has not been recommended for general use. However, knowing the prevalence of the definitions would be useful in helping humans to decide on the proper meaning.

The other main approach to choosing definitions relies on the context of the abbreviation. The correct definition of the abbreviation should be concordant with the content of the remainder of the document.

One indicator of the domain of a MEDLINE citation is its MeSH terms. Yu et al. [6] first proposed the idea of using these terms to disambiguate abbreviations. Adar investigated this further by successfully grouping abbreviations based on the MeSH terms of their documents [5].

An obvious limitation of MeSH-based approaches is that they require manually annotated MeSH terms, which may apply only to MEDLINE citations. However, people also have reported success disambiguating abbreviations based on words. These approaches represent the words surrounding the abbreviation as a vector, where each dimension is either 1 or 0, depending on whether a specific word is present. For every possible long form, vectors describing their contexts are collected. The vector for the ambiguous abbreviation is then matched to see which of the vectors of the long forms it most resembles. The algorithms used to score this match include a cosine score [4], naïve Bayes [21], maximum entropy [22], and support vector machines [23]. These algorithms are very similar to those used for word sense disambiguation in computational linguistics [24].

5.5 Abbreviation Databases

The development of methods to handle abbreviations in text has led to the automatic construction of comprehensive databases of biomedical abbreviations. The four most prominent databases are AcroMed, ARGH, the Stanford Biomedical Abbreviation Database, and SaRAD, which were built using algorithms described previously [1, 3–5]. All are currently available on-line. There has not yet been a comprehensive study comparing their coverage or accuracy.

Earlier in this chapter, we speculated that despite their limits, current abbreviation identification methods might be sufficiently accurate to find all meaningful biomedical abbreviations. Therefore, we evaluated the comprehensiveness of the Stanford Biomedical Abbreviation Database, which was created from all abbreviations found in MEDLINE.

In order to quantify the coverage of the database, we searched it for abbreviations found in a list from the China Medical Tribune, a weekly Chinese language newspaper covering medical news from Chinese journals [25]. Its Web site includes a dictionary of 452 commonly used English medical abbreviations and their long forms. After normalizing for capitalization and punctuation, the algorithm correctly identified 88% of the abbreviations in the list. Out of the 53 abbreviations missed, 11 of them appeared in the database as a close variation, such as *Elective Repeat Caesarean-Section* instead of *Elective Repeat C-Section*.

Since the abbreviation list from the China Medical Tribune was created independently of MEDLINE, the results suggest that the database contains a great number of all biomedical abbreviations, and that automatically created databases are a rich and comprehensive source of biomedical abbreviations.

5.6 Conclusion

Due to the enormous number of abbreviations currently in MEDLINE, and due to the rate at which prolific authors define new ones, maintaining a current dictionary of abbreviation definitions clearly requires automated methods. Since nearly one-half of MEDLINE abstracts contain abbreviations, computer programs analyzing this text will frequently encounter abbreviations and can benefit from their identification. Since fewer than one-half of all abbreviations are formed from simple patterns (e.g., concatenating initial letters of words), automated methods must handle sophisticated and nonstandard constructions.

Nevertheless, algorithms and databases are presently useful resources for readers facing unfamiliar abbreviations in the literature. However, identifying abbreviations ultimately remains an active area of research. The main unsolved problems include finding proper boundaries of long forms using knowledge-based approaches, automatically defining abbreviations in articles, and incorporating abbreviation algorithms and databases into downstream programs. A novel area of investigation is in algorithms that can automatically generate abbreviations from long forms, so that search algorithms might expand their queries for more sensitive results [26]. Sophisticated handling of abbreviations is an area that likely will have a profound impact on the development of intelligent systems for biological text processing.

References

[1] Chang, J. T., H. Schütze, and R. B. Altman, "Creating an Online Dictionary of Abbreviations from Medline," *Journal of the American Medical Informatics Association*, Vol. 9, No. 6, 2004, pp. 612–620.

[2] Jablonski S., (ed.), *Dictionary of Medical Acronyms & Abbreviations*, Philadelphia, PA: Hanley & Belfus, 1998.

[3] Wren, J. D., and H. R. Garner, "Heuristics for Identification of Acronym-Definition Patterns Within Text: Towards an Automated Construction of Comprehensive Acronym-Definition Dictionaries," *Methods for Information in Medicine*, Vol. 41, No. 5, 2002, pp. 426–434.

[4] Pustejovsky, J., et al., "Automatic Extraction of Acronym-Meaning Pairs from Medline Databases," *Medinfo*, Vol. 10, 2001, pp. 371–375.

[5] Adar, E., "Sarad: A Simple and Robust Abbreviation Dictionary," *Bioinformatics*, Vol. 20, No. 4, 2004, pp. 527–533.

[6] Yu, H., G. Hripcsak, and C. Friedman, "Mapping Abbreviations to Full Forms in Biomedical Articles," *Journal of the American Medical Informatics Association*, Vol. 9, 2002, pp. 262–272.

[7] Federiuk, C. S., "The Effect of Abbreviations on Medline Searching," *Academic Emergency Medicine*, Vol. 6, No. 4, 1999, pp. 292–296.

[8] Friedman, C., et al., "Evaluating the UMLS as a Source of Lexical Knowledge for Medical Language Processing," *Proc. American Medical Informatics Association Annual Symposium*, 2001, pp. 189–193.

[9] Schuemie, M. J., et al., "Distribution of Information in Biomedical Abstracts and Full-Text Publications," *Bioinformatics*, Vol. 20, No. 16, 2004, pp. 2597–2604.

[10] Schwartz, A. S., and M. A. Hearst, "A Simple Algorithm for Identifying Abbreviation Definitions in Biomedical Text," *Proc. Pacific Symp. on Biocomputing*, 2003, pp. 451–462.

[11] Yoshida, M., K. Fukuda, and T. Takagi, "Pnad-css: A Workbench for Constructing a Protein Name Abbreviation Dictionary," *Bioinformatics*, Vol. 16, No. 2, 2000, pp. 169–175.

[12] Nenadic, G., I. Spasic, and S. Ananiadou, "Automatic Acronym Acquisition and Management with Domain Specific Texts," *Proc. 3rd Int. Conf. on Language Resources and Evaluation*, 2002, pp. 2155–2162.

[13] Needleman, S. B., and C. D. Wunsch, "A General Method Applicable to the Search for Similarities in the Amino Acid Sequence of Two Proteins," *Journal of Molecular Biology*, Vol. 48, No. 3, 1970, pp. 443–453.

[14] Yeates, S., "Automatic Extraction of Acronyms from Text," *New Zealand Computer Science Research Students' Conf.*, 1999, pp. 117–124.

[15] Taghva, K., and J. Gilbreth, *Recognizing Acronyms and Their Definitions*, Technical report, Information Science Research Institute (ISRI), University of Nevada, Las Vegas, 1995.

[16] Knuth, D., *The Texbook,* Reading, MA: Addison-Wesley, 1986.

[17] Hastie, T., R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, New York: Springer-Verlag, 2001.

[18] Lutz, M., D. Ascher, and F. Willison, *Learning Python*, Sebastopol, CA: O'Reilly, 1999.

[19] Liu, H., Y. A. Lussier, and C. Friedman, "A Study of Abbreviations in the UMLS," *Proc. American Medical Informatics Association Annual Symp.*, 2001, pp. 393–397.

[20] Pustejovsky, J., et al., "Medstract: Creating Large-Scale Information Servers for Biomedical Libraries," *ACL Workshop on Natural Language Processing in the Biomedical Domain*, 2002, pp. 85–92.

[21] Liu, H., S. B. Johnson, and C. Friedman, "Automatic Resolution of Ambiguous Terms Based on Machine Learning and Conceptual Relations in the UMLS," *Journal of the American Medical Informatics Association*, Vol. 9, No. 4, 2002, pp. 621–636.

[22] Pakhomov, S., "Supervised Maximum Entropy-Based Approach to Acronym and Abbreviation Normalization in Medical Texts," *Proc. 40th Annual Meeting of the Association for Computational Linguistics*, 2002, pp. 160–167.

[23] Yu, Z., Y. Tsuruoka, and J. Tsujii, "Automatic Resolution of Ambiguous Abbreviations in Biomedical Texts Using Support Vector Machines and One Sense Per Discourse Hypothesis," *SIGIR 2003 Workshop on Text Analysis and Search for Bioinformatics*, 2003, pp. 57–62.

[24] Manning, C. D., and H. Schütze, *Foundations of Statistical Natural Language Processing*, Boston, MA: MIT Press, 1999.

[25] *China Medical Tribune*, http://www.cmt.com.cn/, 2002.

[26] Tsuruoka, Y., S. Ananiadou, and J. Tsujii, "A Machine Learning Approach to Automatic Acronym Generation," *Proc. ISMB BioLINK SIG: Linking Literature, Information, and Knowledge for Biology* and *13th Ann. Intl. Conf. on Intelligent Systems for Molecular Biology (ISMB 2005)*, Detroit, MI, 2005, pp. 25–31.

6

Named Entity Recognition

Jong C. Park and Jung-jae Kim

6.1 Introduction

To mine useful knowledge from the biomedical literature, we should be able to recognize names of biomedical entities, such as genes, proteins, cells, and diseases [1, 2]. Named entity recognition (NER) refers to the task of recognizing such entity-denoting expressions, or named entities (NE), in natural language documents. In this chapter, we will be mostly concerned with biomedical NER, or NER in biomedicine.

The primary goal of NER is to relate each NE of importance in a natural language document to an individual in the real world. In this regard, NER is different from automatic term recognition (ATR), as introduced in Chapter 4, whose main goal is to associate a given term with a concept in a well-defined semantic framework. In a practical text mining system, the recognized NE then would be further classified with the known types of real-world entities. For example, if the NE denotes a gene, then it would be classified into the class of gene names. The classified NE is then linked to, or grounded by, a concept in the semantic hierarchy. This latter process for biomedical applications takes into account only the biological functions of the biomedical entity denoted by the NE.

Biomedical NER is certainly a nontrivial task, despite the availability of many well-known nomenclatures for biomedical entities, including the one published by the HGNC (http://www.gene.ucl.ac.uk/nomenclature/) for human genes [3]. These resources do not address certain issues in NER, such as

ambiguities and aliases of gene names [4]. We illustrate such issues with examples. First, the protein name *ARF* may denote either a small GTP-binding protein that is involved in vesicular transport, or a tumor suppressor gene product that binds to p53-DNA complexes [4]. Second, the protein name *p53* does not describe the function of the protein, only the weight of the protein. There might be quite a few other proteins with a similar molecular weight of approximately 53,000 [4]. Finally, one gene, officially designated as *SELL* or *selectin L*, which controls cell adhesion during immune responses, is currently known to have as many as 15 aliases [5].

Biomedical NER also requires a considerable linguistic analysis of NEs. In particular, NER, as utilized for IE systems, must deal with a much broader range of linguistic expressions than that for text retrieval (TR) systems. IE systems identify NEs in text to fill in slots of templates for information of interest, as described in Chapter 7, while TR systems create an index with unique NEs identified in texts, and retrieve relevant texts when the user asks for the indexed entities with a query term. For example, a text mining system for protein-protein interactions should identify not only protein names from proteome databases, such as UniProt, but also anaphoric expressions, such as *it* and *the protein*, which refer to interacting proteins, as illustrated in Table 6.1.[1] Such anaphoric expressions subsequently may be replaced with their antecedent protein names as defined in the preceding context, when they are put into relevant slots of protein-protein interaction templates. The text mining system also should identify species information for proteins, especially when such information plays a critical role, as in the following examples (species names underlined): *Plant DNA polymerases and E. coli DNA polymerase I, but not animal DNA polymerases or avian reverse transcriptase, are strongly stimulated by ethidium bromide (EtdBr) … (PMID:6821157).*

Notice that NER for IE systems is, in general, more complicated than that for TR systems. A TR system employs an NER module to recognize a list of

Table 6.1
Example Sentences with NEs From a MEDLINE Abstract

Arfophilin is an ADP ribosylation factor (Arf) binding protein of unknown function. *It* is identical to the Rab11 binding protein eferin/Rab11-FIP3, and we show *it* binds both Arf5 and Rab11. (PMID:12857874)

1. The number following the string "PMID:" is the identifier of a MEDLINE abstract in PubMed (http://www.ncbi.nlm.nih.gov/PubMed/).

unique NEs in a text, as the index for the text. An IE system may extract information of interest by employing another NER module that recognizes such linguistic expressions as anaphoric expressions, which contribute to syntactic structures that encode the information of interest, along with their surrounding expressions. For instance, such an NER module must be able to extract from the passage in Table 6.1 the fact that Arfophilin binds to Arf5 and Rab11, by analyzing the binding event of the pronoun *it*, and resolving the pronoun with the protein name *Arfophilin*.

NER has been well-structured over the years, through several series of evaluation conferences on IE. MUCs are well-known for the extraction of terrorist events from newspapers in their earlier contests [6]. Note that all the MUCs, except MUC-7 [7], have included the task of recognizing NEs, such as the names of persons, organizations, places, and artifacts, as the basis for further levels of information extraction. MUC-7 chose not to include the task of NER in its competition sessions, since the highest scoring system for NER at MUC-6 [6] showed a performance that is comparable to those of human curators. Instead, MUC-6 and MUC-7 included a novel task of recognizing template elements, to identify descriptions of entities (e.g., *a distributor of kumquats*), as well as names, based on the observation that an entity may be mentioned several times, possibly using descriptions and different forms of its name. In this chapter, we focus on NER, rather than the task of template element recognition, since the latter has not yet been seriously considered in the biomedical domain.

Evaluation of system performance for NER, in the aforementioned conferences, is usually accomplished by an automatic scoring program [8], where the scores are based on two measures: recall and precision [9]. Recall is the percentage of the NEs that the system has identified correctly divided by all correct NEs, and precision is the percentage of the correctly identified NEs divided by the NEs identified by the system. The component scores of recall and precision then are used to calculate a balanced F-score [10], where $F = 2 \times P \times R/(P + R)$.[2] For instance, the highest scoring system for NER at MUC-6 showed a recall of 96%, and a precision of 97%, thus showing an F-score of 96.5.

The structure of the rest of the chapter is as follows. We describe the nature and examples of the candidate NEs in the biomedical literature in Section 6.2; discuss the issues in recognizing gene and protein names in Section 6.3; compare previous approaches to gene and protein name recognition in Section 6.4; and present evaluation conferences for biomedical NER, and compare approaches to NER in the general domains with the approaches to gene and protein name recognition, in Section 6.5.

2. P, R, and F are abbreviations for precision, recall, and F-score, respectively.

6.2 Biomedical Named Entities

An NE is a phrase or a combination of phrases in a document that denotes a specific object or a group of objects, such as persons, organizations, places, and artifacts in newspapers [6, 7], courts and parties in court opinions [11], and genes, proteins, cells, drugs, chemicals, and diseases in the biomedical literature [12, 13]. Table 6.2 shows sample NEs for each category in the biomedical literature [3, 12, 14].[3] The databases in Table 6.3 include a comprehensive list of biomedical NEs (see Chapter 3 for further details). For instance, HUGO Nomenclature provides records of more than 21,000 human genes, and Release 47.0 (May 10, 2005) of Swiss-Prot, the curated protein sequence database of UniProt, contains approximately 180,000 entries.[4] Among the biomedical NEs in Table 6.2, we will hereafter focus on genes and proteins[5] because of their importance in

Table 6.2
Example NEs of Biomedical Objects

1.	Genes: Tp53, agaR
2.	Proteins: p53, 'galactosidase, alpha (GLA)'
3.	Cells: CD4+-cells, Human malignant mesothelioma (HMMME)
4.	Drugs: Cyclosporine, herbimycin
5.	Chemicals: 5'-(N-ethylcarboxamido)adenosine (NECA)

Table 6.3
Biomedical Databases and Resources

1.	Genes: Human Genome Nomenclature (http://www.gene.ucl.ac.uk/nomenclature/), GenBank (http://www.ncbi.nlm.nih.gov/Genbank/)
2.	Proteins: UniProt (http://www.expasy.org/sprot/), IPI (http://www.ensembl.org/IPI/)
3.	Cells: Cell database of Riken Bioresource Center (http://www.brc.riken.jp/inf/en/)
4.	Drugs: MedMaster (http://www.ashp.org/), USP DI (http://www.usp.org/)
5.	Chemicals: UMLS Metathesaurus (http://www.nlm.nih.gov/research/umls/)
6.	Diseases: NCBI Genes and Diseases (http://www.ncbi.nlm.nih.gov/disease/), Disease Database (http://www.diseasesdatabase.com/)

3. Several example names in Table 6.2 are from [3, 12, 14], and the cell database of Riken Bioresource Center (http://www.brc.riken.jp/lab/cell/english/).
4. In the rest of this chapter, we do not differentiate gene names from protein names if there is no confusion, since a protein or a gene product is produced from a corresponding gene.
5. See the Web sites of databases for details.

biology, and because of the emphasis that previous studies of NER in biomedicine have put on them.

The gene and protein names in the databases and in the literature show several characteristics in common. For instance, many gene and protein names include special characters of the type shown in Table 6.4 (e.g., *D(1)* or *Thioredoxin h-type 1*). The gene and protein names also may consist of descriptive terms that suggest the characteristics of proteins, including function, localization, species, physical properties, and similarities to other proteins, as exemplified in Table 6.5 [15]. However, in the case of protein names, such as *p21* and *rad51*, it is impossible to uncover the functions of corresponding proteins by an inference over such names alone [4, 16].

Gene and protein names can be expressed in various linguistic forms, including abbreviations, plurals, compounds, anaphoric expressions, and descriptions, as exemplified in Table 6.6 [17].[6] Notice that descriptions show a different range of linguistic realizations, such as prepositional phrases, relative phrases, and even expressions across sentences. This diversity of descriptions is similar to that of terms, as discussed in [18, 19] and in Chapter 4. Biomedical text mining systems must take into account this linguistic variety of gene and protein names with respect to their goals.

Table 6.4
Special Characters in Biomedical NEs

upper case, comma, hyphen, slash, bracket, digit

Table 6.5
Example Protein Names with Descriptive Terms

Semantic Type of Descriptive Term	Example Protein Name
Protein function	growth hormone
Localization	nuclear protein
Species origin	HIV-1 envelope glycoprotein
Physical property	salivary acidic protein
Similarity to other proteins	Rho-like protein

6. Several examples in Table 6.6 are from [17].

Table 6.6
Example Gene and Protein Names in Various Linguistic Forms

Linguistic Forms	Example Gene and Protein Names
Abbreviation	GLA (as in Table 6.2)
Plural	p38 MAPKs, ERK1/2
Compound	Rpg1p/Tif32p
Coordination	91 and 84 kDa proteins
Cascade	kappa 3 binding factor (such that *kappa 3* is a gene name)
Anaphoric expression	it
	this enzyme
Description	an inhibitor of p53
	a protein that does not bind RNA directly but inhibits the activity of eIF4E
Acronym	phospholipase D (PLD)
	c-Jun N-terminal kinase (JNK)
Apposition	PD98059, specific MEK1/2 inhibitor
	U0126 (known as the ERKs inhibitor)

6.3 Issues in Gene/Protein Name Recognition

As mentioned in the introductory section, the recognition of gene and protein names in the biomedical literature is not straightforward, despite many well-known nomenclatures, such as HUGO and Swiss-Prot [20–22]. In this section, we examine several open issues, as related to ambiguous names, synonyms, variations, newly published names, and varying sets of target specific gene and protein names.

6.3.1 Ambiguous Names

Some ambiguous names denote different genes and proteins, as explained with respect to the name *ARF* in Section 6.1. There also are other gene and protein names that may be confused with common English words, such as *can*, *for*, *not*, *vamp*, *zip*, *white*, and *cycle* [20–24]. Hirschman et al. [20] reported that simple pattern matching for gene names shows extremely low precision, 2% for full texts and 7% for abstracts, and that the largest source of errors is the gene names that share their form with common English words. This problem of ambiguous names becomes more acute when we consider the change of entities referred to by the same name over time. For example, the name *p21* formerly denoted a macromolecule associated with a cascade of signals from receptors at cell surfaces

to the nucleus, which stimulates cell division, but currently it denotes a different protein that inhibits the cell cycle [16]. Some gene names remain ambiguous, if we do not take their species information into account. For example, the yeast homologue of the human gene PMS1 is called *PMS2*, whereas yeast PMS1 corresponds to human *PMS2*. There also are other ambiguous names that may denote biomedical entities of different classes. For example, *myc-c* can be a gene name, as well as a protein name, as in *myc-c gene* and *myc-c protein*. Likewise, *CD4* can be a protein name, as well as a cell name, as in *CD4 protein* and *CD4+-cells* [25]. To deal with such ambiguous names, we need to build up a list of such names and rules, or use statistical models that resolve the ambiguous names.

6.3.2 Synonyms

While an ambiguous name may denote different entities, an entity also can be denoted by multiple names in a synonymy relation, also called aliases. Table 6.7 shows example synonyms [16, 26].[7] In addition, some gene and protein names denote the same protein that is identical to its homologous protein in different species. For example, Drosophila and mouse genetics agree that *armadillo* from fruit flies and *β-catenin* from mice are basically the same. Nevertheless, they continue to use two different names [5]. In order to deal with this issue, we need to construct a list of synonymous names from biomedical resources. For an instance of gene and protein sequence databases, HUGO Nomenclature includes more than 23,000 aliases among more than 21,000 human genes, and Release 47.0 of Swiss-Prot contains more than 26,000 synonyms of protein names among approximately 180,000 entries. Since these resources contain only the names that are representative and commonly used, we further need to automatically recognize relations between known names and their unregistered synonyms.

6.3.3 Variations

In addition to synonyms, variations of gene and protein names also denote the same entities by definition. Interestingly, gene and protein names show a high degree of variations in the literature, including character-level variations,

Table 6.7
Example Synonyms of Gene and Protein Names

1.	caspase-3 *or* CASP3 *or* apoptosis-related cysteine protease *or* CPP32
2.	p21 *or* WAF1 *or* CIP1 *or* SDI1 *or* CAP20

7. Examples in Table 6.7 are from [16, 26].

word-level variations, word-order variations, syntactic variations, and variations with abbreviations, as illustrated in Table 6.8 [20, 21, 27].[8] We explain each type of variation as follows. The difference between character-level variations is the presence (or absence) of special characters, or the exchange of indices, such as digits and single alphabets. A word in a name can be replaced with another, or omitted, in the variant of the name. The strings of words in word-order variations show a different word order. A name with a flat structure can be changed into a variant with a prepositional phrase, or vice versa. A subsequence of a full name can be replaced with its abbreviation. Notice that all such synonyms and variations should be associated with their corresponding standard names [28]. To point out the seriousness of this issue, Tuason et al. [21] reported that character-level and word-level variations cause up to 79% of failures in gene name recognition.

6.3.4 Names of Newly Discovered Genes and Proteins

Another source of problems for effective recognition of gene and protein names is the overwhelming growth rate and the constant discovery of novel genes and proteins.[9] It takes time to register new names of the genes and proteins in

Table 6.8
Example Variations of Gene and Protein Names

(1) Character-level variations	(a) D(2) *or* D2 (b) SYT4 *or* SYT IV (c) IGA *or* IG alpha (d) S-receptor kinase *or* S receptor kinase (e) Thioredoxin h-type 1 *or* Thioredoxin h (THL1)
(2) Word-level variations	(a) RNase P protein *or* RNase P (b) Interleukin-1 beta precursor *or* INTERLEUKIN 1-beta PROTEIN *or* *INTERLEUKIN 1 beta* (c) transcription intermediary factor-2 *or* transcriptional intermediate factor 2 (d) the Ras guanine nucleotide exchange factor Sos *or* the Ras guanine nucleotide releasing protein Sos (e) hepatic microsomes *or* liver microsomes
(3) Word-order variations	(a) Collagen type XIII alpha 1 *or* Alpha 1 type XIII collagen (b) integrin alpha 4 *or* alpha4 integrin

8. Several examples in Table 6.8 are from [20, 21, 27].
9. Refer to the statistics on the Web sites of UniProt and GenBank for details.

curated nomenclatures, so the new names may resist an identification method that employs only dictionary look-up. Thus, we need to develop rules and models that recognize novel gene and protein names with their common characteristics. Note that the guidelines for gene and protein nomenclatures, such as those in [3], cannot distinguish gene and protein names from other terms and common words. Furthermore, the existing guidelines for gene and protein nomenclatures are not discriminative enough to distinguish the gene and protein names that follow the guidelines from other terms and common words [3].

6.3.5 Varying Range of Target Names

Though there are various linguistic forms of gene and protein names, as illustrated in Table 6.6, it is not true that all biomedical text mining systems have to recognize all the linguistic forms. For instance, a biomedical text mining system that aims at constructing an index of gene and protein names from biomedical documents does not need to recognize indefinite phrases, such as *an 89 kD protein* and *this protein* [15, 29], although these phrases may denote real genes and proteins. However, a biomedical text mining system that aims at extracting protein-protein interactions may need to explicitly deal with these phrases as valid arguments of interaction information, after first resolving them [30]. Likewise, the names for protein families, such as *protein kinases*, would not be necessary for a system that extracts only names of individual proteins [15], but would be useful for another system that extracts general biomedical interactions. It also depends on the purposes of the particular biomedical text mining system; for example, whether the system extracts adjectives that modify noun phrases as genes and proteins (e.g., *eukaryotic* in *eukaryotic RhoA-binding kinases*), or whether the system extracts a substring of a noun phrase as a gene and protein name (e.g., *RhoA* in *eukaryotic RhoA-binding kinases*) [15].

6.4 Approaches to Gene and Protein Name Recognition

We now explain the current progress in gene and protein name recognition with some selected papers that clearly show methodological improvements. We classify such papers roughly into the following four groups.

1. Dictionary-based approaches that try to find names of the well-known nomenclatures in the literature;
2. Rule-based approaches that manually or automatically construct rules and patterns to directly match them to candidate NEs in the literature;

3. Machine learning approaches that employ machine learning techniques, such as HMMs and SVMs, to develop statistical models for gene and protein name recognition;
4. Hybrid approaches that merge two or more of the above approaches, mostly in a sequential way, to deal with different aspects of NER.

6.4.1 Dictionary-Based Approaches

Unlike the names of persons and locations in the general domain, gene and protein names have been well-managed through databases by leading organizations, such as the National Center for Biotechnology Information (NCBI) (http://www.ncbi.nlm.nih.gov/) and the European Bioinformatics Institute (EBI) (http://www.ebi.ac.uk/). It is a natural consequence that previous approaches to gene and protein name recognition have been heavily dependent on such databases. The approaches usually try to find database entry names directly from the literature [31, 32]. However, they have several limitations, including false positive recognition caused by ambiguous names, false negative recognition due to synonyms, variations, and lack of a unified resource that covers newly published names [33].

Krauthammer et al. [34] deal with morphological variations of gene and protein names by utilizing BLAST [35], a popular tool for DNA and protein sequence comparison. The method first translates gene and protein names into an alphabet of DNA sequences by substituting each character in the names into a predetermined unique nucleotide combination. It also translates biomedical articles into a string of nucleotides in the same way. It then tries to match the translated articles to the translated names, and finally generates matched names if there is any significant alignment (P = 71.7%, R = 78.8%, F = 74.6).[10] This method can recognize unknown names that are similar to registered names, but some of their characters are different from those of the registered names, as in *interleukin-2* and *interleukin-3.*

Hanisch et al. [26] attempt to construct a comprehensive dictionary of genes and proteins by merging HUGO Nomenclature, OMIM database (http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=OMIM), and UniProt. UniProt consists of Swiss-Prot and TrEMBL, the computer-annotated supplement of Swiss-Prot that contains all the translations of EMBL (http://www.ebi.ac.uk/embl/) nucleotide sequence entries not yet integrated in Swiss-Prot. They curate the unified dictionary semiautomatically, by expanding it with morphological variations of its entries, and by pruning redundant, ambiguous, and irrelevant synonyms. They present a method of detecting gene and protein

10. Note that the published performances of work introduced in this paper cannot be directly compared against one another, due to the lack of a common evaluation metric.

names with the unified dictionary. The method processes tokens in a MEDLINE abstract one at a time, and scores each candidate name with two measures; that is, boundary score to control the end of the candidate, and acceptance score to determine whether the candidate is reported as a match (P = 95.0%, R = 90.0%, F = 92.4). The authors utilize robust linear programming (RLP) [36] to compute parameters in the two scoring measures.

Tsuruoka and Tsujii [33] address the aforementioned problems of dictionary-based approaches with a two-phase method. The method first scans texts for protein name candidates, using a protein name dictionary expanded by a probabilistic variant generator. This generator produces morphological variations of names in the class 'Amino Acid, Peptide, or Protein' of the UMLS Metathesaurus, and further gives each variant a generation probability that represents the plausibility of the variant. In the second phase, the method filters out irrelevant candidates of short names, by utilizing a Naïve Bayes classifier [37], with the features of words both within the candidates and surrounding the candidates. The authors evaluate the method on the GENIA corpus of MEDLINE abstracts that are annotated with NEs using a hierarchy of semantic classes [38] (P = 71.7%, R = 62.3%, F = 66.6).

6.4.2 Rule-Based Approaches

The dictionary-based approaches can deal with only morphological variations that correspond to some of the character-level and word-level variations in Table 6.8. Rule-based approaches can deal with a broader range of variations, even covering a few of the word-order variations and syntactic variations in Table 6.8.

Fukuda et al. [27] present a method of protein name recognition that utilizes surface clues on character strings (P = 94.7%, R = 98.8%, F = 96.7). The method first identifies core terms, those that contain special characters in Table 6.4, and feature terms, those that describe biomedical functions of compound words (e.g., *protein* and *receptor*) (see Table 6.5). It then concatenates the terms by utilizing handcrafted rules and patterns, and extends the boundaries to adjacent nouns and adjectives. For example, the method identifies the words *Ras*, *factor*, and *Sos* as core terms from the phrase *Ras guanine nucleotide exchange factor Sos*, and then constructs the whole phrase by applying the rule "Connect nonadjacent annotations if every word between them is either a noun, an adjective, or a numeral."

Proux et al. [23] classify gene names for Drosophila into the following three categories: (1) names including special characters (32%) (e.g., *Hrp54* and *Laer\mt*); (2) names using only lower case letters and belonging to the English language (32%) (e.g., *vamp* and *ogre*); and (3) names using only lower case letters, but not belonging to the English language (36%) (e.g., *ynd* and *zhr*). They argue that the assumption by Fukuda et al., which says that protein names can

be identified according to lexical considerations, does not hold for gene names for Drosophila. Instead of the rule-based approach of Fukuda et al., they employ a tagger with a nondeterministic finite-state automaton that works in three steps: tokenization, lexical look-up, and disambiguation (P = 91.4%, R = 94.4%, F = 92.9). The disambiguation step is based on a hidden Markov model [39], which assigns the tag from a list of relevant candidates for gene names, based on the words that surround the candidates.

Gaizauskas et al. [40] manually construct a context-free grammar for protein name recognition. They first split a protein name into component terms, based on its apparent syntactic structure, and then add corresponding grammar rules in the process of recombining the components. For example, for the enzyme name *calmodulin N-methyltransferase*, they recognize the first word *calmodulin* as a potential 'enzyme modifier' by looking up the dictionary of enzyme modifiers manually constructed from Swiss-Prot and EMTREE (http://www.ovid.com/site/products/fieldguide/embx/EMTREE_Thesaurus.jsp). They also identify the last word *N-methyltransferase* as a potential 'enzyme head', as suggested by the suffix *-ase*. They finally derive the context-free grammar rules in Table 6.9 from the phrase. They construct 160 rules for protein name recognition, and use the names identified with the rules for the applications Enzyme and Metabolic Pathways Information Extraction (EMPathIE) and Protein Active Site Template Acquisition (PASTA). EMPathIE extracts enzyme reactions from articles (enzyme name recognition: P = 96%, R = 98%, F = 97.0), and PASTA extracts protein structure information (protein name recognition: P = 97%, R = 87%, F = 91.7).

6.4.3 Machine Learning Approaches

Rule-based approaches usually utilize handmade rules and patterns, and it is thus difficult to apply them to new domains. This difficulty is quite serious in the biomedical domain, because naming conventions in one research society may be very different from those in another.

Collier et al. [29] use a supervised training method with HMMs to overcome the problem of rule-based approaches (F = 73). The HMM is trained with bigrams, based on lexical and character features in a small corpus of 100 MEDLINE abstracts. For each sentence, the model takes an input that consists

Table 6.9
Example Rules for EMPathIE and PASTA

enzyme —> enzyme_modifier, enzyme.
enzyme —> character, '-', enzyme_head.

of the sequence of words in the sentence and their features. The features used in the model include the presence or absence of each special character in Table 6.4, and whether a word is a determiner or a conjunction. For the given class, the model then calculates the probability of a word belonging to the class. Finally, it produces the sequence of classes with the highest probabilities for the given sequence of words in the sentence. Domain experts mark up, or annotate, the corpus that is used to train the model with classes, such as proteins and DNA.

In order to handle the lack of a training corpus for gene and protein name recognition, Morgan et al. [24] present a method for automatically constructing a large quantity of training corpora by utilizing FlyBase (http://flybase.bio.indiana.edu/), which includes a curated list of genes and the MEDLINE abstracts from which the gene entries are drawn. They apply simple pattern matching to identify gene names or their synonyms in each article (P = 78%, R = 88%, F = 82.7). The noisy corpus, automatically annotated with gene entries of FlyBase, is used to train an HMM for gene name recognition (P = 78%, R = 71%, F = 74.3).

Zhou et al. [17] present another HMM with various features, including word formation patterns (e.g., special characters in Table 6.4); morphological patterns (e.g., prefix and suffix); parts-of-speech; semantic triggers (e.g., head noun trigger and special verb trigger);[11] and name alias features, in order to determine whether a noun phrase denotes the same entity as another noun phrase. However, the great number of features in this model created a data sparseness problem in their system; that is, the lack of a training corpus that is large enough to train the whole set of proposed features. For this reason, they propose a k-NN algorithm, which estimates the probabilities in the model by utilizing the K nearest neighbors of frequently occurring output pattern entries; that is, by restricting the output of the system only to the frequent patterns. They further present a pattern-based postprocessing technique, automatically extracting rules from the training data to deal with the cascaded entity name phenomenon (e.g., <PROTEIN> <DNA>kappa 3</DNA> binding factor </PROTEIN>). The implemented system is evaluated on the GENIA corpus (F = 83.6 ~ 86.2).

Among the features used by Zhou et al., Collier and Takeuchi [41] compare two features, character-level orthographic features and part-of-speech tags. From their experiments with support vector machines, they report an F-score of 72.6 for character-level features, and 68.6 for part-of-speech features. Most unlikely, when they experiment with a combination of both features, they report an F-score of 72.3, slightly lower than that for character-level features. They

11. A head noun is the major noun of a noun phrase. For example, *B cells* in the name *activated human B cells*. Special verbs include verbs indicating protein-protein interactions, such as *activate*, *bind*, and *inhibit*.

conclude that the reason is that the NER system of Zhou et al. can incorporate the evidence from part-of-speech tags into other features in a highly sophisticated way, with the help of the k-NN algorithm, to reduce the search space.

6.4.4 Hybrid Approaches

As the number of features for machine learning systems increases to cover more phenomena in NER, the data sparseness problem becomes more serious. Since the three kinds of approaches discussed above have their own advantages and disadvantages, there is a clear need for combining them for better performance.

In fact, some of the methods introduced in the previous sections already are a hybrid of different kinds of approaches. For example, Hanisch et al. [26] utilize a machine learning technique for computing optimized parameters of scoring measures in a dictionary-based system. Proux et al. [23] also apply a machine learning technique for disambiguation of relevant candidate gene names in a rule-based system. Zhou et al. [17] automatically construct rules to deal with cascaded entity names for their machine learning system. We introduce additional systems next.

Tanabe and Wilbur [26] present a combination of statistical- and knowledge-based strategies. They first apply automatically generated rules from Brill's POS tagger [42] to extract single words of gene and protein names, and apply manually generated rules that recognize the beginning and ends of gene and protein names, in order to find multiword gene and protein names. For example, they extract a rule "NNP gene fgoodleft GENE", which changes the tag of a word from NNP to GENE if the word *gene* can appear to the right, where NNP indicates a proper noun. Then they utilize manually generated rules formed from morphological clues to recover false negative results. For example, they use a contextual rule "x CC ANYGENE", which changes the tag of 'x' to CONTEXTGENE, with the following constraints: 'x' must be a noun, adjective, cardinal number, or preposition; CC must be a coordinating conjunction, ANYGENE must be any gene name identified by the system, and CONTEXTGENE must be a gene name not identified by Brill's tagger but by the recovering rule.

Mika et al. [25] present a system that combines a preprocessing dictionary- and rule-based filtering step with several independently trained SVMs [43] for protein name recognition (P = 76%, R = 75%, F = 75.5). For the dictionary to filter irrelevant words, such as common words, medical terms, and species names, the authors utilize the online-version of the Merriam-Webster dictionary (http://www.m-w.com/), dictionary of medical terms (DMT) (http://cancerweb.ncl.ac.uk/omd/), and species names in UniProt Knowledgebase (http://us.expasy.org/cgi-bin/speclist). They also use a rule to filter names

that are followed by *cell(s)* or *cyte(s)* (e.g., *CD4+T lymphocytes*, *Streptococcus mutans cells*).

6.4.5 Classification and Grounding of Biomedical Named Entities

The names recognized by the systems of biomedical NER often need further classification, when the recognized NEs are ambiguous with respect to their class information, or when the desired selection is for a restricted class of names, such as enzyme names. Note that NE classification is different from term classification, discussed in Chapter 4, because the recognized term from automatic term recognition already corresponds to a concept in the semantic hierarchy in a terminology management system, therefore mapping naturally into some classification. Named entities may be further reclassified into concepts in a relevant semantic hierarchy, using later steps according to their characteristics, such as biological functions.

Torii et al. [44] examine various information sources for classifying biomedical names: (1) name-internal features, such as headwords and suffixes; (2) contextual information, or the occurrence of other words or phrases near the given names; and (3) other sources, including acronyms, apposition, and coordination (F = 86).

Spasic et al. [45] present a method of classifying gene and protein names into ontological concepts, based on verb selectional patterns, where such a pattern is automatically defined as a set of semantic classes of domain-specific verbs that are automatically identified in a corpus, based on their frequencies of occurrence. Given a name, the verb selectional pattern of the most frequently cooccurring domain-specific verb is used to constrain the search space for classification, by focusing on potential classes of the given name.

The recognized and classified terms can be utilized as indices for biomedical documents only when they are sufficiently grounded with relevant entry IDs in biomedical resources, such as Swiss-Prot [21, 22].[12] The results of dictionary-based approaches can be easily grounded with IDs in the dictionary, but it is not straightforward to ground those of the other approaches [33].

Kim and Park [30] present a grounding method that associates gene and protein names identified in MEDLINE abstracts with Swiss-Prot entry names. The method deals with variations of the kinds (1–3) in Table 6.4 with simple patterns, and selects Swiss-Prot entries whose species information corresponds to the local context of identified gene and protein names in an abstract. An example is *mouse* and *human* from the protein names *mouse 6Ckine* and *human 6Ckine*, respectively (P = 59.5%, R = 40.7%, F = 48.3).

12. The grounding task is also referred to as term mapping, and the sequential combination of term recognition, term classification, and term mapping is called term identification [22].

Pustejovsky et al. [46] present a simple method for disambiguating acronyms with multiple meanings (P = 97.6%), where many ambiguous names in MEDLINE abstracts are abbreviations, and the clarifying descriptions of acronyms can be grounded without ambiguities. The method scans for the descriptions in the previous context of a given acronym, since the clarifying descriptions usually occur in the previous context of the same abstracts. Pakhomov [47] also presents a method for the disambiguation of acronyms, by utilizing a maximum-entropy classifier with only four adjacent words (P = 90%).

6.5 Discussion

The reported performances of the approaches discussed in the previous section cannot be directly compared with one another, because of the lack of common evaluation metrics and test corpora [48]. For this reason, researchers in the field have initiated several evaluation conferences for biomedical NER. BioCreAtIve 2003 (http://www.mitre.org/public/biocreative/) presents two tasks for information extraction. The first task is to recognize human gene names in MEDLINE abstracts, and the second task is to annotate the recognized genes with the terms of the Gene Ontology [49]. The second task is motivated to automating the GOA project [50], by making use of the biomedical literature. JNLPBA (http://www.genisis.ch/~natlang/JNLPBA04/), held in conjunction with the International Conference on Computational Linguistics (COLING), shared the task of NER with its participants [51]. The participants used an extended version of the GENIA corpus [38]. Chapter 9 explains in detail such evaluations of terminology management. Note that the target sets of gene names for these competitions may be different from each other, since the GENIA corpus does not annotate anaphoric expressions. Participants in BioCreAtIve 2003 may need to recognize anaphoric expressions that refer to gene names, as discussed in Section 6.1.

When we want to understand the current status of approaches to biomedical NER, it would be worthwhile to examine some approaches to the task in the general domain, and compare them with the approaches in the previous section.

Yangarber and Grishman [52] present a simple method of matching name patterns to proper names in texts. Their method utilizes lexical cues, such as capitalizations, personal titles (e.g., *Mr.* or *Esq.*), and company suffixes (e.g., *Inc.* or *Co.*). However, patterns of this kind can be matched only to a small set of NEs, but not to other NEs that do not host such lexical cues (see Section 6.4.2). In order to consider other kinds of cues, as well as lexical ones, Zhou and Su [53] present an HMM-based system that integrates four types of internal and external evidence: (1) simple deterministic internal features of the words, such as capitalization and digitalization; (2) internal semantic features of important

triggers, such as suffixes and prefixes; (3) internal gazetteer features, such as names of persons, organizations, and locations; and (4) external macro context features. There is rarely enough training data to compute accurate probabilities in an HMM. To overcome this data sparseness problem, the authors apply back-off modeling, which approximates the trained model based on the importance of features or combinations of features [17, 24]. They report that their system achieves F-scores of 96.6 and 94.1, when evaluated on MUC-6 and MUC-7 English NER tasks, respectively.

Borthwick et al. [54] presented a statistical system in a maximum-entropy framework, which is reported to show the existing then-state-of-the-art performance in 1998. The system makes use of various features, including capitalization features, lexical features, and features indicating the current section of text (e.g., headline or main body), as well as using dictionaries of single or multiword terms. When the system is combined with three hand-coded systems—Proteus [55], IsoQuest [56], and a system from the University of Manitoba [57]—it is reported to yield an F-score of 97.12, which is highly comparable to those of human taggers of MUC-7, with F-scores of 96.95 and 97.60 [58] (see Section 6.4.4). However, they have not explained why the combined system outperforms the original system, and what makes the system produce such state-of-the-art performance. It is not yet clear if the combined system can be applied to other domains, without redundancy among combined component systems.

There are also several approaches to NER for language-independent systems. According to Palmer and Day [9], it is possible to perform much of the multilingual entity recognition task with a very simple analysis of texts, using Zipf's Law [59], achieving a recall score higher than 70 for some languages. However, they also contend that making incremental advances above the baseline can be arduous and very language-specific. In order to build a minimally language-dependent system for both NER and classification, Cucerzan and Yarowsky [60] present an iterative learning method that learns from unannotated texts. Their method achieves a competitive performance, with F-scores ranging from 70.5 to 75.4, when trained on a very short labeled name list, and without using other resources. This result is consistent with the observation of Mikheev et al. [61], who also have utilized relatively small gazetteers of well-known names, rather than large gazetteers of low-frequency names. These language-independent approaches also are useful for biomedical NER, because of different characteristics of biomedical NEs (see the comparison of gene names for Drosophila with protein names in [23]).

In summary, the approaches to biomedical NER have evolved similar to those in general domains, probably because many of the methods for biomedical NER are actually adopted from, or further enhanced from, those for the competitions of MUCs in general domains. Machine learning approaches to general

NER have utilized various techniques, such as HMM [53, 62, 63], decision tree [64, 65], and Maximum Entropy (ME) [66], while approaches to biomedical NER have usually utilized HMMs. The state-of-the-art performance in general NER is achieved by a hybrid method [54], but the hybrid approaches to biomedical NER have not yet shown such performance, possibly due to the lack of high quality linguistic resources in biomedicine (see Chapter 3). The approaches to biomedical NER have focused almost exclusively on gene and protein names, while approaches to general NER have dealt with language-independent issues as well. It certainly is an open problem to deal with different kinds of NEs in a uniform way.

6.6 Conclusion

In this chapter, we describe the task of NER from the biomedical literature, along with previous approaches to the task. The reported performance of the systems implementing these approaches shows that some of them are practical enough to be utilized for larger biomedical text mining systems. However, we should note that there is still a need for evaluating them with a common evaluation metric and a common testbed, as discussed in Section 6.5. Much more effort is required to enhance, or fine-tune, the systems, as pointed out by Palmer and Day [9]. We also need to consider the fact that the approaches may not be applicable in a straightforward manner to other systems with different goals, since these approaches are designed only for certain fixed purposes (see Section 6.3.5). This observation leads us to the potentially synergistic influence of NER on biomedical text mining systems, which requires a more integrated development of NER methods, according to their host text mining systems.

References

[1] Benson, D. A., et al., "GenBank: Update," *Nucleic Acids Research*, Vol. 32, Database Issue, 2004, pp. D23–D26.

[2] Bairoch, A., et al., "The Universal Protein Resource (UniProt)," *Nucleic Acids Research*, Vol. 33, Database Issue, 2005, pp. D154–D159.

[3] Wain, H. M., et al., "Guidelines for Human Gene Nomenclature," *Genomics*, Vol. 79, No. 4, 2002, pp. 464–470.

[4] Petsko, G. A., "What's in a Name," *Genome Biology*, Vol. 3, No. 4, March 2002, pp. comment: 1005.1–2.

[5] Pearson, H., "Biology's Name Game," *Nature*, Vol. 411, June 2001, pp. 631–632.

[6] Grishman, R., and B. Sundheim, "Message Understanding Conference-6: A Brief History," *Proc. 16th Int. Conf. on Computational Linguistics*, 1996, pp. 466–471.

[7] Chinchor, N., "MUC-7 Information Extraction Task Definition," *Proc. 7th Message Understanding Conf.*, 1998.

[8] Chinchor, N., and B. Sundheim, "MUC-5 Evaluation Metrics," *Proc. 5th Message Understanding Conf.*, 1995, pp. 69–78.

[9] Palmer, D. D., and D. S. Day, "A Statistical Profile of the Named Entity Task," *Proc. 5th Conf. on Applied Natural Language Processing*, 1997, pp. 190–193.

[10] Van Rijsbergen, C. J., *Information Retrieval*, Newton, MA: Butterworth-Heinemann, 1979.

[11] Jackson, P., et al., "Information Extraction from Case Law and Retrieval of Prior Cases," *Artificial Intelligence*, Vol. 150, No. 1–2, 2003, pp. 239–290.

[12] Wilbur, W. J., et al., "Analysis of Biomedical Text for Chemical Names: A Comparison of Three Methods," *Proc. AMIA Annual Symp.*, 1999, pp. 176–180.

[13] Collier, N., C. Nobata, and J. Tsujii, "Automatic Acquisition and Classification of Terminology using a Tagged Corpus in the Molecular Biology Domain," *Journal of Terminology*, Vol. 7, No. 2, 2001, pp. 239–257.

[14] Rindflesch, T. C., et al., "EDGAR: Extraction of Drugs, Genes and Relations from the Biomedical Literature," *Proc. Pacific Symp. on Biocomputing*, 2000, pp. 517–528.

[15] Franzen, K., et al., "Protein Names and How to Find Them," *International Journal of Medical Informatics*, Vol. 67, No. 1–3, 2002, pp. 49–61.

[16] "Obstacles of Nomenclature," *Nature*, Vol. 389, 1997, p. 1.

[17] Zhou, G., et al., "Recognizing Names in Biomedical Texts: A Machine Learning Approach," *Bioinformatics*, Vol. 20, No. 4, 2004, pp. 1178–1190.

[18] Kim, J.-J., and J. C. Park, "Annotation of Gene Products in the Literature with Gene Ontology Terms Using Syntactic Dependencies," *IJCNLP 2004*, K.-Y. Su, et al., (eds.), Lecture Notes in Artificial Intelligence 3248, Berlin: Springer-Verlag, 2005, pp. 787–796.

[19] Jacquemin, C., "Syntagmatic and Paradigmatic Representation of Term Variation," *Proc. Annual Meeting of the Association for Computational Linguistics*, University of Maryland, 1999, pp. 341–348.

[20] Hirschman, L., A. A. Morgan, and A. S. Yeh, "Rutabaga by Any Other Name: Extracting Biomedical Name," *Journal of Biomedical Informatics*, Vol. 35, 2002, pp. 247–259.

[21] Tuason, O., et al., "Biomedical Nomenclatures: A Source of Lexical Knowledge and Ambiguity," *Proc. Pacific Symp. on Biocomputing*, 2004, pp. 238–249.

[22] Krauthammer, M., and G. Nenadic, "Term Identification in the Biomedical Literature," *Journal of Biomedical Informatics*, Vol. 37, No. 6, 2004, pp. 393–395.

[23] Proux, D., et al., "Detecting Gene Symbols and Names in Biomedical Texts: A First Step Toward Pertinent Information," *Proc. 9th Workshop on Genome Informatics*, 1998, pp. 72–80.

[24] Morgan, A., et al., "Gene Name Identification and Normalization Using a Model Organism Database," *Journal of Biomedical Informatics*, Vol. 37, 2004, pp. 396–410.

[25] Mika, S., and B. Rost, "Protein Names Precisely Peeled off Free Text," *Bioinformatics*, Vol. 20, Suppl. 1, 2004, pp. i241–i247.

[26] Tanabe, L., and W. J. Wilbur, "Tagging Gene and Protein Names in Biomedical Text," *Bioinformatics*, Vol. 18, No. 8, 2002, pp. 1124–1132.

[27] Fukuda, K., et al., "Toward Information Extraction: Identifying Protein Names from Biomedical Papers," *Proc. Pacific Symp. on Biocomputing*, 1998, pp. 707–718.

[28] Hanisch, D., et al., "Playing Biology's Name Game: Identifying Protein Names in Scientific Text," *Proc. Pacific Symp. on Biocomputing*, 2003, pp. 403–414.

[29] Collier, N., C. Nobata, and J. Tsujii, "Extracting the Names of Genes and Gene Products with a Hidden Markov Model," *Proc. 17th Int. Conf. on Computational Linguistics*, 2000, pp. 201–207.

[30] Kim, J.-J., and J. C. Park, "BioAR: Anaphora Resolution for Relating Protein Names to Proteome Database Entries," *Proc. ACL Workshop on Reference Resolution and Its Applications*, 2004, pp. 79–86.

[31] Jenssen, T. K., et al., "A Literature Network of Human Genes for High-Throughput Analysis of Gene Expression," *Nature Genetics*, Vol. 28, No. 1, 2001, pp. 21–28.

[32] Ono, T., et al., "Automated Extraction of Information on Protein-Protein Interactions from the Biological Literature," *Bioinformatics*, Vol. 17, No. 2, 2001, pp. 155–161.

[33] Tsuruoka, Y., and J. Tsujii, "Improving the Performance of Dictionary-Based Approaches in Protein Name Recognition," *Journal of Biomedical Informatics*, Vol. 37, 2004, pp. 461–470.

[34] Krauthammer, M., et al., "Using BLAST for Identifying Gene and Protein Names in Journal Articles," *Gene*, Vol. 259, No. 1–2, December 2000, pp. 245–252.

[35] Altschul, S. F., et al., "Gapped BLAST and PSI-BLAST: A New Generation of Protein Database Search Programs," *Nucleic Acids Research*, Vol. 25, No. 17, 1997, pp. 3389–3402.

[36] Bennett, K. P., and O. L. Mangasarian, "Robust Linear Programming Discrimination of Two Linearly Inseparable Sets," *Optimization Methods and Software*, Vol. 1, 1992, pp. 23–34.

[37] McCallum, A., and K. Nigam, "A Comparison of Event Models for Naïve Bayes Text Classification," *Proc. AAAI-98 Workshop on Learning for Text Categorization*, 1998, pp. 41–48.

[38] Kim, J. D., et al., "GENIA Corpus—A Semantically Annotated Corpus for Bio-Textmining," *Bioinformatics*, Vol. 19, Suppl. 1, 2003, pp. i180–i182.

[39] Kupiec, J., "Robust Part-of-Speech Tagging Using a Hidden Markov Model," *Journal of Computer Speech and Language*, Vol. 6, No. 3, 1992, pp. 225–242.

[40] Gaizauskas, R., G. Demetriou, and K. Humphreys, "Term Recognition and Classification in Biological Science Journal Articles," *Proc. Workshop on Computational Terminology for Medical and Biological Applications*, Patras, Greece, 2000, pp. 37–44.

[41] Collier, N., and K. Takeuchi, "Comparison of Character-Level and Part of Speech Features for Name Recognition in Biomedical Texts," *Journal of Biomedical Informatics*, Vol. 37, 2004, pp. 423–435.

[42] Brill, E., "Some Advances in Transformation-Based Part of Speech Tagging," *Proc. 12th National Conf. on Artificial Intelligence*, 1994, pp. 722–727.

[43] Cortes, C., and V. Vapnik, "Support Vector Networks," *Machine Learning*, Vol. 20, No. 3, 1995, pp. 273–297.

[44] Torii, M., S. Kamboj, and K. Vijay-Shanker, "Using Name-Internal and Contextual Features to Classify Biomedical Terms," *Journal of Biomedical Informatics*, Vol. 37, 2004, pp. 498–511.

[45] Spasic, I., and S. Ananiadou, "Using Automatically Learnt Verb Selectional Preferences for Classification of Biomedical Terms," *Journal of Biomedical Informatics*, Vol. 37, 2004, pp. 483–497.

[46] Pustejovsky, J., et al., "Automatic Extraction of Acronym-Meaning Pairs from MEDLINE," *Proc. Medinfo*, 2001, pp. 371–375.

[47] Pakhomov, S., "Semi-Supervised Maximum Entropy Based Approach to Acronym and Abbreviation Normalization in Medical Texts," *Proc. 40th Annual Meeting of the Association for Computational Linguistics*, 2002, pp. 160–167.

[48] Hirschman, L., et al., "Accomplishments and Challenges in Literature Data Mining for Biology," *Bioinformatics*, Vol. 18, No. 12, 2002, pp. 1553–1561.

[49] The Gene Ontology Consortium, "The Gene Ontology (GO) Database and Informatics Resource," *Nucleic Acids Research*, Vol. 32, Database Issue, 2004, pp. D258–D261.

[50] Camon, E., et al., "The Gene Ontology Annotation (GOA) Database—An Integrated Resource of GO Annotations to the UniProt Knowledgebase," *In Silico Biology*, Vol. 4, No. 1, 2004, pp. 5–6.

[51] Kim, J.-D., et al., "Introduction to the Bio-Entity Recognition Task at JNLPBA," *Proc. Int. Joint Workshop on Natural Language Processing in Biomedicine and its Applications, COLING 2004 Post-Conference Workshop*, 2004, pp. 70–75.

[52] Yangarber, R., and R. Grishman, "NYU: Description of the Proteus/PET System as Used for MUC-7 ST," *Proc. 7th Message Understanding Conf.*, 1998.

[53] Zhou, G. D., and J. Su, "Named Entity Recognition Using an HMM-Based Chunk Tagger," *Proc. 40th Annual Meeting of the Association for Computational Linguistics*, July 2002, pp. 473–480.

[54] Borthwick, A., et al., "Exploiting Diverse Knowledge Sources via Maximum Entropy in Named Entity Recognition," *Proc. 6th Workshop on Very Large Corpora*, August 1998, pp. 152–160.

[55] Grishman, R., "The NYU System for MUC-6 or Where's the Syntax?" *Proc. 6th Message Understanding Conf.*, 1995.

[56] Krupka, G. R., and K. Hausman, "IsoQuest: Description of the NetOwl Extractor System as Used in MUC-7," *Proc. 7th Message Understanding Conf.*, 1998.

[57] Lin, D., and I. Nalante, "Using Collocation Statistics in Information Extraction," *Proc. 7th Message Understanding Conf.*, 1998.

[58] Marsh, E., and D. Perzanowski, "MUC-7 Evaluation of Information Extraction Technology: Overview of Results," *Proc. 7th Message Understanding Conf.*, 1998.

[59] Zipf, G. K., *Selected Studies of the Principle of Relative Frequency in Language*, Cambridge, MA: Harvard University Press, 1932.

[60] Cucerzan, S., and D. Yarowsky, "Language Independent Named Entity Recognition Combining Morphological and Contextual Evidence," *Proc. Joint SIGDAT Conf. on Empirical Methods in Natural Language Processing and Very Large Corpora*, 1999, pp. 90–99.

[61] Mikheev, A., M. Moens, and C. Grover, "Named Entity Recognition Without Gazetteers," *Proc. 9th Conf. of the European Chapter of the Association for Computational Linguistics*, 1999, pp. 1–8.

[62] Bikel, D. M., R. Schwartz, and R. M. Weischedel, "An Algorithm That Learns What's in a Name," *Machine Learning*, Vol. 34, No. 1–3, 1999, pp. 211–231.

[63] Klein, D., et al., "Named Entity Recognition with Character-Level Models," *Proc. 7th Conf. on Natural Language Learning*, 2003, pp. 180–183.

[64] Bennett, S. W., C. Aone, and C. Lovell, "Learning to Tag Multilingual Texts Through Observation," *Proc. 2nd Conf. on Empirical Methods in Natural Language Processing*, 1996, pp. 109–116.

[65] Sekine, S., "Description of the Japanese NE System Used for MET-2," *Proc. 7th Message Understanding Conf.*, 1998.

[66] Chieu, H. L., and H. T. Ng, "Named Entity Recognition: A Maximum Entropy Approach Using Global Information," *Proc. 9th Int. Conf. on Computational Linguistics*, 2002, pp. 786–791.

7

Information Extraction

John M[c]Naught and William J. Black

7.1 Information Extraction: The Task

Simply put, we can characterize information extraction (IE) as follows:

- Take a natural language text from a document source, and extract the essential facts about one or more predefined fact types.
- Represent each fact as a template whose slots are filled on the basis of what is found from the text.

A template is a "form" that conveys a fact when filled. The form has labels, one of which says what type of fact it represents, and the others (the slots) identify the attributes that make up the fact. There are different types of facts. We will be interested here mainly in simple facts and events. Examples of simple facts are:

James Smith works for XYZ Co.
James Smith, Chief Research Scientist of XYZ Co.
Binding of hsp90 to the glucocorticoid receptor requires a specific 7-amino acid sequence at the amino terminus of the hormone-binding domain.

Examples of events are:

XYZ Co. announced the appointment of James Smith as Chief Research Scientist on 4th August 2005.
We hypothesized that retinoic acid receptor (RAR) would activate this gene.

Unless otherwise indicated, we use "fact" to cover both simple facts and events.

A template for the report of the death of a famous person could have slots for the name, the age, the place, the date, and the cause of death. Slots are typically filled by named entities (see Chapter 6), or, in more complex representations, by other facts. In the news domain, named entities are people, organizations, places, dates, numbers, amounts of money, and so forth. IE is increasingly being applied to scholarly papers, where the important entities include the objects of the science (e.g., substances, organisms, apparatus, and cited papers).

IE is typically carried out in support of other tasks, and usually forms part of some application or pipeline of processes. The results of IE typically are either stored in databases and subjected to data mining algorithms or querying [1]; integrated in knowledge bases to allow reasoning; or presented directly to users who require support in dealing with identification, assembly, and comparison of facts (as in data curation tasks). Whereas other upstream tasks in text mining may be carried out largely independently of users' requirements, the fact extraction aspects of IE are often highly dependent on the users specifying the kind of facts they wish extracted. Most IE systems eschew full-scale analysis of texts (i.e., full parsing to achieve one or more representations for each sentence), in favor of partial analysis. It is critical to determine, for any application, what the user wishes the partial analysis to yield.

In this chapter, we focus on recent research into rule-based approaches to fact extraction, and do not consider approaches based on machine learning techniques. A good overview of machine learning techniques for IE from biotext is provided by Nédellec [2], who notes that there are few attempts to apply such techniques to fact extraction, a main reason being the lack of appropriately annotated text for training data. Bunescu et al. [3] offer detailed comparison and evaluation of machine learning techniques to extract protein interactions. In Section 7.2, we place IE in its general context. We then focus in Section 7.3 on the particular problem of extracting relations (simple facts) and events from texts in the biosciences. First, we must be clear how IE is distinguished from its adjacent neighbors.

7.1.1 Information Extraction and Information Retrieval

IE is an application of natural language processing (NLP). As the term implies, the goal is to extract information from text, and the aim is to do so without requiring the end user of the information to read the text.

In contrast, information retrieval (IR) is the activity of finding documents that answer an information need with the aid of indexes. IR systems now tend to be called "search engines," and almost all computer users make habitual use of examples such as Google and PubMed.

Having used a search engine, the user must read each document to know the facts reported in it. However, when the goal is to be able to tabulate the facts reported in large numbers of documents in a literature source, IE becomes a more relevant technology. IE can be used to support a fact retrieval service, or as a step towards text mining based on conceptually annotated text. We sum up the contrasts between IE and IR in Table 7.1.

7.1.2 Information Extraction and Natural Language Processing

Natural Language Processing (NLP) is the activity of processing natural language texts by computer to access their meaning [4, 5]. Chapter 2 considers the various linguistic levels involved in NLP.

NLP systems can analyze (parse) well-formed sentences of great complexity, if the grammar of the language has been encoded for the system, and the lexical resources (dictionary) cover the vocabulary used. This is only feasible on the small scale. In a large body of documents, a significant proportion of sentences will not be fully recognized by the grammatical and lexical resources of any given NLP parser. NLP systems need ways to deal robustly with parts of the text that fall outside these resources.

In IE, where texts are drawn from a single domain of discourse, one meaning of a potentially ambiguous term tends to predominate, and so lexical ambiguity is less of a problem than it would be in open domain NLP. Structural and referential ambiguities can still present a challenge. IE systems have dealt primarily with news sources, and, more recently, with scientific publications. In

Table 7.1
Contrasting Information Extraction with Information Retrieval

Information Retrieval (IR)	Information Extraction (IE)
Returns documents.	Returns facts.
Is a classification task (each document is relevant/not relevant to a query).	Is an application of Natural Language Processing, involving the analysis of text and synthesis of a structured representation.
Can be done without reference to syntax (treating the query and indeed the documents as merely a "bag of words").	Is based on syntactic analysis and semantic analysis.

scientific domains, a general language grammar and dictionary are not enough. The news is dominated by proper names of people, places, and organizations, only a few of which are prominent enough to merit listing in the dictionary. Scientific fields use many technical terms, only a few of which are found in common discourse often enough to merit inclusion in general dictionaries. To some extent, these kinds of items can be listed in auxiliary terminologies or simple lists that supplement the dictionary. However, it is usual for an IE system to attempt to classify unlisted named entities on the basis of their internal structure and the context in which they occur (see Chapter 6). To do this in scientific domains, they must increasingly rely on term extractors (discussed in Chapter 4).

In both news and scientific articles, natural language sentences also incorporate expressions that cannot be exhaustively listed. These include dates and monetary amounts in the news, and formulas and measurements in the sciences.

7.2 The Message Understanding Conferences

Much NLP research was funded by U.S. government agencies in the 1970s and 1980s, but these agencies became frustrated at the difficulty of evaluating competing approaches, when researchers chose their own issues, processing methods, evaluation methods, and data. The solution was to establish a methodology of competitive evaluation, where sponsored researchers and others would agree to develop systems to process the same data, and formalize their analysis results in a standard notation. A series of seven "Message Understanding Conferences" (MUC) was held, with the last in 1998 [6]. The names of the various tasks identified in MUC evaluations, and the methods used for evaluation in MUC, have become widely adopted and adapted outside MUC. Familiarity with IE, as defined in MUC, is advantageous to aid understanding of IE carried out under other auspices (see Chapter 9).

MUC also had a strong influence on the design of IE systems. Systems are usually modular, with a module robustly doing one level of analysis. Hobbs [7, 8] identifies the components of a generic IE system: tokenizer, sentence splitter, tagger, morphological analyzer, chunker, gazetteer, NE module, discourse module, template extractor, and template combiner. Chapter 2 goes into further detail on this aspect.

7.2.1 Targets of MUC Analysis

The ultimate goal of IE is the extraction of templates representing facts of various kinds, including events. In the later MUC evaluations, the analysis was broken down into component tasks (see Table 7.2).

Table 7.2
MUC Component Tasks

Task	Description
Named Entity (NE) Task	Extracts names of people, organizations and locations; and numeric and temporal expressions (see Chapter 6)
Coreference (CR) Task	Links references to the same entity (see Chapter 7)
Template Element (TE) Task	Extracts identifying and descriptive attributes of NEs
Template Relation (TR) Task	Extracts specific relationships between NEs (simple facts)
Scenario Template (ST) Task	Extracts events. For each type of event extracted, one or more slots are filled with instances of TEs or TRs.

Template elements are found by looking for patterns such as those in (7.1).

a. *name* is a *description*
b. *name*, the *description*
c. *name*, the *country*-based *description* (7.1)

Examples of TRs include the 'works_for' relationship between a person and a company, or the 'located_in' relationship between a place and a larger place (e.g., a city and a country). To say that *Fred Smith was appointed as Director of XYZ Co* is to report an event of appointment. To say that *Fred Smith, Director of XYZ Co* did something is to convey the relationship between *Fred Smith* and *XYZ Co*, in the course of reporting a different event. Relations are states rather than events, and are often referred to as "simple facts." Relations are not always conveyed in the text in the form of an assertion. Apposition (i.e., two noun phrases adjacent to each another) often indicates a TR, as illustrated in (7.2). The semantic relation implied by apposition varies with the classes of the individual noun phrases. In (7.2a) and (7.2b), because the noun phrases are drawn from the three classes 'organization', 'job_title', and 'person_name', the ternary relation that the person works for the company in the capacity of the job title is appropriately instantiated. In (7.2c), where there is a conventional address format (e.g., Paris, France), the relation 'located_in' is instantiated between the first named place and the second named area.

a. *company* president *first-name*, *surname*
b. the *job-title* of *company*, *first-name*, *surname*, announced…
c. *place-name*, *place-name* (7.2)

Regarding STs, the arguments or slot fillers (TEs or TRs) have to be identified first. For example, a 'management succession event' is an announcement that the holder of some corporate office has changed. In such an event, we need to have identified the following arguments: the company, the new officeholder, the office, perhaps the outgoing incumbent, and the date of the change. There is variety in the way events can be related. Two ways to report a management succession event might be (7.3a) and (7.3b).

a. *company* today announced the appointment of *person-name* as *job*.
(S)he succeeds *person-name* who has been headhunted by *company*. (7.3)
b. person-name has been replaced as *job* by *person-name*, *company*
announced today.

7.3 Approaches to Information Extraction in Biology

In what follows, we do not consider the problem of NE recognition, since Chapter 6 is devoted to this topic. We concentrate on the "higher" tasks of IE, namely the extraction of template relations (simple facts) and scenario templates (events), and on the task of coreference analysis. We give an overview of IE approaches and attempts, insofar as they relate to addressing problems in biology.

Recent review articles discussing the application of NLP techniques in general to biology include [8–17].

When we consider the relative success of various techniques below, we must remember that "higher" analysis tasks typically depend crucially on good results from the "lower" analysis tasks (e.g., tokenization, sentence splitting, part-of-speech tagging, terminology, and NE recognition). In general, results appear depressed when compared to the state of the art of higher IE in areas such as business newswire processing. This is not to say that the techniques being used on biology texts are inferior. They are typically the same techniques as those used on texts from other domains. However, since later stages of processing typically build on the results of earlier stages, a poorer degree of performance at an earlier stage will have detrimental effects on later stages. As other chapters in this book describe, terminology (see Chapter 4) and NE recognition for biology texts (see Chapter 6) are challenging tasks. Until these tasks deliver results comparable to those seen in other application domains, we will continue to see relatively poorer results for such tasks as scenario template extraction, in comparison with the state of the art elsewhere. Systems processing MUC-type data yields an approximate 80% F-score for identification of template relations, but only an approximate 60% F-score for extraction of events. See [6, 18] for a further discussion of MUC. The challenge for IE in the biosciences is to do at least

as well. Chapter 9 discusses progress in text mining as measured in recent challenge evaluations. However, these evaluations have not focused on the higher tasks of IE. This is also true for other concerns, such as evaluating attempts to support biodatabase curation, through identifying passages giving evidence for a Gene Ontology annotation or through annotating a protein with a Gene Ontology concept.

This is not to criticize these evaluations, which have had positive effects on the field (see Chapter 9), but simply to note that the evaluation of event and simple fact extraction has not yet been conducted in challenge evaluations in biology. Rebholz-Schuhmann et al. are optimistic that fact extraction in biology is "only a matter of time and effort" [19]. However, they refer to fact extraction in the context of full-scale analysis of language, which they conclude as being a long way from being up to the task. Fact extraction can offer useful output through shallow analysis, as is commonly practiced in IE. With the advances being made at the level of NE and terminology recognition in biotexts, improved results from fact extraction can be expected in the very near future.

We offer a classification of various approaches to the higher tasks of IE in biology, together with detailed examinations of the techniques used by various systems. Our objective is both to describe the wide range of approaches, and to provide a basis for appreciating the advantages and limits of the systems discussed (e.g., whether their output can be used for data mining or other tasks). We focus on systems that have been described in the recent research literature. As Chapter 9 makes clear, exact comparison depends on a common evaluation scheme and common gold standard annotated data, which are not yet developed for these higher tasks. Thus, the reader must interpret with due caution any reported recall, precision, or F-measure scores, and is advised to consult the cited sources for further detail. Furthermore, some systems defy any single classification, since they use a multitude of strategies, components, and resources. We have nevertheless assigned such systems to a single class, while making cross reference to others in terms of the features they display.

7.3.1 Pattern-Matching Approaches

As in other areas involving NLP, there are attempts to exploit basic pattern matching techniques to extract useful information from biotexts. These attempts suffer from similar problems, wherever they are applied with a view to obtaining what is essentially semantic or conceptual information. Some measure of semantic processing beyond pattern matching is required over either text strings or annotations connected with surface analyses (e.g., part-of-speech annotations). Moreover, the closer an analysis is to the text, the more patterns that are needed to take account of the large amount of surface grammatical variation in texts. Those who concentrate on accounting for the wide variation in

surface expression face the Zipfian characteristics of text with respect to patterns [19]. Many patterns are seen only once, or a few times, even in a large collection, and one can never gather enough text to ensure that all possible patterns can be accounted for. Thus, there are early upper limits on processing involving basic pattern matching.

Another inherent problem with such approaches is that one cannot easily exploit significant syntactic generalizations, such as the relationship between passive and active sentences. Extraction of phrases by basic pattern matching alone does not solve the problem of semantic and conceptual characterization of phrases. One must keep in mind the goal of particular analyses using basic pattern matching. An early attempt is due to Blaschke et al. [20], and more recent systems still rely on similar techniques. For example, Divoli and Attwood report on the BioIE system [21], which uses patterns to extract entire sentences related to protein families, protein structures, functions, and diseases. Technically, this is a sentence extraction system rather than an information extraction system, since there is no attempt to extract and represent information content in an abstract form. However, such sentence extraction is undoubtedly useful for the reader, since it presents the reader with relevant sentences rather than entire documents, which is an improvement over classic IR. In a curation environment where fact-containing sentences from the literature are linked to biodatabase entries, this system can offer useful support to the curator. A similar approach is adopted by Vailaya et al. [22], with additional user interaction to assign parts of extracted sentences to roles of an interaction between genes or proteins.

The work by Ono et al. [23] and Huang et al. [24] is representative of a class of systems that undertakes slightly deeper analysis, by first tagging text for part-of-speech. Ono et al. apply regular expressions over the part-of-speech tags, including recognition of simple coordination. Rules target simple assertions about protein-protein interaction, utilizing stemming and a small number of keywords. These keywords typically comprise common verbs, such as *interact* or *bind*, and their related deverbal nouns (*interaction*, *binding*). Huang et al. apply sentence alignment and pattern matching to generate patterns that may be useful for extracting protein-protein interactions. Whereas Ono et al. process MEDLINE abstracts, Huang et al. process full articles.

The BioRAT system [25] also belongs to this family. It offers a means to construct templates involving simple regular expressions using stems, part-of-speech tags, gazetteer categories, literal strings, and words. Templates apply to extract matching phrases. Primitive filters are provided, for example, to reject any verbs from consideration as proteins.

The RLIMS-P system [26] extracts information on protein phosphorylation by looking for enzymes, substrates, and sites that can be assigned to ‘agent’, ‘theme’, and ‘site’ roles of phosphorylation relations. The

approach involves applying a part-of-speech tagger, which was originally trained on newswire material and minimally adapted, followed by phrase chunking, semantic typing of chunks, and then identification of relations using pattern-matching rules. Phrase chunking delivers more than basic noun phrase chunking. It is able to combine verbal elements into a verbal group, and handle aspects of coordination and apposition. The results then help simplify the following pattern matching, since fewer and more general rules can be written than otherwise would be needed.

Semantic typing of noun phrases is then carried out, to assign appropriately to types for protein, substrate, amino acid residue, or source (cell, tissue). A combination of clue words, suffixes, other phrases in the immediate context (e.g., appositives), acronyms, and so forth, aid semantic typing, with no reference to any ontology.

Sentences containing the semantically typed elements are then targeted by a number of pattern matching rules, which attempt to assign appropriate elements to 'agent', 'theme,' and 'site' variables. One set of patterns targets sentences containing a verbal form of *phosphorylate*; a second set targets sentences with the nominalized form (*phosphorylation*). The rules allow for optionality of pattern arguments. The RLIMS-P system was evaluated on a task to support evidence tagging over MEDLINE abstracts for curation purposes, and gave good results, with 88% recall and 98% precision. Instances of the various patterns are extracted, but no complex structured representations are produced. The system has a narrow focus, which is exploited to good advantage, and does not attempt to extract information beyond finding plausible matches for patterns involving 'agent', 'theme', and 'site.' Although appositives are identified during chunking, they are only used to increase reliability of identification of the enzymes, substrates, and sites that are the focus of semantic typing. They are not used to provide further information about attributes of such entities. For example, where an effect of phosphorylation is mentioned, this would not be noted. It is not stated whether the system can handle negation. Currently, there is no mechanism to handle cases where 'agent', 'theme', and 'site' information is distributed over more than one sentence, although work on this was planned. Others have found that coreference mechanisms and domain models are required to handle such phenomena [27–29].

Many systems in this family typically do not attempt to produce abstract structured representations, but rather attempt to extract phrases or entire sentences containing matched material. This restricts their usefulness for later mining. End users need to appreciate not only how such systems might fit their needs, but also the variety of other approaches available in the information extraction spectrum.

7.3.1.1 Dictionary-Based Information Extraction

A variant of the pattern matching systems is represented by Martin et al. [30], who rely, for extraction of protein-protein interactions, on a number of dictionaries containing: protein names and their synonyms; protein interaction verbs and their synonyms (e.g., *interaction* is taken as a synonym for *interacts*); and common strings used to help in the identification of unknown proteins (e.g., *protein*, *kinase*). In addition, some 80 patterns were written to match instances of interactions. From the single example given in (7.4), these patterns appear to rely on instantiating variables typed to represent equivalence classes drawn from the dictionaries.

($VarGene $Verb (the)? $VarGene) (7.4)

Intervening material can be specified as regular expressions. In (7.4), an optional *the* can occur between an interaction verb and a following gene. Presumably, reference also can be made to items in the dictionary of common protein indicator strings. Although the power of this formalism cannot be judged from one example, the authors note that "many interactions were not detected because of the complexity of the way they were formulated by authors" in texts. This would appear to indicate that the authors' approach is not employing any form of partial parsing or chunking, although the formalism may be capable of supporting this.

An important characteristic distinguishes the system reported on from the sentence extractors described above. Rather than producing the matching sentence, a template is output, whose slots are filled with the relevant dictionary item, or the canonical dictionary item chosen to represent a group of synonymous items. These templates then can be subjected to data mining phases, if required.

7.3.1.2 Preposition-Based Parsing

Another variant of basic pattern matching, due to Leroy et al., is based on a notion of preposition-based parsing [31, 32]. This technique is used to fill basic templates with material surrounding prepositions in biomedical texts, including information on negation. The motivation behind this approach is that prepositions, such as *by* and *of*, are often indicative of agent or patient roles, as in the examples in (7.5).

a. *inhibition of cell proliferation*—cell proliferation (patient) is being inhibited (action).
b. *apoptosis induced by the p53 tumor suppressor*—the p53 tumor suppressor (agent), is inducing (action) apoptosis (patient). (7.5)

The parsing technique used here involves no syntactic analysis, but rather relies on finding specific types of elements that can be taken to indicate the boundaries of nominal or verbal groups. Examples of these elements are punctuation, function (noncontent-bearing) words, and so forth. A claimed advantage of this technique is that it allows for more general extraction, rather than an extraction of specific items from a prespecified vocabulary. Following the initial extraction phase, further straightforward processing takes place to combine templates, yielding new, more informative, templates. Combining also is carried out where instances of two templates are connected by a conjunction (*and*, *or*), with the condition that the same preposition occurs in each template. Cascaded finite state automata are used to guide acceptance and extraction of template fillers. Transitions between states are authorized, as usual, only if appropriate input is encountered, but also can involve a further maximum distance criterion. That is, the desired input is permitted to be found within a specific window, whose size is variable, based on the individual transition. This gives flexibility in matching.

Initial results [31] revealed that almost 30% of errors in processing were due to taking in too much material for an agent or patient. The process to establish the boundaries of nominal groups was allowing incorporation of material that should be excluded. A noun phrase chunker was then adopted [32], although it was modified to not return prepositional phrases as whole entities, since prepositions lie at the heart of the applied strategy. Where nominalizations previously had been mapped to infinitival forms when extracted, this strategy was changed, so that if a verb such as *suggest* or *speculate* was present in the sentence, then mapping of nominalizations to infinitival form was not done, since the sentence was not assumed to involve a fact. This and other improvements led to an increase in precision.

In this work, no attempt was made to map template fillers to any semantic or conceptual entity. However, further developments have been reported in [33], which include mapping template fillers to GO, HUGO, and the UMLS metathesaurus. As might be expected, results of this mapping are not good, due to the issues discussed in Chapter 4. In common with the majority of other pattern matching approaches discussed above, there is a focus on extracting strings. That is, the fillers of template slots are essentially strings, or canonical forms of strings, found in the text. In [33], there is no sophisticated named entity or term extraction component, which can appeal to context to tag entities with appropriate semantic or conceptual tags. There is no basis on which to assign a semantic or conceptual entity to strings during the main processing phase. Therefore, the only means available to assign a semantic or conceptual label is essentially the look-up of resources at the end of the main phase.

7.3.2 Basic Context Free Grammar Approaches

As there was much earlier in the history of NLP, attempts to process biotexts have been made, applying parsers based on straightforward context free grammars with little consideration of linguistic constraints. This can be considered an improvement from basic pattern matching. Temkin and Gilder [34] adopt compiler generator tools to produce a lexical analyzer and a parser customized to extract interactions involving genes, proteins, and small molecules. The parser operates in top-down mode, in which hypotheses are made about the structure of the sentence before seeing the words. This is in contrast to many rule-based IE systems, which typically apply a bottom-up strategy or a mixed mode. In bottom-up processing, the evidence of words in the sentence is used to guide structure building. A popular mixed mode is bottom-up processing combined with top-down filtering. This is more commonly seen with full parsers, which we discuss in the next section, since a top-down mode is of little relevance to partial parsers.

Whereas top-down processing is appropriate for languages with little or no syntactic ambiguity, and with many similar types of sentences (e.g., artificial programming languages), it is not generally used as the sole strategy for processing natural language. Temkin and Gilder's system [34] relies on a small number of interaction keywords, grouped by semantic category, although semantic categories do not appear in their sample outputs. Since the grammar is simple, problems are reported with long, complex sentences, characterized as "unstructured text representations." However, it is precisely the job of IE to be able to deal with such sentences, especially where they conform to common writing practice in the domain under study. This attempt demonstrates the limitations of simple context-free grammar-based approaches, using few linguistic constraints, in conjunction with a top-down processing strategy.

7.3.3 Full Parsing Approaches

Earlier attempts at IE using full parsing showed no improvement in accuracy in the MUC competitions [35, 36]. However, there have been recent attempts to apply full parsing to biotexts. In contrast to Temkin and Gilder's approach [34], full parsing here implies reference to a theory of syntax. A challenge for syntactic theory-based full parsers is to deal with the special language of domain texts. Syntactic theories tend to be constructed with reference to the general language. Available computational grammars are often more suited to general language processing, and it is not evident that they can be easily specialized to handle domain texts. Nonrobust performance is immediately an issue for such parsers.

There has been a worthwhile attempt [37] to apply Link Grammar [38] to the extraction of protein-protein interactions. This work is significant, in that a thorough evaluation was undertaken, with detailed analysis of causes of failure.

A general English grammar was adapted to account for biomedical sublanguage phenomena. The Link Grammar parser finds all possible analyses (linkages) according to its grammar. It was found that postprocessing was required to apply other constraints. Postprocessing is a computationally expensive process, given the number of linkages for ambiguous sentences found by the parser. The number of analyses was reduced by random sampling, and by applying heuristics (based on general English) to yield the best parses first. Since insistence on achieving a full parse would cause failure due to the variety of sentence types, processing constraints were relaxed, to allow the parser to produce partial analyses if a full analysis could not be found. Moreover, a pragmatically ascertained time limit was applied, to ensure that restricted parses are still produced in cases where the parser was engaged in discovering a large number of analyses for a sentence. Only 7% of sentences in the test set had a fully correct analysis in the first analysis, while 28% had a fully correct analysis somewhere in the parse results. A maximum number of 10,000 results was permitted per sentence. No titles were properly analyzed, since the titles often lack verbs, which are critical to successful link grammar analysis. An upper limit of 60% of protein interactions could be extracted. Whenever a time limit was invoked, interactions were unlikely to be found. Time-limited parsing was found to be a significant cause of error whenever it was invoked.

Errors due to "ungrammatical sentences" ran at 8%. The authors comment on the apparent level of mistakes in their PubMed corpus due to nonnative writers of English, and cite phenomena such as missing possessive markers and determiners, and errors in number agreement. They note that grammar constraints could be relaxed to deal with these errors, but that this would lead to more ambiguity, and more difficulty in identifying correct analyses. Coordination of compound noun modifiers also was found to be a problematic phenomenon, requiring modification of the grammar. Unknown grammatical structures accounted for 34% of errors, solvable by local modifications to the grammar. Strategies for handling unknown words were responsible for 16% of errors. Some of these would require modification of the parser rather than of the grammar.

Failure to recognize domain-specific named entities was responsible for 28% of parsing errors. An experiment was conducted to import more than 125,000 words from the UMLS Specialist Lexicon, but this led to an overall improvement in parsing of only 2.5%, although it reduced the need to invoke a time limit. The authors concluded that such modest improvement does not indicate that it is hardly worthwhile to add lexical entries, but rather that the improvements rely on modifying the grammar or the parser, not the dictionary.

What lessons can we take from the authors' valuable detailed evaluation of this attempt? First, not all parsing strategies or grammatical frameworks are directly suitable for processing texts in our domain of interest. Careful study

must be made to select, develop, or modify an information extraction engine and a grammatical approach that are appropriate for texts in the domain at issue.

Second, we must be wary of characterizing the language of (largely) well-edited material as "ungrammatical," because articles or abstracts may be written by apparently nonnative speakers of English. This point is not pursued in any depth in [37]. However, one may note that a sublanguage often demonstrates deviations from the general language, but that these "deviations" are not seen as such from within the sublanguage—they are seen simply as natural sublanguage behavior [39]. They would trip up a general language analyzer. If we wish to apply an analyzer to sublanguage texts, then we should ensure that it is able to handle sublanguage phenomena. The ungrammatical sentences reported may be truly ungrammatical, even with respect to the sublanguage of biomedicine. However, regardless of their nature in this particular corpus, the lesson really is that whenever we approach a corpus of text, we need to take into account any possible sublanguage characteristics. Much is written about processing of terminology in biomedicine, although there is an almost exclusive focus on nominal terms, with an unhelpful blurring of the notions of ontology, documentation thesaurus, and terminology. The effects of this unfortunate blurring can be best appreciated in the numerous failed attempts to treat concept labels of ontologies and descriptors of documentation thesauri as term forms for text string look-up. However, a sublanguage consists of more than nominal terms, and indeed more than a lexical component. It can display special behavior, with respect to the general language, at many linguistic levels, covering many linguistic and textual phenomena. In particular, sublanguage verbs take on crucial importance, as we shall see in Section 7.3.6.

Third, a lesson to be taken is that ambiguity is the bane of parsing natural language, and any means of reducing ambiguity must be explored. This was one reason why partial parsing became the preferred mode of processing for rule-based IE systems—partial parsers do not attempt to discover every possible analysis. For a full parser, many ambiguities arise from the multitude of ways it can bring constituents together to build representations of even a modestly complex sentence.

Fourth, parsing of fragmentary phrases (e.g., titles, headings, captions, text in table cells, and verbless fragments) can pose problems. We note further that this is not confined to full parsers, and partial parsers can have problems with such phenomena. This is a well-known problem in IE [7], and has not, as yet, met with workable solutions.

Link Grammar also has been used in the CADERIGE project [40]. Here, it is explicitly used in a shallow parsing mode to extract specific grammatical relations.

Yakushiji et al. [41] apply a full parser with a general purpose HPSG grammar [42] and lexicon, with a view to extracting events from biomedical

texts. However, partial parsing is applied before the main parsing process to overcome some of the drawbacks of full parsing. The partial parser used [43] attempts to resolve local ambiguities that would otherwise persist from part-of-speech tagging, by excluding unlikely part-of-speech tags from further analysis, using an analysis of local context. The full parser delivers general deep syntactic argument structures to be mapped to domain-specific event frames, although this latter step was not yet available in the reported work. No precision details are given, but recall for full parsing is given as 47%. Among the problems discussed is the treatment of modifiers. Since a general language grammar was used, based on general verbal subcategorization criteria, this proved to be a severe problem. Many constituents taken as modifiers in general language should be taken more properly as arguments in sublanguage texts. The effectiveness of the approach described in [41] cannot be precisely characterized, especially since mapping to event frames was not undertaken. Taken at face value, it does not offer a good solution to the processing of domain-specific texts.

Kim and Park's BioIE system [44] (to be distinguished from the system of the same name by Divoli and Attwood [21]) also undertakes full parsing, including every sentence containing instances of predefined patterns involving certain keywords. This system is based on Combinatory Categorical Grammar (CCG) [45], and is used to extract general biological interaction information, which is then annotated with GO concepts. CCGs belong to the class of mildly context-sensitive grammars. The CCG formalism employs function combination and function composition to combine consecutive phrases into larger phrasal units. It can handle complex linguistic phenomena, such as long distance dependency and coordination, and has been shown to handle relatively free word order.

However, BioIE's CCG parser is not deployed in the initial stages of syntactic analysis. First, some 1,300 patterns are applied to find instances of patterns involving a number of specific keywords, which are drawn from lists used in other studies [23, 46]. Keywords indicate either basic biological interactions or interactions between basic interactions. The patterns are intended to find potential arguments of the interaction keywords (mainly verbs or nominalizations), and are regular expressions over parts-of-speech and morphological features. Heuristics are used to limit the search to the left and right of a keyword for potential arguments. A CCG parser is applied only after this bidirectional pattern matching step, in order to determine whether the candidate arguments are indeed functioning as required, syntactically, and whether a sentential parse can be built. Subsequently, validated arguments are mapped to GO concepts [44]. Coordination proved a problem for parsing, as did missing verbal subcategorization information in the earlier version of the system [46]. The verbal subcategorization problem was rectified to some extent [44].

The authors claim that their system is retargetable, and that less detailed linguistic information is needed compared to GENIES [47]. However, they also note that the interaction keyword patterns, which do the bulk of the work in the system, are "not so simple" and "not . . . so straightforward to generalize." This would then argue in favor of a more linguistic approach of the type they criticize, which does have the advantage of allowing more significant generalizations to be made at a deeper level than is achievable through the type of patterns they advocate. While precision values are apparently roughly similar for BioIE and GENIES, GENIES has better recall. This would indicate possible failures with the CCG-based approach to achieve a sentential parse, and an advantage for the fallback strategy of GENIES in the face of potential parse failure. No direct information is given in [44], but an example of processing stages shows that parsing does aim to achieve an overall sentential parse. Poorer recall also is a characteristic of systems that are unable to deal with the variety of surface patterns found in texts, whereas the hybrid syntax-semantic sublanguage approach adopted by GENIES offers less sensitivity to surface characteristics.

We further note that BioIE's original syntactic parsing performance was improved only after addition of subcategorization information on the verbal interaction keywords. This seems to have been done on an ad hoc basis as a result of error analysis, rather than on the basis of sublanguage analysis leading to motivated subcategorization frames. If so, then this leaves open the question whether or not the number and type of arguments used in BioIE subcategorization frames are more indicative of general language than of special language.

Another example of full parsing is provided by Daraselia et al. [48], who employ a full sentence parser and a domain-specific filter to extract information on protein-protein interactions. The parser is described in detail in [49]. A two-step approach is employed. First, all possible syntactic analyses are discovered, using a unification-based context free grammar and a variant of Lexical-Functional Grammar [50]. Second, each alternative parse is mapped to its corresponding semantic representation. Many tens of thousands of syntactic parses can be produced per sentence. The output of the semantic component for each sentence is a set of semantic trees, in which lexemes are linked by relations indicating either thematic roles (e.g., 'agent' or 'patient') or attributive roles. A custom-built, frame-based ontology is used, since no other ontology was found to be directly applicable. The results of syntax-to-semantics mapping, still potentially many thousands of representations per sentence, are filtered against the ontology to yield a frame tree, in which ontological frames are instantiated and linked according to the constraints expressed in the ontology. A preference mechanism is used to help control construction of the frame tree, which takes into account the different status of thematic and attributive roles. That is, if an attributive role cannot be incorporated in the current frame tree according to

frame constraints, then it is ignored as being less important to the core meaning of the sentence, whereas failure to incorporate a thematic role will lead to rejection of the current interpretation.

As a result of this preferential processing, which may lead to the lack of consideration of less important structures, the same frame tree may be produced from numerous different semantic representations for the same sentence. An additional filter is applied to ensure that duplicates are excised. A final processing phase converts frame trees to conceptual graphs, which then can be subjected to querying, or used as a basis for mining.

Evaluation of extracted protein-protein interactions revealed that, although precision was high, recall was very low (21%). This was found to be due to the low coverage (34%) of the parser. In addition to the authors' comments on performance, we may note that this system, like many classic full sentence analysis systems, suffers from having to deal with potentially many tens of thousands of analyses. The various filtering techniques that are employed reduce that number progressively. Novichkova et al. [49] report that the single correct ontological interpretation is retained, even when more than 100,000 syntactic structures have been produced for a sentence, which normally would be surprising. Presumably, competing final representations are not found in the results, because "inessential" components of semantic trees are ignored, according to the preference mechanism. These "inessential" components often are a major cause of ambiguity for general analyzers.

However, a question arises as to the exhaustiveness and specificity of extracted information, since the distinction between thematic and attributive roles is not well articulated. For example, it is not always the case that a prepositional phrase, which is treated as an inessential attributive role in this work, has in fact that role. It may be playing a thematic role beyond the "normal" range of thematic roles envisaged by treatments based on general language notions of argument structure. In other words, when deciding how to treat items such as prepositional phrases, care must be taken to characterize their role in relation to the verb in the sublanguage under study, otherwise important information will be missed. Almost 30% of parsing failures were due to grammar-related issues, including problems with analysis of relative phrases and "unknown domain-specific noun phrase constructs." There is little detail given on how syntactic arguments of verbs and roles of semantic frames are described, or on how such descriptions are determined. However, we note that a "generally accepted set of roles" is used, and that there is a key distinction made between thematic and attributive roles. Taken together, these would then indicate that an approach derived from techniques more applicable to general language has been adopted.

It is undeniable that the authors have attempted to deal with domain-specific phenomena, and have gone some way towards achieving that aim. However, inherent in their approach is an apparently unquestioned belief that

attributive elements are of lesser importance. This is in contrast to the investigation of attributive elements by Leroy and Chen [33], and to the strong sublanguage approach of Friedman et al. [51], for example.

We note in passing that augmented transition networks, used in this work to encode grammatical descriptions, are today largely deprecated for natural language processing. They merit hardly a mention in the current standard textbook for the field [5], and were taken severely to task by Johnson [52].

7.3.4 Probability-Based Parsing

As we have seen, full sentence parsing is problematic when faced with the variety of language behavior, which has led to attempts to reduce ambiguity through partial parsing or chunking prior to the launching of the full sentence parser, or to heuristically limiting the number of parses.

Probability-based parsing offers a different solution, which essentially assigns weights, derived from large-scale corpus processing, to grammar rules. Such parsers offer robust processing, and, where based on dependency grammars, less sensitivity to surface variation. Rinaldi et al. [53] report on preliminary work involving a dependency-based probabilistic parser to extract relations involving genes and proteins from the GENIA corpus. Partial but still useful results are returned if no parse can be achieved. In the FlySlip project at the University of Cambridge, a mature probabilistic parser, RASP [54], is being applied to extract relations from biotexts for FlyBase. This project was recently started at the time of writing, so we look forward to learning of its results.

7.3.5 Mixed Syntax-Semantics Approaches

In a classic modular system for IE, each module handles one level of linguistic analysis. Although this organization has its advantages, it also has its drawbacks. For example, in a strictly compartmentalized system, we lose the ability to profit from different kinds of knowledge at the same time. Such considerations have prompted investigation of approaches that attempt to exploit syntactic and semantic knowledge together.

One such mixed syntax-semantics approach is adopted by McDonald et al. [55]. A part-of-speech tagger was trained to apply a tag set, containing both part-of-speech and semantic word tags. This essentially combines part-of-speech tagging with (limited) NE recognition. The motivation behind this step is to remove reliance on consulting a lexicon to obtain semantic constraints, and to enable relation extraction to be more generally applicable, through not being tied to a small number of prespecified templates involving specific verbs. However, a noted feature of this approach is that accurate postfiltering is required to reject nonrelevant relations. Basic phrase chunking is then applied to construct

noun and verb groups. A cascade of four finite-state automata then apply to construct partial parses. These automata implement approximately 1,800 grammar rules, whose application is constrained by limited reference to left and right context, including punctuation. The rules combine sequences of categories to form noun phrases, verb phrases, prepositional phrases, relative clauses, and subordinate clauses. The examples of rules cited indicate that prepositional phrases are given separate categories according to the preposition involved, which would account in part for the high number of rules.

Relation identification is then carried out, using a similar kind of rule formalism to match "knowledge patterns" (i.e., sequences of syntactic categories). A relation consists of only three roles, "loosely comparable to subject, verb, object constructs." Instead of building a single category, a knowledge pattern rule at this point builds as many relations as are specified by its left-hand side, and, for each relation, assigns categories on the right-hand side of the rule to roles (taken from a set of 10 roles). Thus, for some sequence of categories, a number of potentially overlapping relations may be found. Some 200 knowledge pattern rules were written.

Finally, each relation has to pass through a "semantic filter" before extraction is complete. The filter checks that at least one word from the tokens comprising the predicate role fillers does exist in a resource such as GO or HUGO, and that at least one word making up the predicate appears in a list of some 150 verb stems, specified by a domain expert. The precision of extracting pathway relations was 61%, and recall was 35%. Similar to Leroy and Chen [33], it was found that a major source of failure were late-stage attempts to map tokens of relation predicates to concept labels in ontologies. Approximately one-third of failures were due to incomplete extraction rules, and to a lack of treatment of coreference.

Although McDonald et al. claim to be exploiting a hybrid syntactic-semantic approach in their tagging, it is very hard to judge from the reported work the nature of the semantic tags, since only a very limited sample is given. Moreover, none of the sample rules given refers to semantic categories, and the sample output of parsing and relation identification refers only to syntactic categories. Therefore, it is difficult to judge the degree of any potential contribution of semantic-based processing. The examples given would lead to the conclusion that relation identification (i.e., before validation via the final semantic filter) was wholly concerned with establishing syntactically-based triples, and that there was indeed little true semantic knowledge in this system. Our impression may be due simply to injudicious exemplification in the reported work.

From a grammar writing point of view, we may note that it is unusual to see a rule producing more than one output (knowledge pattern rules). This can lead to lack of transparency, affect ease of maintenance, and be a source of overgeneration. Such an approach is highly indicative of a lack of exploitable

semantic features that would otherwise help constrain the analysis. As a result of the overall strategy chosen, which eschews exploitation of constraints such as subcategorization frames, rather flat syntactic representations are produced. This leads to long sequences of syntactic categories being subjected to knowledge patterns, which then derive numerous, potentially overlapping relations from these, which must then be validated. Patterns applied over long sequences of categories further indicate a rather surface-oriented treatment of linguistic phenomena. The express intention of this work is to achieve generality, and to do away with the need to express a lot of information about constraints. However, this work gives up much in doing so. In particular, it gives up the possibility of being able to tell us how relevant subparts of sentences relate to each other. That is, numerous overlapping relations can be produced, but there is no subsequent step to tie any together. This is not necessary to carry out a full semantic analysis, but is required to account for the embedding of semantic constituents and their relation to the main predicate. This analysis apparently does not go much beyond simple subject-verb-object relations, and does not attempt to determine how semantic relations or other constituents can be related semantically to the main predicate, in order to yield a meaningful analysis of the sublanguage message.

A more complex mixed syntax-semantics approach is due to et al. [29]. This work concentrates on extracting relations concerning gene regulation by using syntactic-semantic rules, encoded as a cascade of finite state transducers for application by a partial parser [56]. Domain-relevant nouns and names and relevant verbs (of activation, repression, regulation, coding, or containing) are semantically tagged. Noun phrases and noun chunks are built incrementally, and, in certain instances, are labelled as being potential semantic agents or themes via inspection of nearby prepositions. For example, the semantic role of *binding site* is ambiguous, but can be resolved by noting whether *of* or *for* follows the phrase. Compare this approach to that of Leroy and Chen [33]. Next, three types of relation are extracted—up-regulation, down-regulation, and unspecified regulation of expression. Verbal syntactic and semantic subcategorization information is exploited together to constrain the analysis, although no details are given. Relation chunking can handle passive, as well as active, constructions. Relations involving nominalizations (e.g., *binding of . . .*) also are extracted. There is no treatment of anaphoric reference. Precision of approximately 83% was reported, but recall was low, at 20%. The authors comment that their rule-based approach would benefit from being ontology-driven, a point we address in the penultimate section.Before that, we go into further detail on the sublanguage approach to IE.

7.3.6 Sublanguage-Driven Information Extraction

Sublanguage-based processing has a long history in natural language processing. A good introduction to the concerns, theory, and applications of sublanguage, with emphasis on biomedicine-related applications and pointers to the sublanguage literature, is provided by [57]. The central notion of sublanguage is that the language of a special community (e.g., biologists) displays a particular set of constraints, with respect to the general language. These constraints operate at all levels. This gives rise to vocabulary items with special meaning (terms), to specific ways of composing terms (term formation rules), and, importantly here, to special ways of structuring sentences (sublanguage syntax) to communicate special information (sublanguage semantics). Some of these constraints will cause the general linguist who is unaware of sublanguages to complain of ill-formed sentences or deviant usage of language.

However, such deviance is being seen from the perspective of general language. To the sublanguage practitioner and the domain expert, there is no deviance, only normal ways of communicating in that domain. For example, one may find that the range of tenses is restricted; that certain derivational mechanisms are prevalent in the sublanguage although rare in general language; that other mechanisms common to general language are hardly used; and that terminological verbs take on specific syntactic-semantic behavior. Consider the examples in (7.6).

a. The patient presented with influenza to the doctor.
b. The patient presented the doctor with influenza. (7.6)

The sentence of (7.6a) belongs to medical sublanguage, since, in that sublanguage, *present* demands a particular sentence structure to render the desired sublanguage meaning, whereas the sentence of (7.6b) belongs to the general language.

Such constraints give rise to what Harris [39] calls the "informational structure" of the sublanguage under study. Much of this structure is realized through relatively fixed phraseology (compared to general language), and often these canonical ways of putting words together are conditioned by the demands of the sublanguage verb. Scientific sublanguages display highly conditioned canonical structure, while the sublanguages of the social sciences do so to a lesser degree. Thus, as GENIES and other sublanguage-based systems demonstrate, we can exploit knowledge of sublanguage structure to achieve more accurate and informative IE compared to approaches based on general language.

As we saw, Leroy and Chen [33] demonstrate that prepositions are useful indicators of important information in biotext. Daraselia et al. [48] distinguish thematic roles from attributive roles, with the latter containing prepositional

phrases Their preference mechanism is tuned to ignore attributive roles, where these might lead to analysis failure. Here, then, are two contrasting views of how to treat prepositional phrases in biotexts. We also noted that the performance of the BioIE system of Kim and Park [44] was only improved after addition of basic subcategorization information, of the type required for analysis of general language. None of these systems is taking any strongly motivated sublanguage approach. Daraselia et al. apparently use roles related to general language. Kim and Park were forced to adapt their system, but only to handle general language subcategorization of a basic kind. Leroy and Chen recognize that prepositional phrases play important roles in biotext but the authors do not integrate their description, which remains relatively surface-oriented, in any kind of overall model accounting for the structure of scientific messages.

Why are we making so much of the status of prepositional phrases? If one recognizes that the syntactic or semantic arguments of verbs play a role in constraining and guiding analysis, then the issue is the quantity and type of arguments associated with the verbs. The answer is apparently straightforward: as many arguments as are required. However, this straightforward answer is not one that sits easily with any approach based on linguistic descriptions intended for processing general language. In general language, the number and type of verbal arguments is quite restricted. This is because the informational structure of general language is much looser than that for scientific sublanguages. In general language, we add attributive elements as needed, and in many cases these are signaled by prepositional phrases. Their omission or addition does not radically alter the core meaning expressed by the verb and the fillers of its few thematic roles.

However, in the sublanguage world, verbs have special status. They have particular requirements for subcategorization frames. The average number of arguments in sublanguage subcategorization frames has been found to be much higher, compared to the number traditionally recognized for general language verbs. In work carried out on sublanguage-based machine translation [58, 59], it was found that up to 15 arguments were needed for verbs in the domain of satellite telecommunications. Many of these arguments are signaled by prepositional phrases, often manifesting themselves in relatively rigorous orderings. Omission of such prepositional phrases would detract from the overall meaning imparted by the sentence, and in many cases would render the sentence meaningless. We note that the UMLS Specialist Lexicon allows for only five types of verbal frame, which is inadequate for the description of terminological verbs having higher numbers of frame arguments.

Thus, sublanguage studies have shown that there is much to be gained from capitalizing on the informational structure of special language texts. Good recall and good precision result from efforts to describe the behavior of sublanguage verbs and of sublanguage nominalizations, which are derived from

verbs, in terms of the frequently large number of mandatory or optional arguments they can take, compared to general language counterparts. The GENIES approach has therefore much to recommend it.

GENIES takes a strong sublanguage approach to extraction of biomolecular interactions relevant to signal transduction and biochemical pathways. As is typical of sublanguage approaches, hybrid syntactic-semantic rules are used. This reinforces the previous discussion that the relatively heavily syntactified expressions in sublanguage texts give strong indications of sublanguage semantic roles. Thus, syntactic and semantic constraints can be referred to in the one rule. As Friedman et al. [51] note: "In the biomolecular domain, the primary information concerns descriptions of biomolecular pathways consisting of complex interactions and other relations [. . .] Since a pathway itself is complex and consists of sequences of interactions, the language expresses the sequences using complex and highly nested relations. Thus, an argument of an interaction can be another interaction and so forth." Such complexity is daunting for many of the other systems, which typically manage to recover only part of the information in a complex message. However, the sublanguage approach of GENIES is able to cope with such complexity, through application of the constraints specified on sublanguage verbs to extract representations of complex nested chains of interactions. These representations consist of (possibly embedded) frames. For example, a frame expresses a relation between a semantic action and the fillers of its arguments, or between an object and its properties (possibly represented as states).

A small number of semantic categories relevant to the biomolecular domain is used, including a number of subcategories. To aid specification of these categories, a domain-specific ontology was developed [60], covering both entities and events.

Although GENIES employs a full parsing strategy by default, it is more robust than other full parsing systems, for two reasons. First, much ambiguity is excluded, due to sublanguage constraints imposed by the grammar and lexicon. There are fewer opportunities to go astray in analysis. Second, if a full parse cannot be achieved, then motivated partial parsing is invoked, which ensures that at least a good level of recall is maintained. This dual technique has proven its worth in comparison to, for example, a "panic mode" [37] invoked after some time limit has expired without finding a parse. Evaluation of GENIES yielded 96% precision and 63% recall.

The main disadvantage of the strong sublanguage approach that is often cited is the need for laborious efforts to retarget to another domain. However, in terms of precision, recall, and ability to produce output representations that can be exploited by other components, there are distinct advantages. Friedman et al. [51] concede the difficulty of establishing sublanguage grammars and lexicons, but explain how such effort could be reduced to manageable proportions.

As an aside, it is appropriate for the reader to reflect on the following: (1) biology is hard, with no easy solutions; and (2) processing natural language is equally challenging, demanding much effort to codify language behavior in sufficient depth, and to a sufficient degree of accuracy, to permit the kind of analyses required by users of information systems. Simple techniques can get us only a little way, and sophisticated techniques demand concomitantly sophisticated resources in the form of grammars, lexicons, and ontologies. These are expensive to develop and maintain. In recent years, there have been numerous advances that reduce the cost of configuring and tuning sublanguage-based systems: lexical standards that enable reusability; machine learning techniques that discover patterns of sublanguage behavior in large annotated text corpora to help grammar writers; development of ontologies that can act as domain models; major developments that aid in extracting and characterizing terminology, including compound terms and acronyms; massive amounts of freely available text; and so forth. It also is no accident that it is in areas such as biomedicine and the biosciences in general that we are seeing sublanguage systems such as GENIES and GeneWays [61] being built. These domains are particularly exercised by the need to construct ontologies and terminologies, which are prerequisites for sophisticated sublanguage-based processing.

Not only are such resources required, but they also must contain appropriate information to support language analysis. Terminologies typically do not contain the linguistic information required (e.g., terminological verbal behavior), and ontologies may give insufficient information on event or action type concepts, particularly where events involve a relation over numerous concepts, including over other events. There is an additional need to tie lexical (including terminological) descriptions to ontological descriptions, particularly where verbs are concerned. Friedman et al. [51] note that fine-grained lexical descriptions are required for sublanguage verbal behavior, and describe how the semantic categories used in these descriptions (e.g., the semantic categories that verbs subcategorize for, in verb frames) are mapped to the more general categories found in ontologies.

One major advantage of building a linguistically sophisticated terminological lexicon is that (assuming lexical standards are judiciously employed) it can be reused to support many other tasks besides information extraction. It would appear timely to engage in community efforts to build such a resource.

7.3.7 Ontology-Driven Information Extraction

The classification of approaches we have developed in this chapter can be viewed as having the following questions: the amount of knowledge required to analyze and extract, how that knowledge is obtained, how it is represented, and how it is manipulated. We have seen that limited results can be expected from systems

that use little linguistic or conceptual knowledge. We also have seen that linguistically-based systems that do not fully take account of sublanguage behavior meet with difficulties. We now turn to consider how ontologies can be used to aid extraction of relations and events.

Up until recently, most rule-based IE systems have used neither a sophisticated linguistic lexicon, nor an ontology of entities, nor by implication, an ontology of events. They have relied on gazetteers, which are essentially flat lists giving a mapping between a look-up string and a tactically useful semantic category. In MUC, as we have seen, there was a limited number of such semantic categories. Rule bases tended to be small. Any large rule bases typically had many rules handling the same type of phenomenon, since it is easier to write numerous rule variations than it is to think up only a few complex rules that may require a greater degree of sophistication in terms of linguistic processing capability. For a small number of events, event rules could directly encode (hardwire) the information needed to control role assignment.

However, it has become apparent that the traditional gazetteer-based approach is not so well suited to IE in the biosciences and biomedicine, especially with the shift to extract information from full scientific texts rather than abstracts. As discussed in Chapter 4, terminology plays a crucial part in characterizing knowledge in scientific text. This places a heavy demand on an IE system, since it must be able to handle terminology and the concepts to which the terms refer. Users want to find facts of many types in texts. Moreover, the same user's requirements will typically change along with the topic under study. This places a further burden on a system, especially one deployed as a service, to be able to handle many kinds of facts as they occur in all their variants in full text, just in the one domain. Setting aside terminological concerns for the present, we look now at IE systems that attempt to use ontologies to better respond to user needs.

We here distinguish ontology-based processing from ontology-driven processing. Ontology-based systems include those that attempt to map a discovered entity, often expressed as a simple string, to a concept in an ontology. This mapping usually occurs late in the analysis. The ontology is essentially used in a passive way, to provide a concept through simple look-up. We have already discussed several such systems.

In contrast, ontology-driven systems make heavy, active use of the ontology in processing, to strongly guide and constrain analysis. A useful side effect of this approach is that the number of rules required can be substantially smaller, compared to the number required in an ontology-based system or a system that does not consult an ontology at all. Without an ontology of events, filling event templates by rule can involve a large set of rules for every template type. Thus, if we can describe template structures in an ontology, independently of the pattern-matching rules, then there is much to be gained [62].

As we have seen above, GENIES makes use of an ontology in a slightly different way. The emphasis in this system is on leveraging the ontology to help in the specification of the rich sublanguage lexicon and the sublanguage grammars, which is why we placed GENIES in a class of its own. We now look at two other ontology-driven systems that do not deploy a strong sublanguage approach.

The PASTA system [28] extracts information from MEDLINE abstracts on the roles of specific amino acid residues in protein molecules. It is based on a pipeline of processes, similar to MUC-type processing, and carries out template element and template relation extraction, but not scenario template extraction (events having been found to be not relevant to the topics of interest). Due to the restricted topics under consideration, only three template elements and two binary template relations were identified. The template elements covered 'residue', 'protein', and 'species'; the template relations were 'in_protein' (between residue and protein), and 'in_species' (between protein and species). In a 'residue' template element, there are slots for name, residue number in sequence, site and function (in a combined slot), secondary and quaternary structural arrangements, region in which residue is found, and interaction (the latter holding references to various types of atomic contact). Template relations contain references to the 'protein' or 'species' template elements forming their arguments. Templates for 'protein' and 'species' store a single attribute for protein and species name, respectively.

Regarding processing, 12 term classes were used in preparsing processes to label terms of interest. Bottom-up chart-based phrasal parsing (based on a general grammar) was then entered, whose output is mapped to logical (semantic) predicate-argument structures. However, these structures are not used to fill templates directly. This would lead to many partially filled 'residue' templates, since information that could fill such templates is typically not all contained in the one sentence. A discourse processing module takes the predicate-argument structures as input, and carries out inferencing, based on a limited domain model (ontology). Each predicate-argument structure is consecutively integrated in the domain model, also under the control of coreference resolution, which merges new instances of representations with existing ones, where possible. Inference and coreference are further used, to help fill in information only implicit in the text. Finally, a template-writing module filters instances of representations from the domain model that are relevant for the PASTA templates.

PASTA yielded 66% precision and 75% recall for filling template elements, and 65% precision and 68% recall for filling template relations, with an overall combined precision of 65% and recall of 68%, using the somewhat strict MUC scoring software. Since these are apparently good results, we examined the detailed evaluation scores given by the authors. From these, we note that an 'article' template element task was included in the evaluation, which extracted information on title, author, and source, for a MEDLINE reference. Precision

and recall for such extraction were both 99%, as might be expected for this straightforward task. If we remove from consideration the 'article' template element task, then the recall and precision scores for the core biology-related template element and template relation tasks become somewhat depressed. PASTA stopped short of extracting events, since these were not relevant for the system's users. However, it is clear that the overall PASTA framework could support event extraction, if required and so configured.

A good example of an ontology-driven system, which primarily targets events, is presented by GenIE [27]. The motivation behind this work is that it is not enough to extract single events, since much useful information might be missed in so doing. For example, where a sentence refers explicitly to a binding action, a following sentence might refer simply to *this interaction*, but may have something critically important to add. Anaphoric reference then must be handled, and the information from each sentence merged, or, at least, a dependency relation must be established between the events in each sentence. Some sentences also may discuss how a protein binds to an element and inhibits transcriptional activation by another protein. It is then important to capture the dependency between these two events—that is, the binding is responsible for the inhibition. This argues for a linguistic approach capable of identifying conceptual relations between events. This further argues for an ontology-driven approach to discourse analysis, and use of a domain model.

GenIE extracts information on biochemical pathways, and on sequences, structures, and functions of genomes and proteins. Input data can be as diverse as articles or database comment lines. In contrast to the pipeline of processes of the classic IE system, where problems may arise due to early-stage analysis errors, and where exclusively shallow processing prohibits the possibility of deeper analysis, GenIE is organized to allow deeper analysis to be entered. Such an organization implies adding not only appropriate deep analyzers, but also the resources (e.g., lexicon, ontology) required to support them. An explicit, well-defined semantic representation formalism, which is a variant of Discourse Representation Theory (DRT) [63], is employed, but any similar formalism could be substituted.

Like other classic IE systems, GenIE uses a partial parser [56], but this is only used for chunking purposes. GENIES initially attempts full parsing and uses partial parsing only when necessary as a safety net, while GenIE systematically applies chunking, followed by full parsing. Chunking of a sentence in GenIE typically will deliver subtrees for the same chunk, which contain either wholly syntactic elements in flat subtrees (like most chunkers), or a mixture of syntactic and semantic elements in deeper subtrees. The latter type of tree results from look-up of a multiword lexicon that contains semantically-typed terms. Since no (especially multiword) lexicon is ever complete, and since some sequences of words may be wrongly taken to be a multiword, chunking should

not attempt to resolve ambiguities involving multiwords. Instead, it should deliver both shallow and slightly deeper analyses for consideration by the following stage. This is in contrast to other partial parsing approaches, which offer only one analysis per chunk or multiple analyses at the same linguistic level. Here, multiple analyses for the same chunk are offered at two different levels: syntactic and synactic-semantic.

The next stage attempts to build a full sentential deep syntactic parse by combining and connecting the syntactic and syntactic-semantic subtrees from the partial parser. Furthermore, a semantic representation for each sentence is produced in the form of a DRT structure. However, Logical Description Grammar [64] is used as a basis, as described in [65]. This has been extended to combine syntactic, semantic, and ontological knowledge, and to handle scoping phenomena, which are a notorious source of ambiguity. The technique implies heavy use of an ontology, and of a semantic lexicon containing semantically underspecified entries in the form of "elementary trees" (tree schemata containing some uninstantiated, but typed, elements). A lexeme may map to multiple elementary trees (i.e., a word may have several readings).

Semantic processing involves choosing the correct elementary tree, or trees, to plug into the overall semantic representation being built. For example, an elementary tree for a lexeme that is a noun will contain, at its root, the category into which it can plug, and will give a mapping to a concept in the system's ontology and a description of the lexeme's semantics. The semantic description may be complex, as in the elementary tree given for *stem loop II*, which notes that:

- *stem loop II* is_a 'protein_domain';
- *stem loop II* has a part_of relation to a concept that is either 'protein' or a specialization thereof;
- The part_of relation is realized by a prepositional phrase;
- A partial order (reflecting dominance and scoping) exists between the semantic representation of this elementary tree and that of any elementary tree that will become embedded in the prepositional phrase.

As a further example, the elementary tree for the verbal lexeme for *bind* contains the information that:

- *bind* takes:
  - A subject agent that is a 'protein' concept or a specialization thereof;
  - An object patient that is a 'protein_domain' or specialization thereof.
- A partial order exists over the nodes in this elementary tree.

- The event that is represented by *bind* is encoded using Davidsonian-like reification.

Reification in this way allows the event variable to be referred to in other semantic constraints. This means that reference can be made to events in the same way as to other discourse referents. See the cited works for a full description of this approach.

As processing proceeds, the semantic, syntactic, and ontological constraints combine to filter out implausible readings, and retained appropriate elementary trees are integrated in the overall semantic representation. There is no insistence that a verb's arguments be satisfied in the one sentence. The possibility is left open for other sentences to supply missing arguments.

Like GENIES, GenIE relies on extensive subcategorization information for verbs to be present in its lexicon. This is provided via automatic corpus processing. It also requires an ontology of biochemical events. This has been constructed from a linguistic viewpoint, rather than from a conceptual viewpoint. That is, verbs were grouped into ontological classes that were assigned appropriate semantics, rather than a number of classes being established, and then verbs found to refer to them. Compare this with the GENIES approach to ontological mapping. However, the ontology is for test purposes, and is thus small (129 concepts). With the ontology and the DRT representations, it becomes possible to establish conceptual relations between events that are antecedents of other events, states, and/or entities.

Evaluation of GenIE proved problematic, due to low interannotator agreement on discourse-level annotation for the evaluation corpus. See the discussion in Chapter 9 on gold standard annotation. Highest recall was 55%, and precision was highly variable. However, the authors conclude that the system extracts well above 50% of the most reliably annotated relations, and extracts fewer than 10% of relations that are clearly incorrect. A final point to note is that GenIE does not appear, from the available description, to have employed a strong sublanguage approach. It has appropriate mechanisms for handling subcategorization and selectional restriction, but no mention is made of whether the argument structures are based on general language principles or on GENIES-like sublanguage principles. The elementary tree for *stem loop II* described above would tend to indicate that a sublanguage description approach had not been followed.

7.4 Conclusion

Our review of recent research on rule-based extraction of relations (simple facts) and events from biotexts reveals that there are many types of approaches. These

vary greatly in terms of the degree to which they incorporate linguistic information, and in their use of such information. They also vary greatly in terms of how they position themselves, with respect to the no parsing versus/partial/parsing versus full parsing debate. There is evidence of much experimentation with parsing strategies: full parsing on its own; full parsing followed by fall-back partial parsing; partial parsing followed by full parsing; partial parsing on its own; parsing applied to all sentences; parsing applied only to sentences judged relevant to the task; and so forth. There is a marked difference between pattern-matching approaches, on the one hand, and more NLP-oriented approaches, on the other. NLP techniques become essential when there is a need to extract abstract representations of facts rather than matching fragments of text. Hybrid syntactic-semantic approaches offer promising results, particularly where these are based on a strong sublanguage approach. Ontology-driven approaches also are promising [62]. A system that combined a fully ontology-driven approach with a syntactic-semantic sublanguage-based approach would offer an interesting direction for future research. We have not yet seen such a system being applied to biotexts, although GenIE and GENIES represent different and overlapping aspects of such a system. Table 7.3 summarizes the various types of approaches.

Unfortunately, there are two bottlenecks hampering development in the field: ontology and sublanguage lexicon. It is not enough to rely on one or the other. Both are needed if we wish to produce highly accurate results required by biologists, and to obtain broad coverage of biotexts. As we have seen in the work of GENIES and GenIE, it is essential to focus on describing the syntactic and semantic behavior of sublanguage verbs, and on the description of domain event concepts. It is notable that both GENIES and GenIE had to develop their own ontology of events and their own lexicons, with the GENIES lexicon being especially rich in sublanguage verbal description. If rule-based approaches are to improve beyond their current stage to deliver the desired results, then the challenge for the field is to develop appropriate ontological and sublanguage lexical resources to support and link the kind of processing required.

In closing, we note that IE has come a long way from its MUC days. To a great extent, the nature of biology text and the need for accurate and comprehensive extraction of facts from scientific text have been instrumental in driving research in IE beyond the classic MUC model, towards greater understanding of how scientific knowledge is communicated in texts, and towards greater exploitation of such understanding. There is still much work to be done. Fact extraction systems for biology are still very much at the research stage. There are issues to be tackled, such as in scalability and efficiency, particularly where intensive use is made of lexical and ontological resources, and in multilevel analysis of language. However, the breadth, depth, and sheer variety of activities in IE in

Table 7.3
Summary of Rule-Based Approaches to Fact Extraction

General Class	Subtype/Characteristics	Comments
Simple Pattern Matching	Over text strings	Many patterns required, no generalizations possible
	Over parts of speech	Some generalizations, but ignores sentence structure
	POS tagging, chunking, semantic typing, semantic pattern matching	Limited generalization, some account taken of structure, limited consideration of sublanguage patterns (in reported systems)
	Pattern matching over results of dictionary look-up	Many patterns required, no great generalization possible
	Preposition-based: inspect context around prepositions	Fragments fill slots in templates, but cannot be easily mapped to ontology
Bare Context Free Parsing		Few linguistic constraints, nonrobust
Full Parsing	Full parsing on its own, or parsing done in combination with various other techniques (chunking, partial parsing, heuristics) to reduce ambiguity and filter out uninteresting sentences or implausible readings	Theories of general language inappropriate and difficult to specialize for biotext; plethora of analyses per sentence; missing of relevant information for sublanguage meanings
Probabilistic Parsing	Uses rules weighted from corpus evidence	Robust processing; core algorithms may require adaptation to deal with sublanguage phenomena
Syntactic-Semantic Parsing	Leverages two levels simultaneously	The natural technique to deal with sublanguages, in which semantic meaning is heavily syntactified
Sublanguage Approach	Exploits a rich lexicon, describing especially sublanguage verbal behavior, and a syntactic-semantic grammar, derived from close study of informational structure of sublanguage texts	Current systems would benefit from adopting also an ontology-driven approach
Ontology-Driven Approach	Uses event concept frames to strongly guide processing and the integration of extracted information	Current systems would benefit from adopting also a sublanguage approach
Ontology-Driven Sublanguage Approach	Leverages both event concept frames and sublanguage descriptions	Promising avenue for future research

biology-related domains are highly indicative of a research community whose efforts are likely to lead to major advances in the near future.

References

[1] Spiliopoulou, M., et al., "Coupling Information Extraction and Data Mining for Ontology Learning in PARMENIDES," *Proc. RIAO2004*, Avignon, 2004.

[2] Nédellec, C., "Machine Learning for Information Extraction in Genomics—State of the Art and Perspectives," *Text Mining and Its Applications*, S. Sirmakessis, (ed.), Berlin: Springer-Verlag, Studies in Fuzziness and Soft Computing, Vol. 138, 2004, pp. 99–118.

[3] Bunescu, R., et al., "Comparative Experiments on Learning Information Extractors for Proteins and Their Interactions," *Artificial Intelligence in Medicine*, Vol. 33, 2005, pp. 139–155.

[4] Allen, J., *Natural Language Understanding*, 2nd ed., Reading, MA: Bradford Books, 1998.

[5] Jurafsky, D., and J. H. Martin, *Speech and Language Processing*, Prentice-Hall Series in Artificial Intelligence, Upper Saddle River, NJ: Prentice-Hall, 2000.

[6] Hirschman, L., "The Evolution of Evaluation: Lessons from the Message Understanding Conferences," *Computer Speech and Language*, Vol. 12, 1998, pp. 281–305.

[7] Hobbs, J. R., "The Generic Information Extraction System," *Proc. Fifth Message Understanding Conf. (MUC-5)*, Baltimore, MD: Morgan Kaufmann, 1993, pp. 87–92.

[8] Hobbs, J. R., "Information Extraction from Biomedical Text," *Journal of Biomedical Informatics*, Vol. 35, No. 4, 2002, pp. 260–264.

[9] Blaschke, C., L. Hirschman, and A. Valencia, "Information Extraction in Molecular Biology," *Briefings in Bioinformatics*, Vol. 3, No. 2, 2002, pp. 1–12.

[10] Cohen, A. M., and W. R. Hersh, "A Survey of Current Work in Biomedical Text Mining," *Briefings in Bioinformatics*, Vol. 6, 2005, pp. 57–71.

[11] De Bruijn, B., and J. Martin, "Getting to the (C)ore of Knowledge: Mining Biomedical Literature," *International Journal of Medical Informatics*, Vol. 67, 2002, pp. 7–18.

[12] Hale, R., "Text Mining: Getting More Value from Literature Resources," *Drug Discovery Today*, Vol. 10, No. 6, 2005, pp. 377–379.

[13] Krallinger, M., R. A.-A. Erhardt, and A. Valencia, "Text-Mining Approaches in Molecular Biology and Biomedicine," *Drug Discovery Today*, Vol. 10, No. 6, 2005, pp. 439–445.

[14] Mack, R., and M. Hehenburger, "Text-Based Knowledge Discovery: Search and Mining of Life-Sciences Documents," *Drug Discovery Today*, Vol. 7, No. 11, 2002, pp. S89–S98.

[15] Shatkay, H., and R. Feldman, "Mining the Biomedical Literature in the Genomic Era: An Overview," *Journal of Computational Biology*, Vol. 10, No. 6, 2004, pp. 821–855.

[16] Yandell, M. D., and W. H. Majoros, "Genomics and Natural Language Processing," *Nature Reviews/Genetics*, Vol. 3, 2002, pp. 601–610.

[17] Yeh, A. H., L. Hirschman, and A. A. Morgan, "Evaluation of Text Data Mining for Database Curation: Lessons Learned from the KDD Challenge Cup," *Bioinformatics*, Vol. 19, Suppl. 1, 2003, pp. i331–i339.

[18] Hirschman, L., A. A. Morgan, and A. S. Yeh, "Rutabaga by Any Other Name," *Journal of Biomedical Informatics*, Vol. 35, No. 4, 2002, pp. 247–259.

[19] Rebholz-Schuhmann, D., H. Kirsch, and F. Couto, "Facts from Text—Is Text Mining Ready to Deliver?" *PLoS Biology*, Vol. 3, No. 2, 2005, pp. 0188–0191, http://www.plosbiology.org, accessed June 2005.

[20] Blaschke, C., et al., "Automatic Extraction of Biological Information from Scientific Text: Protein-Protein Interactions," *Proc. 7th Int. Conf. on Intelligent Systems for Molecular Biology (ISMB)*, 1999, pp. 60–67.

[21] Divoli, A., and T. K. Attwood, "BioIE: Extracting Informative Sentences from the Biomedical Literature," *Bioinformatics*, Vol. 21, No. 9, 2005, pp. 2138–2139.

[22] Vailaya, A., et al., "An Architecture for Biological Information Extraction and Representation," *Bioinformatics*, Vol. 21, No. 4, 2005, pp. 430–438.

[23] Ono, T., et al., "Extraction of Information on Protein-Protein Interactions from the Biological Literature," *Bioinformatics*, Vol. 17, No. 2, 2001, pp. 155–161.

[24] Huang, M., et al., "Discovering Patterns to Extract Protein-Protein Interactions from Full Biomedical Texts," *Bioinformatics*, Vol. 20, 2004, pp. 3604–3612.

[25] Corney, D. P., et al., "BioRAT: Extracting Biological Information from Full-Length Papers," *Bioinformatics*, Vol. 20, No. 17, 2004, pp. 3206–3213.

[26] Hu, Z., et al., "Literature Mining and Database Annotation of Protein Phosphorylation Using a Rule-Based System," *Bioinformatics*, Vol. 21, 2005, pp. 2759–2765.

[27] Cimiano, P., U. Reyle, and J. Šarić, "Ontology-Driven Discourse Analysis for Information Extraction," *Data & Knowledge Engineering*, Vol. 55, No. 1, 2005, pp. 59–83.

[28] Gaizauskas, R., et al., "Protein Structures and Information Extraction from Biological Texts: The PASTA System," *Bioinformatics*, Vol. 19, No. 1, 2003, pp. 135–143.

[29] Šarić, J., L. J. Jensen, and I. Rojas, "Large-Scale Extraction of Gene Regulation for Model Organisms in an Ontological Context," *Silico Biology*, Vol. 5, No. 0004, 2004, http://www.bioinfo.de.isb/2004/05/0004/, accessed June 2005.

[30] Martin, E. P., et al., "Analysis of Protein/Protein Interactions Through Biomedical Literature: Text Mining of Abstracts vs. Text Mining of Full Articles," *Knowledge Exploration in Life Science Informatics*, J. López, E. Benfenati, and W. Dubitzky, (eds.), Berlin: Springer-Verlag, Vol. 3303 of *Lecture Notes in Artificial Intelligence*, 2004, pp. 96–108.

[31] Leroy, G., and H. Chen, "Filling Preposition-Based Templates to Capture Information from Medical Abstracts," *Proc. Pacific Symp. on Biocomputing 7 (PSB)*, 2002, pp. 362–373.

[32] Leroy, G., H. Chen, and J. D. Martinez, "A Shallow Parser Based on Closed-Class Words to Capture Relations in Biomedical Text," *Journal of Biomedical Informatics*, Vol. 36, 2003, pp. 145–158.

[33] Leroy, G., and H. Chen, "Genescene: An Ontology-Enhanced Integration of Linguistic and Co-Occurrence Based Relations in Biomedical Texts," *Journal of the American Society for Information Science and Technology*, Vol. 56, No. 5, 2005, pp. 457–468.

[34] Temkin, J. M., and M. R. Gilder, "Extraction of Protein Interaction Information from Unstructured Text Using a Context-Free Grammar," *Bioinformatics*, Vol. 19, No. 16, 2003, pp. 2046–2053.

[35] Grishman, R., "The NYU System for MUC-6 or Where's the Syntax," *Proc. 6th Message Understanding Conf. (MUC-6)*, Columbia, MD, 1995, pp. 167–176.

[36] Hobbs, J. R., "SRI International's TACITUS System: MUC-3 Test Results and Analysis," *Proc. Third Message Understanding Conf. (MUC-3)*, San Diego, CA: Morgan Kaufmann, 1991, pp. 105–107.

[37] Pyysalo, S., et al., "Analysis of Link Grammar on Biomedical Dependency Corpus Targeted at Protein-Protein Interactions," *Proc. Int. Joint Workshop on Natural Language Processing in Biomedicine and its Applications (JNLPBA)*, 2004, pp. 15–21.

[38] Sleator, D., and D. Temperley, "Parsing English with a Link Grammar," *3rd Int. Workshop on Parsing Technologies*, 1993, pp. 277–291.

[39] Harris, Z., "The Structure of Science Information," *Journal of Biomedical Informatics*, Vol. 35, 2002, pp. 215–221.

[40] Alphonse, E., et al., "Event-Based Information Extraction for the Biomedical Domain: The Caderige Project," *Proc. Int. Workshop on Natural Language Processing in Biomedicine and its Applications (JNLPBA)*, 2004, pp. 43–49.

[41] Yakushiji, A., et al., "Event Extraction from Biomedical Papers Using a Full Parser," *Proc. Pacific Symp. on Biocomputing 6*, 2001, pp. 408–419.

[42] Pollard, C. J., and I. A. Sag, *Head-Driven Phrase Structure Grammar*, Chicago, IL: University of Chicago Press, 1994.

[43] Voutilainen, A., "Designing a (Finite-State) Parsing Grammar," in *Finite-State Language Processing*, E. Roche, and Y. Schabes, (eds.), Boston, MA: MIT Press, 1996, pp. 283–310.

[44] Kim, J., and J. C. Park, "BioIE: Retargetable Information Extraction and Ontological Annotation of Biological Interactions from the Literature," *Journal of Bioinformatics and Computational Biology*, Vol. 3, No. 3, 2004, pp. 551–568.

[45] Steedman, M., *The Syntactic Process*, Boston, MA: MIT Press, 2000.

[46] Park, J. C., H. S. Kim, and J. J. Kim, "Bidirectional Incremental Parsing for Automatic Pathway Identification with Combinatory Categorical Grammar," *Proc. Pacific Symp. on Biocomputing 6*, 2001, pp. 396–407.

[47] Friedman, C., et al., "GENIES: A Natural-Language Processing System for the Extraction of Molecular Pathways from Journal Articles," *Bioinformatics*, Vol. 17, 2001, pp. S74–S82.

[48] Daraselia, N., et al., "Extracting Human Protein Interactions from MEDLINE Using a Full-Sentence Parser," *Bioinformatics*, Vol. 20, No. 5, 2004, pp. 604–611.

[49] Novichkova, S., S. Egorov, and N. Daroselia, "MedScan, a Natural Language Processing Engine for MEDLINE Abstracts," *Bioinformatics*, Vol. 19, 2003, pp. 1699–1706.

[50] Kaplan, R., and J. Bresnan, "Lexical-Functional Grammar: A Formal System for Grammatical Representation," in *The Mental Representation of Grammatical Relations*, J. Bresnan, (ed.), Cambridge, MA: MIT Press, 1982, pp. 173–281.

[51] Friedman, C., P. Kra, and A. Rzhetsky, "Two Biomedical Sublanguages: A Description Based on the Theories of Zellig Harris," *Journal of Biomedical Informatics*, Vol. 35, No. 4, 2002, pp. 222–235.

[52] Johnson, R. L., "Parsing with Transition Networks," in *Parsing Natural Language*, M. King, (ed.), London, England: Academic Press, 1983, pp. 59–73.

[53] Rinaldi, F., et al., "Mining Relations in the GENIA Corpus," *Proc. Second European Workshop on Data Mining and Text Mining for Bioinformatics*, 2004, pp. 61–68.

[54] Briscoe, E., and J. Carroll, "Robust Accurate Statistical Annotation of General Text," *Proc. 3rd Int. Conf. on Language Resources and Evaluation (LREC 2002)*, 2002, pp. 1499–1504.

[55] McDonald, D. M., et al., "Extracting Gene Pathway Relations Using a Hybrid Grammar: The Arizona Relation Parser," *Bioinformatics*, Vol. 20, No. 18, 2004, pp. 3370–3378.

[56] Abney, S., "Partial Parsing via Finite-State Cascades," *Natural Language Engineering*, Vol. 2, No. 4, 1996, pp. 337–344.

[57] Friedman, C., (ed.), "Special Issue: Sublanguage," *Journal of Biomedical Informatics*, Vol. 35, No. 4, 2002.

[58] Michiels, A., "The Frame-for-Term Approach to Terminology in Eurotra," *Meta*, Vol. 39, No. 1, 1994, pp. 184–193.

[59] Selsøe Sørensen, H., "The Use of Knowledge-Based Frames for Terms in Eurotra," *Proc. Terminology and Knowledge Engineering*, 1990, pp. 300–309.

[60] Rzhetsky, A., et al., "A Knowledge Model for Analysis and Simulation of Regulatory Networks," *Bioinformatics*, Vol. 16, No. 12, 2000, pp. 1120–1128.

[61] Rzhetsky, A., et al., "GeneWays: A System for Extracting, Analyzing, Visualizing and Integrating Molecular Pathway Data," *Journal of Biomedical Informatics*, Vol. 37, No. 1, 2004, pp. 43–53.

[62] Black, W. J., et al., "Ontology-Enablement of a System for Semantic Annotation of Digital Documents," *Proc. Int. Semantic Web Conf. 2004 Workshop on Knowledge Markup and Semantic Annotation (Semannot 2004)*, 2004, http://CEUR-WS.org, June 2005.

[63] Kamp, H., and U. Reyle, *From Discourse to Logic*, Dordrecht: Kluwer, 1993.

[64] Muskens, R., "Talking About Trees and Truth-Conditions," *Logic, Language and Information*, Vol. 10, No. 4, 2001, pp. 417–455.

[65] Cimiano, P., and U. Reyle, "Talking About Trees, Scope and Concepts," *Proc. 6th Int. Workshop on Computational Semantics (IWCS-6)*, Tilburg, The Netherlands, 2005, in press.

8

Corpora and Their Annotation

Jin-Dong Kim and Jun'ichi Tsujii

8.1 Introduction

Text mining is the extraction of information or knowledge from databases of literature, while data mining assumes databases of structured data as the source of knowledge. Contrary to the usual case in data mining, knowledge in text mining is assumed to be encoded in natural language text, which poses problems for computational analysis (see Chapter 2). It is desirable for text mining systems to be made aware of language, in order to identify effectively the structures of interest in the text.

Natural Language Processing (NLP) techniques have been successfully applied to fields that need to exploit natural language texts. Corpora, which are the collections of written (or spoken) material of a language upon which linguistic analysis is based, have always been at the heart of NLP research, providing the reference material to put flesh on the skeleton of theoretic models. Increasingly, annotations are made over corpora, to make explicit the structures of interest that implicitly reside in text, and to provide direct reference to these structures.

This chapter gives an overview of corpora and annotations that are available in the domain of biomedicine, and discusses related issues and problems.

8.2 Literature Databases in Biology

Literature databases contain a large amount of knowledge, and become knowledge sources to be mined for knowledge nuggets. This section introduces literature databases available in the biomedical research community. Copyright issues also are addressed, since text mining is inherently subject to copyright conditions.

8.2.1 Literature Databases

In the history of text mining for biology, Medical Literature, Analysis, and Retrieval System Online (MEDLINE) has played a primary role as the main source of knowledge of biology and medical sciences. MEDLINE is the U.S. National Library of Medicine's (NLM) database of indexed journal citations, covering nearly 4,500 journals in life sciences with a concentration on biomedicine. Currently, it includes more than 13 million references to articles indexed from 1966 to the present.[1] Approximately one-half of MEDLINE entries have abstracts,[2] and the texts of the abstracts have become the target of text mining for biology.

MEDLINE can be accessed from the Internet via several search services. NLM provides MEDLINE access via the PubMed service (http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=PubMed), which is sometimes confused with MEDLINE. According to NLM's Fact Sheet (http://www.nlm.nih.gov/pubs/factsheets/factsheets.html), "What's the Difference Between *MEDLINE* and *PubMed*?" PubMed contains the following, in addition to MEDLINE citations:

- OLDMEDLINE for pre-1966 citations;
- Citations to articles that are out-of-scope (e.g., covering plate tectonics or astrophysics) from certain MEDLINE journals, primarily general science and general chemistry journals, from which the life sciences articles are indexed in MEDLINE;
- In-process citations, which provide a record for an article before it is indexed with MeSH and added to MEDLINE, or converted to out-of-scope status;
- Citations that precede the date that a journal was selected for MEDLINE indexing (when supplied electronically by the publisher);

1. At the time of this writing, MEDLINE contains 13,239,648 entries.
2. At the time of this writing, 7,457,624 MEDLINE entries out of 13,239,648 (56%) have abstracts.

- Some life science journals that submit full text to PubMedCentral and may not have been recommended for inclusion in MEDLINE, although they have undergone a review by NLM, and some physics journals that were part of a prototype PubMed in the early- to mid-1990s.

While MEDLINE is still playing a key role in text mining for biology as a primary knowledge source, interests have begun to move to full texts. Several efforts now exist to build literature databases providing access to full texts. Currently, PubMed Central (PMC) (http://www.pubmedcentral.nih.gov/) is archiving articles deposited by voluntary publishers, and all journals in PMC provide free access to full text, sometimes on a delayed basis.[3] PMC is a digital archive operated by the NCBI, a division of NLM. It is also accessible via PubMed, since every article in PMC has a corresponding entry in PubMed.

8.2.2 Copyright Issues

Literature databases contain mostly published articles that are intellectual properties, and so the text mining research that is based on them is inherently subject to copyright conditions. Likewise, since copyright protection may pose a barrier to academic progress, there has been an open source movement to publish scholarly literature on the Internet, and make it available to readers free of charge, and free of licensing restrictions. The research on text mining for biology also benefits from this movement. It is said that the free access principle, which applies to all full text content viewable in PMC, removes price barriers (e.g., subscriptions, licensing fees, and pay-per-view fees), offering unlimited access to the content through the Internet. Furthermore, many PMC journals make all, or most, of their content available as open access publications.[4] Open access goes beyond free access, by removing most copyright and licensing restrictions. The Budapest Open Access Initiative (http://www.soros.org/openaccess/) puts it in this way:

> By "open access" to this literature, we mean its free availability on the public internet, permitting any users to read, download, copy, distribute, print, search, or link to the full texts of these articles, crawl them for indexing, pass them as data to software, or use them for any other lawful purpose, without financial, legal, or technical barriers other than those inseparable from gaining access to the internet itself. The only constraint on reproduction and

3. At the time of this writing, articles from 185 journals are being archived in PMC.
4. At the time of this writing, 142 PMC journals provide all or part of their article as open access.

> distribution, and the only role for copyright in this domain, should be to give authors control over the integrity of their work and the right to be properly acknowledged and cited.

Examples of publishers supporting open access include BioMed Central (BMC) (http://www.biomedcentral.com/), and Public Library of Science (PLoS) (http://www.plos.org/). BMC is an independent, commercial publishing house, committed to providing immediate, free access to peer-reviewed biomedical research. Every peer-reviewed research article appearing in any journal published by BMC is open access, and is archived without delay in PMC. PLoS is a nonprofit organization of scientists and physicians, committed to making the world's scientific and medical literature a public resource. The articles of PLoS journals also are immediately available from PMC.

8.3 Corpora

A corpus is a special collection of texts, which are carefully chosen according to a certain set of criteria, to be representative of a certain language. Such a corpus is often referenced when building a language model to process text. For example, the *Wall Street Journal* (WSJ) corpus is a collection of WSJ articles from Dow Jones, Inc., and represents a journalistic newswire style of writing. Another example is the Brown corpus, which consists of text samples ranging from journalistic and science domains, to fiction and speech transcriptions, and was intended to represent general English.

8.3.1 Corpora in Biology

Since MEDLINE contains a large number of texts that are abstracts of biomedicine research, the whole or a subset of the MEDLINE abstracts can constitute a corpus that represents scientific writing covering the general, or a specific, domain of life sciences. In fact, the three most widely used corpora [1] in this domain (the GENIA corpus [2], the GENETAG corpus [3], and the Yapex corpus [4]) are comprised of texts taken from MEDLINE.

GENIA is intended to cover biological reactions concerning transcription factors in human blood cells. Correspondingly, the corpus consists of the text of abstracts resulting from the following search query on PubMed: "*Human*"[MeSH] AND "*Blood Cells*"[MeSH] AND "*Transcription Factors*"[MeSH]. The main value of the GENIA corpus comes from the annotation encoded in it, which provides a guide to the interpretation of the text. The detail of the annotation will be discussed in the following section.

Likewise, Caderige [5] consists of sentences from MEDLINE abstracts retrieved by using the focused query *"Bacillus subtilis"* AND *"transcription"*. The

Caderige team chose this specific domain because *Bacillus subtilis* is a model bacterium, and *transcription* is a central phenomenon in functional genomics involved in genic interaction.

The PennBioIE corpus [6] is intended to cover two focused interests on cancer genomics and drug development. With these applications in mind, this corpus was collected in the oncology domain and the cytochrome P450 (CYP) domain.

The observations in this section suggest that the development of text mining systems may benefit from understanding the characteristics of the subject domain.

8.3.2 Collecting MEDLINE Abstracts

Despite the growing interest in full text articles, the MEDLINE database is still considered as a primary knowledge source of biology and medical science, and it is popular to build corpora of specific subdomains by collecting text from MEDLINE abstracts. This section is intended to be a guide for those who wish to select MEDLINE abstracts of relevance to their interest in a specific subject area.

PubMed is equipped with a Web query interface, so that indexed citations of interest can be conveniently searched. However, if we want to run a search on PubMed to retrieve numerous items, it is much more convenient to use Entrez Programming Utilities (E-Utilities). Designed to be utilized by automated programs over the Internet, E-utilities provide functions to search and retrieve Entrez data. Figure 8.1 is a simple perl script to retrieve abstracts from MEDLINE that were published within a specific period.

Note that this script utilizes 'esearch' (line 11) and 'efetch' (line 28) utilities, located at http://www.ncbi.nlm.nih.gov/entrez/eutils/ (line 6). The search is limited to the citations indexed in MEDLINE that are written in English and have abstracts (line 15). The retrieved abstracts will be sorted by the publication date (line 14). The citation results will return in XML (line 29), and only text in abstracts will be extracted (lines 37–39). As seen in line 15, terms may be tagged to specify the fields under which the terms are to be qualified. In the above example, the term 'medline' will be searched in the 'subset' field whose tag name is 'sb,' and the term 'english' in the 'language' field whose tag name is 'la'.

The 'Publication Type [PT]' and 'MeSH Terms [MH]' fields are other examples of search fields that the authors think deserve to be mentioned. MeSH (http://www.nlm.nih.gov/mesh/meshhome.html) is NLM's controlled vocabulary thesaurus. All citations in PubMed are assigned Publication Types (e.g., Clinical Trial, Journal Article, and so forth), and MeSH terms from the thesaurus, which permit searching at various levels of specificity. A list of the available

```
#!/usr/bin/perl-w
# Sample script to download abstracts from MEDLINE
# modification made by Jin-Dong Kim (http://jdkim.net)
 # on the script by Oleg Khovayko(http://olegh.spedia.net)
use LWP::Simple;
my $utils= "http://www.ncbi.nlm.nih.gov/entrez/eutils";
my $mindate= "1989/07/01";
my $maxdate= "1989/07/01";
my $query   = "";

# commit the query to ESearch
my $esearch= "$utils/esearch.fcgi?db=Pubmed" .
                        "&retmax=1&usehistory=y" .
                        "&mindate=$mindate&maxdate=$maxdate" .
                        "&datetype=pdat&sort=pub+date" .
                        "&term=medline[sb]+AND+english[la]+AND+hasabstract";
if ($query) {$esearch= $esearch. "+AND+" . $query}
my $esearch_result= get($esearch);
$esearch_result=~
        m|<Count>(d+)</Count>.*<QueryKey>(d+)</QueryKey>.*<WebEnv>(S+)</WebEnv>|s;
my $Count    = $1;
my $QueryKey= $2;
my $WebEnv= $3;
print STDERR "$Count citation results from esearch.\n";

# store the abstracts of the citation results from Efetch
my $retmax= 2000;
my $retstart;
for($retstart=0; $retstart<$Count; $retstart+=$retmax) {
        my $efetch= "$utils/efetch.fcgi?".
                        "rettype=xml&retmode=text" .
                        "&retstart=$retstart&retmax=$retmax&" .
                        "db=Pubmed&query_key=$QueryKey&WebEnv=$WebEnv";

        my $efetch_result= get($efetch);
        my @articles =
                        split m|</PubmedArticle>\s*<PubmedArticle>|, $efetch_result;
        foreach $article (@articles) {
                        $article =~ m|<AbstractText>([^<]+)</AbstractText>|s;
                        my $abstract = $1;
                        print "$abstract\n\n";
        }
}
```

Figure 8.1 Sample perl script to retrieve MEDLINE abstracts.

field names, tags, and brief field descriptions may be found in the on-line PubMed Help, under "Search Field Descriptions and Tags."

8.3.3 Comparing Corpora

This section provides comparisons between various corpora, as a way of obtaining insights into them. We analyze a sample of the MEDLINE corpus to highlight the characteristics of the domain that it covers. We also analyze GENIA and Caderige to uncover the characteristics of the focused domains that the two

corpora cover. To achieve this, WSJ was used as the reference corpus to which each of the two study corpora was compared. WSJ is included in the Penn Treebank corpus [7], which often serves as a benchmark for corpus-based NLP research.

We used WordSmith version 4.0, a lexical analysis tool from Oxford University Press, to analyze and compare the corpora [8]. Using this software, Dunning's Log-likelihood (G^2) statistic [9] was calculated for all the words in each corpus. The words with a high G^2 value were chosen as the keywords characterizing the domain of each corpus. It is generally thought that the keywords extracted by WordSmith indicate "aboutness" of the text analyzed [10].

Table 8.1 lists the 50 top-ranked keywords characterizing the texts in MEDLINE abstracts. The study corpus was comprised of 14,451 MEDLINE abstracts, published on June 1, 1998, that were retrieved using the script of Figure 8.1.

We observe that there are many content words characterizing the MEDLINE domain (e.g., PATIENTS, CELLS, PLASMA, CELL, CLINICAL, DISEASE, BLOOD, TREATMENT, PATIENT, THERAPY, DIAGNOSIS, RENAL, ACID, CORONARY, and so forth), and words that often appear in scientific writing (e.g., STUDY, SIGNIFICANTLY, STUDIED, SIGNIFICANT, OBSERVED, ASSOCIATED, FOUND, and so forth). There are also many occurrences of "WERE," "WAS," and "WITH." Tables 8.2 and 8.3 list the three-word clusters with which "WERE" and "WITH" often appear in MEDLINE abstracts, respectively. We note that a large part of these are related to scientific language, describing experimental processes (e.g., WAS ASSOCIATED, WERE COMPARED, WERE TREATED, WERE STUDIED, WERE FOUND, WERE OBSERVED, WERE PERFORMED, THE RESULTS WERE, and so forth).

Table 8.4 shows the list of 50 top-ranked keywords describing the GENIA domain. The study corpus was comprised of 399 MEDLINE abstracts published from 1987 to 1991, which were retrieved by using the script of Figure 8.1, with the following modifications:

> $mindate = "1987/01/01" (line 7); $maxdate = "1991/12/31" (line 8); $query = "human[mh]+AND+blood+cells[mh]+AND+transcription+factors[mh]" (line 9).

The list of keywords indicates that the literature in the GENIA domain, when compared to the general domain, contains a significant number of terms about biological entities (e.g., CELL, PROTEIN, GENE, DNA, and so forth); their functions (e.g., ENHANCER, RECEPTOR, PROMOTER, and so forth); and their interactions (e.g., BINDING, TRANSCRIPTION, ACTIVATION, INDUCED, ACTIVITY, INDUCTION, and so forth).

Table 8.1
50 Top-Ranked Keywords of MEDLINE

PATIENTS, WERE, WITH, OF, CELLS, AND, WAS, STUDY, SUBJECTS, PLASMA, CELL, CLINICAL, SIGNIFICANTLY, DISEASE, NORMAL, BLOOD, TREATMENT, PATIENT, STUDIED, SERUM, LESS, IN, THERAPY, SIGNIFICANT, THESE, CASES, #, OBSERVED, ML, CONCENTRATIONS, ASSOCIATED, FOUND, DIAGNOSIS, MG, RATS, INDUCED, AGE, LEVELS, RENAL, EFFECTS, CONCENTRATION, ACID, DECREASED, GROUPS, DURING, PERFORMED, TREATED, SHOWED, CORONARY ...

Table 8.2
20 Top-Ranked Three-Word Clusters Appearing with "WITH"

# PATIENTS WITH	IN PATIENTS WITH
WAS ASSOCIATED WITH	COMPARED WITH THE
OF PATIENTS WITH	WERE COMPARED WITH
ASSOCIATED WITH A	WITH # #
# # WITH	IS ASSOCIATED WITH
WHEN COMPARED WITH	COMPARED WITH THOSE
PATIENTS WITH A	WITH RESPECT TO
WERE TREATED WITH	WITH A #
# COMPARED WITH	P LESS THAN
A PATIENT WITH	AND # WITH

Table 8.3
20 Top-Ranked Three-Word Clusters Appearing with "WERE"

# # WERE	WERE COMPARED WITH
WERE STUDIED IN	WERE FOUND TO
IN # PATIENTS	THERE WERE NO
AND # WERE	THE PATIENTS WERE
WERE FOUND IN	WERE OBSERVED IN
FOUND TO BE	SIGNIFICANT DIFFERENCES WERE
WERE # #	# # #
IN # #	WERE TREATED WITH
WERE USED TO	THE RESULTS WERE
WERE PERFORMED IN	# YEARS WERE
. . .	

Table 8.4
50 Top-Ranked Keywords of GENIA Domain

CELLS, CELL, BINDING, EXPRESSION, B, PROTEIN, T, RECEPTOR, KAPPA, HUMAN, GENE, NF, TRANSCRIPTION, LYMPHOCYTES, DNA, ACTIVATION, ENHANCER, PATIENTS, RECEPTORS, ALPHA, SPECIFIC, FACTOR, HIV, GLUCOCORTICOID, PROMOTER, GENES, MRNA, PROTEINS, INDUCED, C, IL, SEQUENCE, OF, OH, ACTIVITY, NUCLEAR, THESE, VIRUS, SITES, INDUCTION, BETA, ELEMENT, GR, FOS, CORTISOL, JUN, LEVELS, DIFFERENTIATION, TRANSCRIPTIONAL, . . .

Table 8.5 shows the 50 top-ranked keywords describing the Caderige domain. The study corpus was extracted by using the script of Figure 8.1, with the following modifications:

$mindate = "1987/01/01" (line 7); $maxdate = "1991/12/31" (line 8);
$query = "Bacillus+subtilis+AND+transcription" (line 9).

The keyword list indicates that biological entities (e.g., GENE, PROTEIN, DNA, RNA, MRNA, and so forth) are frequently mentioned in this domain, in a similar fashion to the GENIA domain. It seems that the major difference between the two domains comes from the subject organisms: Bacillus subtilis (monocell organism) in the case of Caderige, and humans (multicell organism) in the case of GENIA. In particular, terms about monocell organisms (e.g., SUBTILIS, BACILLUS, COLI, E, ESCHERICHIA, and so forth) are frequently observed, as well as terms related to their transcriptional events (e.g., SIGMA, TRANSCRIPTION, PROMOTER, OPERON, POLYMERASE, PROMOTORS, UPSTREAM, and so forth). Some frequently observed terms about events are domain specific (e.g., SPORULATION), and some terms seem to be highly ranked due to the simple structure of the subject organism (e.g., SYNTHESIS, DELETION). It is also perceived that experimental

Table 8.5
50 Top-Ranked Keywords of Caderige Domain

GENE, SUBTILIS, SIGMA, TRANSCRIPTION, PROMOTER, SEQUENCE, EXPRESSION, BACILLUS, PROTEIN, DNA, GENES, SPORULATION, RNA, COLI, OF, REGION, E, B, OPERON, POLYMERASE, AMINO, MUTATIONS, BINDING, SEQUENCES, ACID, PLASMID, PROMOTERS, UPSTREAM, CELLS, ESCHERICHIA, LACZ, CLONED, MRNA, SYNTHESIS, TRANSCRIPTIONAL, SITE, MUTATION, CONTAINING, BETA, PROTEINS, FACTOR, ANALYSIS, DELETION, VITRO, CELL, LOCUS, NUCLEOTIDE, MUTANT, WAS, . . .

devices differ, depending on the subject organism (e.g., LACZ, CLONED, MUTATION).

8.4 Corpus Annotation in Biology

The text in literature databases is written in natural language, which makes machine access to desired information difficult. Since it is generally assumed that humans are the most suited to interpret natural language text, corpus annotation is often carried out to encode humans' interpretation into text, and consequently to provide machines with direct access to the innate text structure. This section introduces annotations made at various levels over corpora in the domain of biomedicine.

8.4.1 Annotation for Biomedical Entities

The most fundamental structures of interest in biomedical research are biological entities like proteins or genes, and identifying occurrences of such entities in texts is considered crucial to access useful information. There are a number of corpora providing annotation at this level, including GENIA, GENETAG, and PennBioIE.

GENIA provides extensive annotation for biomedical terms that include entity references. The annotation is grounded on the GENIA ontology, which defines biomedically meaningful nominal concepts. Figure 8.2 shows the GENIA ontology, where concepts are classified in a hierarchy. Note that the terminal concepts are presented in bold boxes. They define the terms that need to be identified from the literature, and become the target of annotation. The figures next to the labels of terminal concepts indicate their frequency of occurrence in the GENIA corpus, version 3.01.

Such biomedical terms as entity names are often compared to named entities in the newswire domain, such as names of humans, organizations, and so forth. From a linguistic perspective, however, they are quite different. Named entities are mostly proper nouns, which means they rarely appear with specifiers or qualifiers [11][5] preceding them. On the other hand, biomedical terms are mostly general nouns that often appear in text with a variety of specifiers or qualifiers. The syntactic definition of GENIA terms, which is given in Figure 8.3, states that the term expression may include preceding qualifiers, but not specifiers.

5. Specifiers include ordinals, cardinals, and determiners. Qualifiers include adjectives and noun modifiers. We follow Allen's [11] definitions.

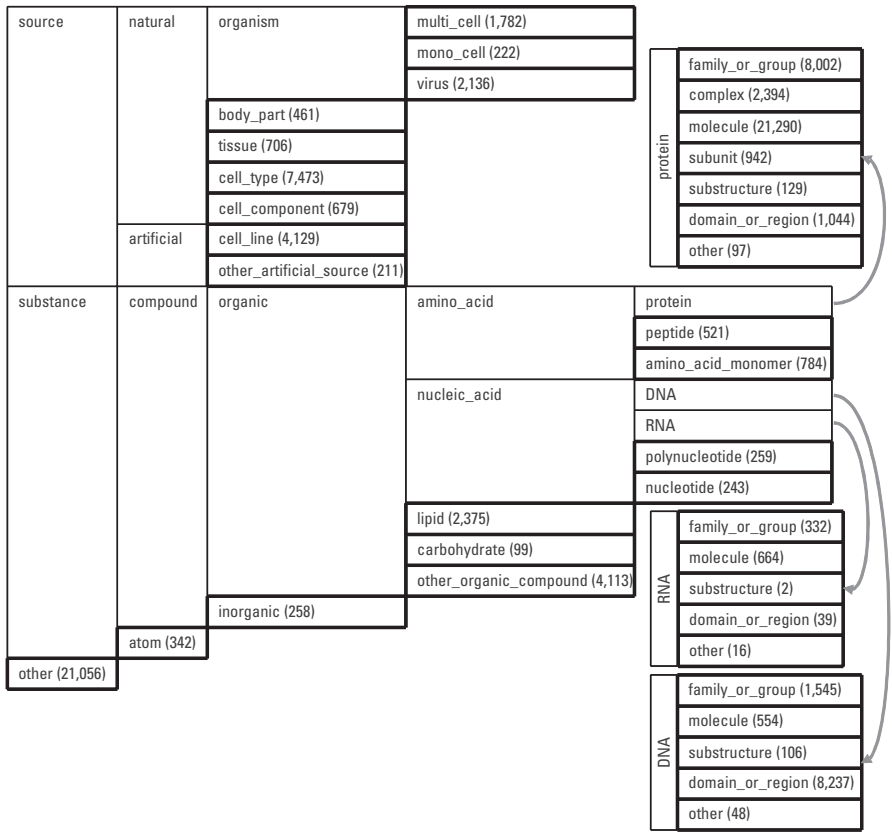


Figure 8.2 GENIA ontology and statistics from the GENIA corpus.

<term> := <qualifier> * <head noun>
<qualifier> := <adjective> | <noun modifier>

Figure 8.3 Syntactic definition of GENIA terms.

The semantic and syntactic definitions still leave much room for arbitrary decisions by annotators. Annotation guidelines need to be prepared to reduce the potential for inconsistent annotation. In the case of GENIA term annotation, the classes of terms to be annotated are defined by the GENIA ontology (e.g., 'protein molecule'), but the specific terms are not (e.g., *interleukin-2*). The guidelines need to include descriptions concerning whether or not specific terms are to be annotated. Sometimes the decision depends on specific contexts, about which proper descriptions also need to be included in the guideline. Problematic examples include the following:

1. classII-positive B cell:
   classII-positive B cell versus *B cell*
2. IL-2 receptor:
   whether to annotate *IL-2* or not

With regard to the first example, *classII-positive B cell*, it needs to be decided whether the longer expression, *classII-positive B cell*, or the shorter one, *B cell*, is to be annotated. This decision is related to the granularity of concepts to be identified from the text, and will depend on the purpose of annotation. In the case of GENIA, the longer expression is annotated following a policy of "more specific concepts."

In the second example, *IL-2 receptor*, two textual expressions will be considered for annotation: *IL-2* for one kind of protein molecule, and *IL-2 receptor* for another kind. *IL-2 receptor* may be rather safely annotated, while the question of whether or not to annotate *IL-2* will depend on the policy. In the case of GENIA, *IL-2* is not annotated following a policy of "mentioned substance only." In this example, the mentioned substance is only *IL-2 receptor*, and the appearance of *IL-2* in the textual expression is due to the naming of the substance. A similar policy is also applied to the annotation of GENETAG.

Sometimes, even the application of policies as explained above is not clear, and the guidelines need to provide comprehensive examples and recommended decisions. It is no wonder that annotation guidelines often become very long. Another problematic aspect of annotation guidelines is that these guidelines hardly can be prepared before starting the actual annotation work, but instead will grow gradually as the annotation continues. Consequently, the annotation that has already been made also needs to be continuously updated to reflect any changes in the guidelines.

PennBioIE provides more focused annotation for a smaller number of terms. The base texts of the corpus consist of abstracts from two domains: the oncology domain and the CYP domain. The text set from the oncology domain is annotated for genes, variation events, and malignancies; and the text set from the CYP domain is annotated for CYP450 enzymes, other substances, and quantitative measurements. It may be expected that this simplification of the annotation scheme would reduce the complexity of the annotation work. However, the PennBioIE team also maintains a long list of guidelines for the entity level of annotation. It is notable that the guidelines are maintained in two versions. One version is for previously released annotation, and thus is frozen, while the other version is for ongoing annotation, and thus is evolving.

GENETAG was originally developed for the *BioCreAtIvE* task 1A competition [12], to provide a gold standard to which automated systems for gene and protein name identification were compared. Later, it was updated to

GENETAG-5, and combined into the MedTag corpora [13]. It provides gene and protein name annotations for a wide range of organisms and biomedical contexts (e.g., molecular biology, genetics, biochemistry, clinical medicine, and so forth). MedTag also contains the ABGene corpus, which provides annotations for protein/gene names.

A simplified version of the GENIA corpus, containing annotations only for proteins, DNAs, RNAs, cell lines and cell types, was also used as a gold standard for the evaluation of automated bioentity recognition systems competing at the JNLPBA workshop [14].

Franzén et al. [4] also have produced a protein annotated corpus. Using this, they developed and evaluated a protein name tagger called Yapex. The corpus consists of 99 abstracts for development, and 101 for evaluation. These were retrieved from MEDLINE using the following query: "*protein binding*" [MeSH] AND "*interaction*" [MeSH] AND "*molecular*" [MeSH]. Fifty-three of the abstracts in the evaluation set also are found in the GENIA corpus.

Table 8.6 lists the corpora and their annotation for entities available in the domain of biomedicine.

8.4.2 Annotation for Biological Processes

Examples in the previous section show that annotation criteria for terms vary, depending on different views or tasks that the annotation groups consider. Such difficulties become more conspicuous when we annotate biologically significant

Table 8.6
Corpora and Their Annotation for Entities

Corpus	Type of Annotation (Size)	Remarks
GENIA	Term (2,000 abstracts)	Terminal concepts in GENIA ontology
GENIA-JNLPBA		Proteins, DNAs, RNAs, cell lines, cell types
PennBioIE-Oncology	Entity (1,157 abstracts)	Genes, variation events, malignancies
PennBioIE-CYP	Entity (1,100 abstracts)	CYP450 enzymes, other substances, quantitative measurements
GENETAG-05 (MedTag)	Entity (15,000 sentences)	Gene/protein names
ABGene (MedTag)	Entity (4,265 sentences)	Gene/protein names
Yapex	Entity (200 abstracts)	Protein names

processes or relations. Establishing annotation criteria for biological processes or relations is thus much more precarious.

The Caderige corpus [5] is notable for including annotations for interactions between biomedical entities, making it applicable to IE tasks. IE addresses the problem of extracting useful information from texts (e.g., to complete predefined information templates). Generally, IE is regarded as a basic task of text mining. From a linguistic perspective, the annotation in Caderige is exceptionally well crafted. The annotation is encoded directly into the text, revealing the type, the agent, and the target of interactions.

The GENIA team has tested the annotation scheme of Caderige with a small portion of the GENIA corpus. A careful analysis of the preliminary annotations suggests that there are potentially two ways of interpreting identical processes. When processes are interpreted in terms of interactions, which is the case with Caderige, pairs of entities that participate in the corresponding interactions are the focus of attention. Focusing on the procedural aspect of processes can make another interpretation. The following example demonstrates various problems that occur regarding interaction annotation.

> "The PKC-dependent **activation** of ERK were sensitive to **inhibition** by forskolin."

The example includes references to three entities that are underlined, and to two procedures that are in bold. With regard to the example, an annotator who focused on interactions recognized the following interactions, and made annotations as such:

> PKC —(**activates**)→ ERK
> forskolin —(**inhibits**)→ ERK

Another annotator who focused on the procedure recognized the following processes, and made corresponding annotations:

> PKC —(**activates**)→ ERK
> forskolin —(**inhibits**)→ activation

Note that there was a difference in finding the target of the inhibition process. The annotator for interactions marked *ERK*, which is the beneficiary of the process, as the target of the process. On the other hand, the annotator for procedure marked *activation*, which is actually what is inhibited. Figure 8.4 illustrates the difference in the two interpretations when the processes are hierarchical.

It is hard to say which interpretation is better, but it may depend on the purpose of annotation, the topics of the text, and so forth. One also may

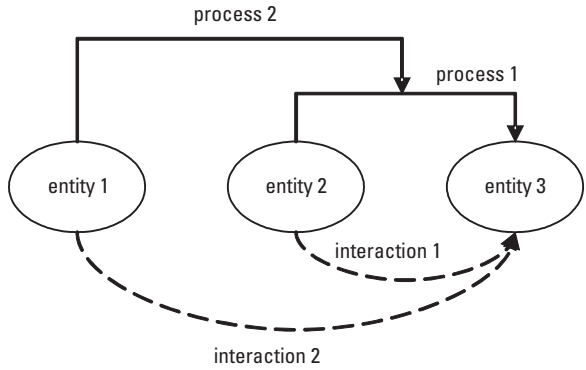


Figure 8.4 Identification of interactions and processes.

imagine that some texts will be written with a clear intention to explain procedural aspects of biological processes. In that case, one might expect that a procedural interpretation would work better. Similarly, a text focusing on interactions may benefit from the alternative interpretation.

8.4.3 Annotation for Linguistic Structure

Although the linguistic structure of text, such as the phrasal or dependency structure, may not be the main interest of text mining practitioners, they often study it to improve their text mining systems. It is generally accepted that information about the linguistic structure of text is helpful in accessing the information encoded in the text. Knowing about the linguistic structure of text is like having a map of a mine's topography, showing paths and suggesting potential places to dig for pieces of knowledge.

At the fundamental level, linguistic annotation includes sentence segmentation, word tokenization, POS labeling, and so forth, which may be linearly crafted; while higher level annotation includes treebanking [7], propbanking [15], and so forth, which usually involve hierarchical analysis.

Tokenization and POS labeling often are regarded as the first step of NLP processing, to determine the basic units of a sentence and their properties (e.g., grammatical or syntactic identity). Figure 8.5 shows an example of a sentence that has been tokenized and POS-labeled. Note that punctuation and parentheses are usually split from adjoining words to make separate tokens.

However, as Tateisi and Tsujii [16] point out, even with tokenization, annotators may experience difficulties that are caused by the nature of biomedical literature. Such scientific writing inherently will contain many chemical and numerical expressions, for which completely different languages are used. Examples include *1,25(OH)2D3*, *beta-(1,3)-glucan*, *t(3;3)(q21;q26)*, and so forth. To

```
<sentence><w c="NN">Type</w> <w c="CD">II</w> <w c="NN">corticosteroid</w>
<w c="NNS">receptors</w> <w c="VBP">are</w> <w c="RB">not</w>
<w c="VBN">reduced</w> <w c="IN">by</w> <w c="NN">excess</w> <w c="IN">of</w>
<w c="JJ">endogenous</w> <w c="NNS">corticosteroids</w>
<w c="LRB">(</w><w c="NN">Cushing</w><w c="POS">'s</w>
<w c="NN">syndrome</w><w c="RRB">)</w><w c="PERIOD">.</w></sentence>
```

Figure 8.5 Example of sentence segmentation, tokenization, and POS labeling.

avoid chaotic results, such expressions need to be isolated from usual tokenization, forcing them into single tokens.

Treebanking reveals the syntactic structure of sentences, while propbanking finds predicates and their arguments. Figure 8.6 illustrates the syntactic structure (with solid lines) and the predicate-argument structure (with dotted lines) of a sentence.

GENIA and PennBioIE are currently the only corpora of biology texts providing manually curated linguistic annotations made at various levels. Both provide annotations for sentence segmentation, tokenization, POS labeling, and treebanking. PennBioIE also provides annotations for paragraph segmentation. The tokenization, POS labeling, and treebanking for each corpus follow the Penn Treebank II (PTB) annotation guidelines [17], with minor modifications of each other [6, 16]. The MedPost corpus, included in MedTag [13], covers the domain of molecular biology and clinical medicine, and provides annotation for tokenization and POS labeling.

Another dimension of linguistic annotation that has been worked on is coreference, a term describing the situation in which two expressions referring to the same item are linked together. Castano et al. [18] point out that identifying the local arguments of biological relations may not offer enough information for determining the actual entities involved in the relation. Their product, the

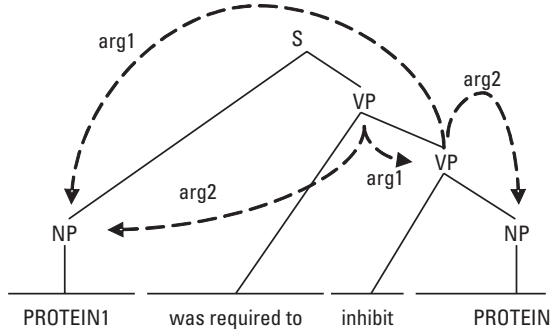


Figure 8.6 Example of syntactic and predicate-argument structure.

MEDSTRACT corpus (http://medstract.org/), provides annotations for anaphora and acronym resolution. The MEDCo corpus (http://nlp.i2r.a-star.edu.sg/medco.html) consists of the GENIA portion of MEDLINE abstracts, and provides extensive annotation covering various kinds of coreference. Examples include the following:

- Anaphora: *TCF-1 could transactivate through its cognate motif …*
- Spelling variation: *The rate of transcription initiation difected by the long terminal repeat (LTR) of HIV-1 increases …* and *Here we show that the response of the HIV-1 LTR may be governed by …*
- Relative pronoun: *a sequence GGGGACGTCCCC, named B2, which is similar to the kappa B sequence …*
- Appositive: *TCF-1, a T lymphocyte-specific transcription factor containing a sequence-specific HMG box …*
- Part-whole: … *cytoplasmic retention of either p65* or c-Rel … *both are …*

To provide annotation guidelines, the MedCo team took a set of guidelines that had been prepared for the newswire domain (MUC-7) (http://www.itl.nist.gov/iaui/894.02/related_projects/muc/proceedings/co_task.html), and made modifications to reflect differences between the newswire domain and the biomedical domain. One of the major differences is that the entities of interest are mostly instantiated ones, in the case of the newswire domain (e.g., a particular person), while they are often conceptual classes (e.g., a particular type of protein), in the case of the biomedical domain.

Table 8.7 lists corpora, and their annotation for linguistic structure, which are currently available in the domain of biomedicine.[6]

8.5 Issues on Manual Annotation

This section highlights the issues that need to be considered when designing a corpus.

8.5.1 Quality Control

The reliability of manually curated annotation affects the reliability of the development or evaluation of information processing systems that are based on the annotation. The degree of agreement between different instances of annotation

6. The data in Tables 8.1 and 8.2 were surveyed in May 2005.

Table 8.7
Corpora and Their Annotation for Linguistic Structure

Corpus	Type of Annotation
GENIA	Sentence, tokenization, POS (2,000 abstracts), treebank (200 abstracts)
PennBioIE-Oncology	Paragraph, sentence, token, POS (1,157 abstracts), treebank (318 abstracts)
PennBioIE-CYP	Paragraph, sentence, token, POS (1,100 abstracts), treebank (325 abstracts)
MedPost (MedTag)	Token, POS (6,700 sentences)
MedStract	Anaphora (32 abstracts), acronym (288 abstracts)
MEDCo	Coreference (228 abstracts)

for the same text is often considered to be a useful indicator of the reliability of annotation. The agreement between different annotators (inter-annotator agreement) allows conclusions about the stability of annotations, while the agreement for each annotator with himself (intra-annotator agreement) indicates the reproducibility of annotations [19].

Both inter- and intra-annotator agreement can be calculated simply by the absolute agreement rate, or, more popularly, by Cohen's Kappa coefficient [20, 21]. The absolute agreement rate is the proportion of agreement, as given in (8.1).

$$P(A) = \frac{\text{number of times annotators agree}}{\text{number of items to annotate}} \tag{8.1}$$

The Kappa coefficient is the proportion of agreement, corrected for expected agreement by chance, as given in (8.2).

$$K = \frac{P(A) - P(E)}{1 - P(E)} \tag{8.2}$$

where $P(E)$ is the expected proportion of agreement by chance. Kappa equals 1 when there is complete agreement. When the absolute agreement exceeds the chance agreement, Kappa is positive, with a magnitude reflecting the strength of agreement. Kappa may be between −1 and 0 when the absolute agreement is less than the chance agreement.

Let us demonstrate the calculation with a simple example. Suppose two annotators are given a list of 10 protein names, and asked to look for abbreviations in the list. The results are given in Table 8.8.

Table 8.8
Example of Annotation to Find Abbreviations

Protein Name	1	2	3	4	5	6	7	8	9	10
Annotator1	F	A	F	F	F	F	F	F	F	A
Annotator2	F	A	F	A	F	F	A	F	F	F

Note: A = abbreviation, F = full name

In the table, names considered to be abbreviations are marked with an "A." The two annotators agree in 7 of the 10 protein names, yielding 70% of absolute agreement rate. With the results, we can create a contingency table comparing their annotation results, as shown in Table 8.9.

Now, we see that Annotator1 answered negatively ("F") in 80% of the protein names, and Annotator2 answered negatively in 70%. This means that, for any given protein in the list, they will both simultaneously state an "F" with a 56% probability, and will both state an "A" with a 6% probability. This yields a total agreement of 62%, purely by chance. The Kappa coefficient is calculated to be 21%, following (8.2).

Landis and Koch [22] characterize the ranges of Kappa, as shown in Table 8.10. Although the divisions are determined rather arbitrarily, they often become benchmarks when discussing the agreement rate of annotation [19]. According to the table, the Kappa coefficient of the example task described above indicates only "fair" agreement.

With respect to annotations made for corpora in the biomedicine domain, only a few studies have been reported with inter-annotator agreement, and none with intra-annotator agreement. Tateisi and Tsujii [16] reported interannotator agreement, measured in two stages, with regard to POS-tagging in GENIA. In the first stage, they followed the Penn Treebank II Guidelines, resulting in 86.7% agreement (Kappa coefficient). A careful analysis was then carried out on

Table 8.9
Contingency Table of Annotation Example

		Annotator1		
		A	F	Total
Annotator2	A	1 (0.06)	2 (-)	3 (0.30)
	F	1 (-)	6 (0.56)	7 (0.70)
	Total	2 (0.20)	8 (0.80)	10

Table 8.10
Classification of the Kappa Coefficient

Kappa Coefficient	Strength of Agreement
<0.00	Poor
0.00–0.20	Slight
0.21–0.40	Fair
0.41–0.60	Moderate
0.61–0.80	Substantial
0.81–1.00	Almost Perfect

the result, to determine which portions of the guidelines are not well-suited to annotating the corpus. In the second stage, they improved the agreement rate to 98.5% with the revised guidelines.

Parkhomov et al. [23] reported 93.44% interannotator agreement (Kappa) on POS-tagging for clinical notes. Their work also includes an interesting report on agreement rate between annotators trained in different domains. They let their annotators, who had been trained for the clinical domain, annotate a small portion of the Penn Treebank corpus, and then measured the agreement rate. They reported an average 87.95% absolute agreement.

With regard to the task of entity annotation, the number of items to annotate is generally unknown, leading to a different number of annotations from different annotators. In such a case, one of the annotators is usually assumed to be authoritative, and the number of annotations by this annotator is determined to be the correct number of items to annotate. In this case, the absolute annotation rate effectively means recall, in the terminology of information science. That is, the rate of recovered information by another annotator is compared to that of the annotation of the authoritative annotator. When the work of two annotators is being compared, the authoritative annotator is sometimes rotated, and the harmonic mean of the resulting two versions of recall is taken. The result is known as the "F-score." Morgan et al. [24] report an F-score of 87% concerning the inter-annotator agreement in their annotation for Drosophila genes.

The issue of reliable annotation has been studied thoroughly in the speech community. Gut and Bayer [19] pointed out that the quality of manual annotation has been criticized in the following ways:

- *Implicit incoherence:* The manual labeling procedure is incoherent due to human variability in perceptual capabilities and other factors. Intra-annotator reliability can never be perfect.

- *Lack of consensus on coding schema:* Manual annotations reflect the variability of the interpretation and application of the coding schema by the annotators.
- *Annotator characteristics:* Individual characteristics of coders, such as familiarity with the material, amount of former training, motivation and interest, and fatigue-induced errors, influence the quality of annotations.

They also present an extensive evaluation of annotator reliability in a multilevel, phonetically annotated speech corpus. The results indicate that the reliability of manual annotation is mostly influenced by the complexity of the annotation task. They also indicate that the number of annotation schema categories that lead to confusion among annotators is often relatively small, so that an improvement of annotation reliability can be achieved fairly easily, by carrying out systematic error analyses and changing the annotation schema accordingly. Tateisi and Tsujii's work [16] is an example of this case.

8.5.2 Format of Annotation

The format of annotation is receiving increasing attention as the annotated information of interest becomes more complex (i.e., moving from entities to events). This is also related to the reusability and exchangeability of a corpus, which are natural requirements for such expensive resources. Manual annotation is time-consuming and labor-intensive work.

In the natural language processing community, the tabular format (e.g., Susanne corpus), and the Brill tagger format (e.g., Brown corpus) have been widely used. In the tabular format, each token takes up one sequential line, and annotation for each token is written on the corresponding line as tab separated values (TSV). In the Brill tagger format, tokens are separated by single space characters, and annotation is attached to each token with slash ('/') characters as the delimiter. When the annotation is simply described by single labels, the Brill tagger format is equivalent to the tabular format, with space characters corresponding to new line characters, and slashes to tab characters. These simple formats are easy to implement, but have clear limitations, since only linear information may be encoded. Both formats are often criticized for contaminating the base text, and corrupting the word boundary information.

The shared task version of GENIA is encoded in the tabular format, and GENETAG is encoded in the Brill tagger format. This is partly because the formats are familiar to machine learning practitioners, and partly because the annotated information is simple enough to be encoded in the simple formats.

Extensible Markup Language (XML) is an emerging standard for a general encoding framework, which also has been applied to encode a number of corpora and their annotations. While the tabular or Brill tagger formats are suitable to encode a linear information structure, XML is inherently suitable for a tree-like information structure, which is often assumed to be the basic structure of language. One of the major benefits of using XML as a corpus encoding framework is the ease of validation. XML provides an easy way to define an encoding scheme (referred to as a document model), and to validate a document (referred to as a document instance), which ensures that the document represents the encoding scheme.

The most common way to define an encoding scheme is with a document-type definition (DTD). Figure 8.7 shows a sample DTD for a corpus with annotation for proteins. It defines three elements (tags): **set**, **sentence** and **protein**. From the beginning, the DTD states that an XML document conforming to it may have a **set** element, containing one or more **sentence** elements; a **sentence** element may contain character data or **protein** elements; again, a **protein** element may contain character data, or **protein** elements (recursively); for each **protein** element, the **sem** attribute must be specified with a value of 'molecule', 'complex', or 'family'; and for each **protein** element, the **lex** attribute may be (optionally) specified with character data.

Figure 8.8 shows a sample corpus conforming to the DTD. It consists of a set of sentences with annotation for proteins. Each protein identified is classified into the protein molecule, family, or complex, as specified by the **sem** attribute. The **lex** attribute is set to the character string indicating the lexical information. Bold type has been used here to enhance readability.

Standoff annotation is a rather new trend in encoding strategy. The idea here is to store the base document and the annotation separately. See Figure 8.10 for an example. The main advantages of the standoff annotation model have been described as follows [25]:

1. The base material may be read-only and/or very large.

```
<!-- A Simple DTD for Protein Annotation     -->

<!ELEMENT    set          (sentence+)>
<!ELEMENT    sentence     (#PCDATA | protein)*>
<!ELEMENT    protein      (#PCDATA | protein)*>
<!ATTLIST    protein
                          sem  (molecule | complex | family)    #REQUIRED
                          lex   CDATA                           #IMPLIED
>
```

Figure 8.7 A simple DTD for protein annotation.

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/css" href="protein.css"?>
<!DOCTYPE set PUBLIC "-//TMBOOK//DTD Protein Annotation 0.1//EN" "protein.dtd">

<set>
<sentence>Activation of the<protein lex="CD28_surface_receptor" sem="family"><protein
lex="CD28" sem="molecule">CD28</protein>surface receptor</protein>provides a
major costimulatory signal for T cell activation resulting in enhanced production of
<protein lex="interleukin-2" sem="molecule">interleukin-2</protein>(<protein lex="IL-2"
sem="molecule">IL-2</protein>) and cell proliferation.</sentence>
...
</set>
```

Figure 8.8 A sample corpus with annotation for proteins.

2. The annotation may involve multiple overlapping hierarchies.
3. Distribution of the base document may be controlled (e.g., by copyright condition), but the annotation is intended to be freely available.

There is an effort (ISO/TC 37/SC 4/WG 1-1) to establish an international standard for a linguistic annotation framework [26] that is based on standoff annotation and XML.

Standoff annotation offers a way to overcome the limitations of inline annotation, but it lacks stable linking mechanisms to connect the annotation and its corresponding text. For the third point above to make sense, the base must be stored permanently in an accessible place. When the base document is owned by other individual(s), as is the case of the first point, it may be changed or removed unexpectedly, making some links invalid. The second point is also criticized in that, by removing the nonoverlapping restriction, annotators will create not only meaningful but also erroneous overlapping. The choice for standoff annotation is therefore subjective, depending on the application.

GENIA is encoded in an inline XML format, and comes with corresponding DTDs. PennBioIE is encoded in a standoff XML format due to their annotation tool, WordFreak, which is discussed in Section 8.6.2. The MedTag corpora provide standoff annotations stored in a relational database, so that they can be easily searched.

Automatic conversion between different encoding schemes is another idea to raise the potential of exchangeability. Erjavec et al. [27] present a fully automatic conversion of GENIA into Text Encoding Initiative (TEI) format (http://www.tei-c.org/). The TEI offers a well-designed and widely accepted general architecture, which has often been used for linguistic annotation. It is expected that porting GENIA to TEI format will make the corpus better suited for interchange.

8.5.3 Discontinuous Expressions

Separated or unconnected expressions sometimes need to be annotated together. Some examples of such discontinuous expressions, followed by their meanings (as integrated forms), are given below. The integrated forms of discontinuous expressions are underlined.

- *cytokeratins 8 and 18* : *cytokeratins 8 and cytokeratins 18*
- *PKC-alpha, -epsilon, or –zeta* : *PKC-alpha, PKC-epsilon, or PKC–zeta*
- *erythroid and/or megakaryocytic lineages* : *erythroid lineages and/or megakaryocytic lineages*
- *neither LMP1 nor LMP2B mRNA* : *neither LMP1 mRNA nor LMP2B mRNA*
- *c-jun but not c-fos mRNA* : *c-jun mRNA but not c-fos mRNA*
- *gamma-interferon- as well as glucocorticoid response elements* : *gamma-interferon-response elements as well as glucocorticoid response elements*

All the above examples are all coordinated clauses involving ellipsis, which is the most common case of discontinuous expressions. To annotate such expressions, GENIA takes a compositional approach, where it is assumed that the meaning of a coordinated clause may be incrementally constructed by using the fragments inside it.

Figure 8.9 illustrates the concept of this approach with an example clause: *neither LMP1 nor LMP2B receptors*. First, the spans of text in the term composition (*LMP1*, *LMP2B*, and *receptors*, in this case) are identified. The rest are considered as a linguistic template, which decides the structure of the clause. Second, the fragments, *LMP1* and *LMP2B*, are coordinated by the template 'neither X nor Y,' composing the meaning '(NEITHER_NOR 'LMP1' 'LMP2B')'. Third, the remaining fragment *receptors* is distributionally concatenated to the coordinated construct, yielding the meaning '(NEITHER_NOR 'LMP1 receptors' 'LMP2B receptors')'. Finally, the two composed expressions, 'LMP1 receptors' and 'LMP2B receptors', are identified as RNA molecules, yielding '(NEITHER_NOR G#RNA_molecule G#RNA_molecule)' as the semantic meaning of the entire clause. The prefix 'G#' indicates that the semantic class, 'RNA_molecule', is defined in the GENIA ontology. With this compositional process in mind, the actual annotation of GENIA has been minimized to reduce the burden of manual annotation, by skipping the intermediate level of annotation (indicated in gray in the figure), and leaving only the annotation for the entire clause and the fragments of textual term expressions.

```
<cons sem="(NEITHER_NOR G#RNA_moleculeG#RNA_molecule)"
      lex="(NEITHER_NOR 'LMP1 receptors' 'LMP2B receptors')">
          <cons lex="(NEITHER_NOR 'LMP1''LMP2B')">
                    neither
                    <cons lex="LMP1">LMP1</cons>
                    nor
                    <conslex="LMP2B">LMP2B</cons>
          </cons>
          <conslex="receptors">receptors</cons>
</cons>
```

Figure 8.9 Compositional approach for discontinuous expressions: case of GENIA corpus.

While the compositional approach suggests a gentle way of annotating discontinuous expressions without messing up the underlying tree structure, it may be criticized in two ways. First, it assumes a separate layer of processing for lexical and semantic computation, which may have a high cost. Second, the textual parts of each term expression are not explicitly indexed.

The PennBioIE corpus adopts a chaining mechanism to annotate discontinuous expressions [6]. Figure 8.10 shows an example of annotation for discontinuous expressions using chaining. In the example, for the clause *P1-450, P2-450, and P3-450 proteins*, one continuous expression (*P3-450 proteins*) and two discontinuous expressions (*P1-450* + *proteins* and *P2-450* + *proteins*) are annotated as 'cyp450'. For the continuous expression, *P3-450 proteins*, a single *Annotation* tag is created and annotated as 'cyp450'. For the discontinuous expression *P1-450* + *proteins*, both text spans are annotated separately as 'cyp450'. Their connection is recorded in an additional *Chain* tag listing their annotation-ID numbers, which are unique within each annotation file.

The pros and cons of this approach are the converse of those of the compositional approach. It does not require additional high-level analysis, except for the chaining, which is rather cheap to implement. However, the chaining may contaminate the tree structure, which may cause conflicts when integrated with higher level annotation (e.g., treebanking).

8.6 Annotation Tools

Corpus annotation is expensive work, usually involving extensive time and labor. It is important for the annotators to be equipped with a suitable suite of tools. Examples include visualization tools, editors, workflow management tools, and so forth.

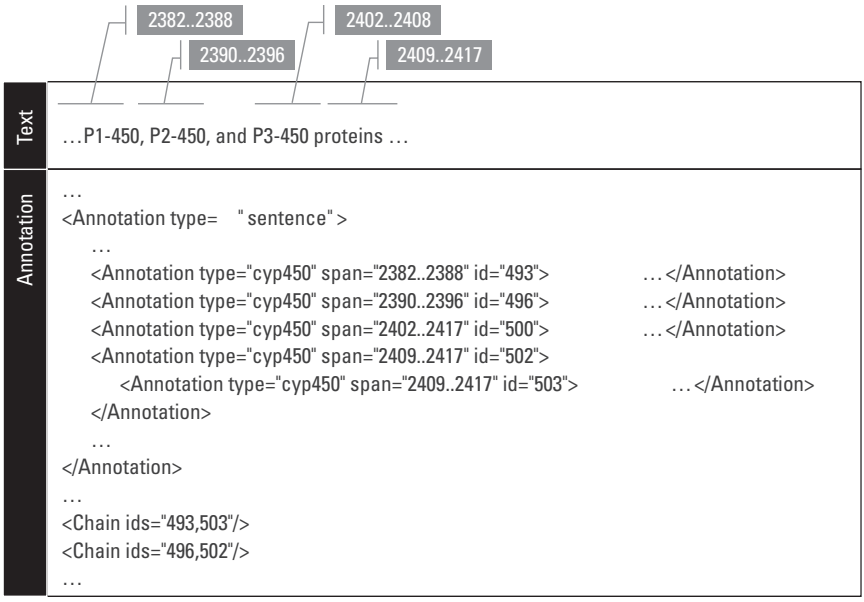


Figure 8.10 Chaining for discontinuous expressions: case of PennBioIE corpus.

8.6.1 Reuse of General Purpose Tools

If the corpus we are going to work with is encoded in XML, it is most likely that we already have numerous tools to deal with the corpus. There are plenty of tools supporting XML, and there is a high chance that tools already exist that meet our purposes. In fact, this is one of the attractions of choosing XML as the encoding framework. For example, many Web browsers now support XML parsing, and can generate styled views, so they can be used to offer customized views of a corpus.

Figure 8.11 is a simple stylesheet prepared for the corpus shown in Figure 8.8. According to it, all the sentences are to be shown using a font of size 10 point, **protein** elements with the **sem** attribute set to 'molecule' are to be bold, and **protein** elements with the **sem** attribute set to other values are to be underlined and colored. The stylesheet is written as a Cascading Style Sheet (CSS), currently the most frequently used style description language for XML documents, recommended by the World Wide Web Consortium (W3C) (http://www.w3c.org/). Figure 8.12 shows the styled view of the corpus rendered by Mozilla (http://www.mozilla.org/), a popular XML and HTML rendering engine that has been integrated in many Web browsers.

The GENIA team is utilizing general purpose tools, instead of creating their own tools, to develop the GENIA corpus. Their development environment

```
/* A Simple CSS for Protein Annotation */

sentence                {font-size: 10pt;}
protein[sem="molecule"] {font-weight: bold;}
protein[sem="family"]   {text-decoration:underline; background-color: cyan;}
protein[sem="complex"]  {text-decoration:underline; background-color: coral;}
```

Figure 8.11 Simple CSS for protein annotation.

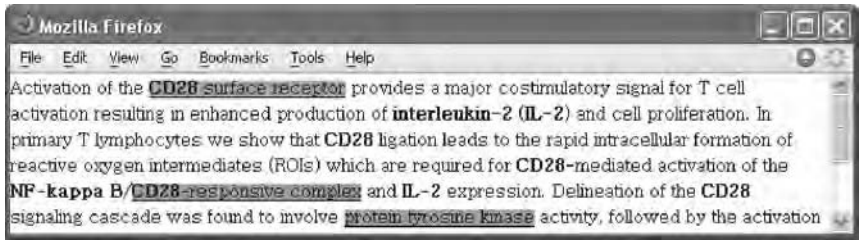


Figure 8.12 Styled view of corpus rendered by a Web browser.

is constructed on top of Eclipse (http://www.eclipse.org/), a widely used software development platform. It has open architecture for plug-ins, and there are thousands of plug-ins developed in industry and academia. Among the plug-ins, the VeX XML editor (http://vex.sourceforge.net/) is used to edit the corpus files. VeX is a general purpose XML editor, providing a word processor–like view. Not only does the editor show a CSS-styled view like some Web browsers, but it also lets users make changes to what they are seeing. It also self-customizes the editing interface, according to a specified DTD, so that erroneous changes are automatically prohibited.

Figure 8.13 shows the VeX editor plug-in for Eclipse. The sample corpus shown in Figure 8.8, together with the corresponding DTD and CSS files shown in Figures 8.7 and 8.11, are given to the editor. The editing area looks very similar to the view of the Web browser above, except that it displays a caret (a blinking vertical bar), indicating the location where changes can be made. Note that the caret is now located in the middle of the string, *IL-2*, in the third line. The span of text is tagged as a 'protein' element (see Figure 8.8), which also can be seen in the editor, since the string is bold. The properties pane, located to the right of the editing area, shows the properties (attributes) of the element. The **lex** attribute is now set to IL-2, and the **sem** is 'molecule'. The properties pane also lets users change the property values. Following the DTD in Figure 8.7 that has been given to the editor, the value of **lex** may be directly edited and the value of **sem** may be chosen from the list of available values.

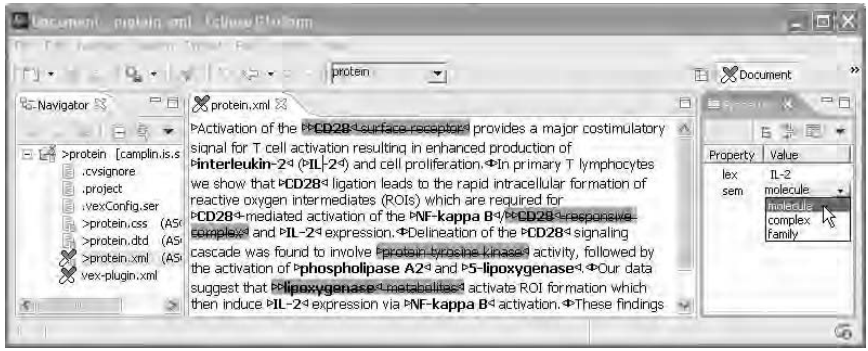


Figure 8.13 Editing an annotation with an XML editor.

Figure 8.14 shows another example of creating XML tags. The *T-lymphocytes* string has been selected (highlighted), and the "Insert" command has been chosen from the context menu. Note that only the 'protein' tag may be chosen. This is because the selected text span is inside a **sentence**, which may contain only characters or 'protein' elements according to the specified DTD (see Figure 8.7).

Another tool, which searches a number of text files for a pattern specified by the user, is the concordancer. It also shows a list of all instances matching that pattern. The GENIA team developed XConc, a concordancer for corpora encoded in XML. XConc is designed to be DTD-independent, so that it can be used with most XML-encoded corpora. Figure 8.15 shows XConc, together with VeX, plugged into Eclipse. Users may specify a pattern in the XConc Search pane, for instances matching a pattern to be listed in the XConc Search

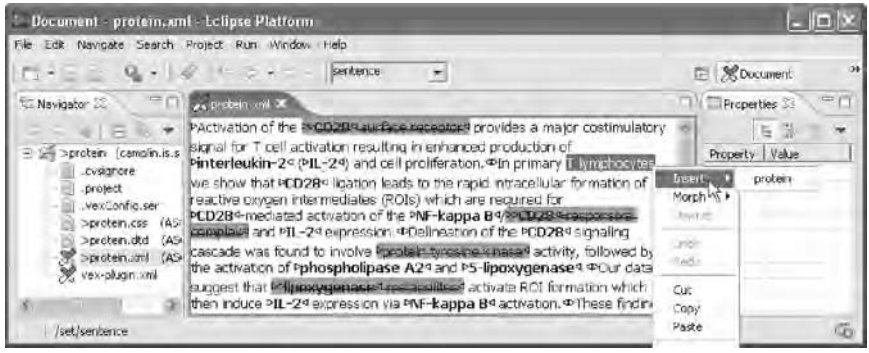


Figure 8.14 Making a new annotation with an XML editor.

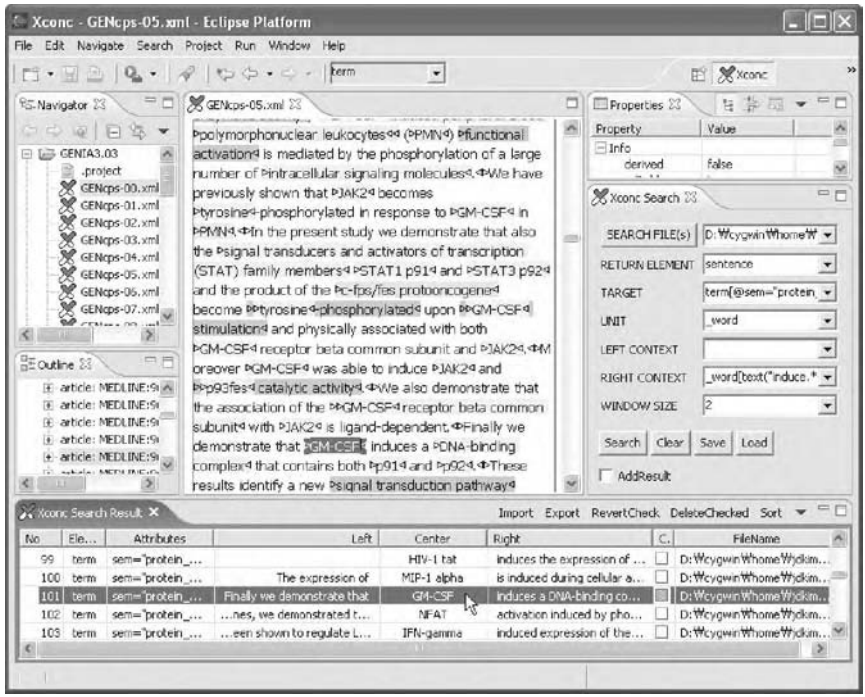


Figure 8.15 Concordancer and XML editor.

Research pane. The example in Figure 8.15 shows the results of the following search query:

```
TARGET=cons[@sem="protein_molecule"],
RIGHT=_word[text("activat.*")], UNIT=_word, WINDOW=2.
```

The query attempts to find **cons** elements with 'protein_molecule' as the value of the **sem** attribute, in a context where the word pattern "activat.*" is on its right side. For the context, it searches up to two words on either side. Basically, the syntax for pattern specification follows XPath, which is a sophisticated language for marking locations and selecting sets of nodes within an XML document, with an extension for string values represented as regular expressions. By clicking one of the instances in the result list (*GM-CSF* in the example), the user can execute VeX to open the file containing the instance. VeX will highlight the target string and locate the caret at the end of it.

Corpus annotation is usually a long-term project involving several annotators. In addition to the search and edit tools mentioned above, tools that support the management of work history and multiuser collaboration are of great use.

Concurrent Versions System (CVS) may be a straightforward option. It is a version control system that has emerged as a standardized platform for software development. It supports long-term projects, and makes it easy for developers to view a project's history at any point, and consequently correct problematic changes that have been made since the project started. CVS also supports multiuser collaboration, by detecting conflicts between the work of different developers, and reporting the conflicts in a convenient form. Eclipse supports the full functions of CVS, letting a user log on to a CVS server, check out tasks, commit their changes, and compare changes made by other users when conflicts arise.

The above examples suggest that, just by choosing standardized methods, we can save the effort that would otherwise be wasted to develop tools to replicate the methods. We do not need to paddle ourselves all the way to reach a goal. Instead, we could just ride on a current that is flowing in a similar direction to minimize the distance and effort.

8.6.2 Corpus Annotation Tools

There are many tools designed especially to support corpus annotation. Since these tools are committed to corpus annotation as the primary application, they are expected to provide more sophisticated functions than general purpose tools, at least for annotation work. They range from a simple editor to an integrated development environment (IDE) bundled with a bunch of related tools.

WordFreak (http://wordfreak.sourceforge.net/) is a linguistic annotation tool designed to support human and automatic annotation of linguistic data. It was developed by the PennBioIE team, originally to support its own annotation work. It supports tokenization and POS labeling, entity annotation, treebanking, and concordancing based on the standoff annotation scheme defined for PennBioIE. Furthermore, using plug-ins can extend the functionality. There are already several plug-ins developed for WordFreak to support a number of standardized projects. Users are recommended to check the list of existing plug-ins before writing their own. If there is no appropriate plug-in, then users can write their own plug-in to customize the interface and functionality, and to support their annotation scheme. The GENIA team also has developed a plug-in for WordFreak, which reads from and writes to the GENIA format, facilitating data exchange between PennBioIE and GENIA.

CADIXE is an XML editor developed by the Caderige team. It is DTD-aware, provides a styled view of XML documents by supporting a "CSS-like" stylesheet language, and is easily customizable. Unlike other XML editors, it allows the user to add tags without having any constraint on the insertion order, which may be useful for annotation work. It is used in the Caderige project and at the Swiss Institute of Bioinformatics (SIB) as part of the European project, BioMint.

There are also tools developed to explicitly support multiuser annotation. The LAW Workflow Management System developed by the PennBioIE team is intended to provide a general file/user/task management system for projects that need to manage a pipeline of processes that are performed on some type of source file. It is now supporting their annotation work, integrating multilevel linguistic information.

8.7 Conclusion

A corpus with appropriate good quality annotations often sets up a critical infrastructure for NLP research. Text mining research that is based on NLP technology also benefits from the existence of such a corpus. However, building such a corpus is expensive, requiring a long time and much manual collaborative work. Therefore, the related problems should be well studied to avoid waste of time and labor. Several topics concerning corpora in biology have been discussed. The focus is to introduce existing resources, and to describe known difficulties, with the purpose of delivering useful information on building corpora for text mining of biology texts. In this chapter, emphasis is placed on the GENIA corpus, which does not necessarily mean that this corpus is the most important. It is emphasized simply because the authors are involved in the construction of the corpus and can report on first-hand experience. For other resources, we include as many references as possible. Therefore, the reader is recommended to consult the corresponding citations for further detailed information.

Acknowledgments

We would like to thank our colleagues within the GENIA project, in particular, those who supported us during the production of this chapter: Tomoko Ohta, Yuka Tateisi, and Yoshimasa Tsuruoka.

References

[1] Cohen, K., et al., "Corpus Design for Biomedical Natural Language Processing," *Proc. ACL Workshop on Linking Biological Literature, Ontologies and Databases: Mining Biological Semantics*, 2005, pp. 38–45.

[2] Kim, J., et al., "GENIA Corpus—A Semantically Annotated Corpus for Bio-Textmining," *Bioinformatics*, Vol. 19, Suppl. 1, 2003, pp. i180–i182.

[3] Tanabe, L., et al., "GENETAG: A Tagged Corpus for Gene/Protein Named Entity Recognition," *BMC Bioinformatics*, Vol. 6, Suppl. 1, p. S3.

[4] Franzén, K., et al., "Protein Names and How to Find Them," *Medical Informatics*, Vol. 67, 2002, pp. 49–61.

[5] Alphonse, E., et al., "Event-Based Information Extraction for the Biomedical Domain: The Caderige Project," *Proc. COLING Workshop on Natural Language Processing in Biomedicine and Its Applications*, 2004, pp. 43–49.

[6] Kulick, S., et al., "Integrated Annotation for Biomedical Information Extraction," *Proc. NAAACL/HLT Workshop on Linking Biological Literature, Ontologies and Databases: Tools for Users*, 2004, pp. 61–68.

[7] Marcus, M. P., et al., "Building a Large Annotated Corpus of English: The Penn Treebank," *Computational Linguistics*, Vol. 19, No. 2, 1994, pp. 313–330.

[8] Scott, M., "Comparing Corpora and Identifying Key Words, Collocations, and Frequency Distributions through the WordSmith Tools Suite of Computer Programs," in *Small Corpus Studies and ELT: Theory and Practice*, M. Ghadessy, A. Henry, and R. L. Roseberry, (eds.), Amsterdam: Benjamins, 2001, pp. 47–67.

[9] Dunning, T., "Accurate Methods for the Statistics of Surprise and Coincidence," *Computational Linguistics*, Vol. 19, No. 1, 1993, pp. 61–74.

[10] Berber-Sardinha, T., "Comparing Corpora with WordSmith Tools: How Large Must the Reference Corpus Be?" *Proc. ACL Workshop on Comparing Corpora*, 2000, pp. 7–13.

[11] Allen, J., *Natural Language Understanding Second Edition*, Redwood, CA: Benjamin/Cummings, 1995, pp. 25–28.

[12] Yeh, A. S., et al., "BioCreAtIve Task 1A: Gene Mention Finding Evaluation," *BMC Bioinformatics*, 2005, Vol. 6, Suppl 1, p. S2.

[13] Smith, L. H., et al., "MedTag: A Collection of Biomedical Annotations," *Proc. ACL Workshop on Linking Biological Literature, Ontologies and Databases: Mining Biological Semantics*, 2005, pp. 32–37.

[14] Kim, J., et al., "Introduction to the Bio-Entity Recognition Task at JNLPBA," *Proc. COLING Joint Workshop on Natural Language Processing in Biomedicine and its Applications*, 2004, pp. 70–75.

[15] Gildea, D., and M. Palmer, "The Necessity of Parsing for Predicate Argument Recognition," *Proc. Annual Meeting of the Association for Computational Linguistics (ACL)*, 2002, pp. 239–246.

[16] Tateisi, Y., and J. Tsujii, "Part-of-Speech Annotation of Biology Research Abstracts," *Proc. Int. Conf. on Language Resource and Evaluation (LREC)*, 2004, pp. 1267–1270.

[17] Santorini, B., *Part-of-Speech Tagging Guidelines for the Penn Treebank Project*, Technical Report MS-CIS-90-47, Department of Computer and Information Science, University of Pennsylvania, 1990.

[18] Castano, J., et al., "Anaphora Resolution in Biomedical Literature," *Int. Symp. on Reference Resolution*, Alicante, Spain, June 3–4, 2002.

[19] Gut, U., and P. Bayer, "Measuring the Reliability of Manual Annotations of Speech Corpora," *Proc. Int. Conf. on Speech Prosody (SP)*, 2004, pp. 565–568.

[20] Cohen, J., "A Coefficient of Agreement for Nominal Scales," *Educational and Psychological Measurement*, Vol. 20, No. 1, 1960, pp. 37–46.

[21] Carletta, J. C., "Assessing Agreement on Classification Tasks: The Kappa Statistic," *Computational Linguistics*, Vol. 22, No. 2, 1996, pp. 249–254.

[22] Landis, J., and G. Koch, "The Measurement of Observer Agreement for Categorical Data," *Biometics*, Vol. 33, 1997, pp. 159–174.

[23] Pakhomov, S., A. Coden, and C. Chute, "Creating a Test Corpus of Clinical Notes Manually Tagged for Part-of-Speech Information," *Proc. COLING Joint Workshop on Natural Language Processing in Biomedicine and Its Applications*, 2004, pp. 62–65.

[24] Morgan, A., et al., "Gene Name Extraction Using FlyBase Resources," *Proc. ACL Workshop on Natural Language Processing in Biomedicine*, 2003, pp. 1–8.

[25] Thompson, H., and D. McKelvie, "Hyperlink Semantics for Standoff Markup of Read-Only Documents," *Proc. SGML Europe*, Barcelona, 1997, http://www.infoloom.com/gcaconfs/WEB/TOC/t0063_.HTM, accessed June 2005.

[26] Ide, N., L. Romary, and E. Clergerie, "International Standard for a Linguistic Annotation Framework," *Proc. HLT-NAACL Workshop on the Software Engineering and Architecture of Language Technology*, 2003, pp. 25–30.

[27] Erjavec, T., et al., "Encoding Biomedical Resources in TEI: The Case of the GENIA Corpus," *Proc. ACL Workshop on Natural Language Processing in Biomedicine*, 2003, pp. 97–104.

9

Evaluation of Text Mining in Biology

Lynette Hirschman and Christian Blaschke

9.1 Introduction

A growing number of groups are now working in text mining for biology. Two strands of research have converged on this problem: on the one hand, biology and bioinformatics, and on the other, computer science/natural language processing. The biology and bioinformatics research communities have felt an increasing need for improved tools to access and mine the rich textual information found in the biological literature, as well as the semistructured informa-tion in comment fields of the biological databases, such as GenBank (http://www.ncbi.nlm.nih.gov/Genbank/) or Swiss-Prot (http://us.expasy.org/sprot/). Computer science researchers, including people working in natural language processing, information retrieval, machine learning, and artificial intelligence, have long been interested in the general problem of accessing and mining textual information. What is new is the increased interaction among these communities.

On the natural language side, we have seen a series of workshops on biomedical text processing at the Association for Computation Linguistics (ACL), and Human Language Technology meetings since 2002 [1–3]. In addition, the Text Retrieval Conference (TREC) [4] started a new Genomics Track in 2003 [5, 6].

On the biology side, there have been Special Interest Group (SIG) meetings on text mining for biology at the Intelligent Systems in Molecular Biology (ISMB) conference since 2001 [7]. The Pacific Symposium on Biocomputing

(http://psb.stanford.edu) has held sessions on text mining for biology since 2000. There also have been joint activities, including a workshop at the University of Pennsylvania in February 2001 [8], workshops at the University of Tokyo in 2002 [9] and 2005, a number of workshops in Europe, such as the annual E-BioSci/Oriel workshops (http://www.e-biosci.org/), as well as workshops and sessions on language and biology in computational linguistics, such as [10].

However, despite the increased activity in this area, there are still relatively few commercial text mining tools deployed on a large scale. Moreover, until recently, it has not been possible to compare different approaches, because the various groups involved are addressing different problems, often using private datasets [11]. This is similar to the situation in text processing in the early 1990s, prior to the introduction of the MUCs. The first MUC was held in 1987, with six participating systems and no clear evaluation criteria. By the time of MUC-7 in 1998, there were four well-defined evaluation tasks, and 17 groups participating with many different systems. A few years after the start of MUC, the first TREC was organized in 1992, with the goal of creating realistic-sized document collections, in order to assess the state of the art in information (document) retrieval. TREC has now been held annually for the past 13 years, adding new tracks to reflect new research directions [4]. This includes tracks for answering questions, and the new Genomics Track, which attracted 33 participants in 2004.

These activities have made it possible to compare approaches to general document retrieval and understanding through the introduction of common evaluations, shared resources, and standardized metrics. This has allowed the research community to assess what techniques do and do not work, and to demonstrate the progress being made in the fields of information extraction and information access and retrieval.

A similar situation existed in areas of molecular biology in the 1990s. For example, in protein structure prediction, many groups were making claims about methods of prediction structure from primary protein sequence data. However, in the absence of shared datasets and standardized evaluation measures, it was not possible to compare these claims. Starting in 1994, the Critical Assessment of Protein Structure Prediction (CASP) evaluation began one of the earliest community-wide experiments to benchmark the state of the art of protein structure prediction [12]. CASP has now been running for more than a decade. CASP, in turn, served as a model for later initiatives. These include Critical Assessment of Microarray Data Analysis (CAMDA) [13], which assesses the performance of microarray bioinformatics tools, and Critical Assessment of PRediction of Interactions (CAPRI), which assesses protein interaction prediction techniques [14]. In addition, there has been an assessment for genome bioinformatics—the Genome Annotation Assessment Project (GASP) [15]—as well as the Genome Access Workshop (GAW) for statistical genetics techniques

[16], and the Predictive Toxicology Challenge (PTC) for computational toxicology approaches [17].

These histories illustrate that, once objective common evaluations become available, there is real eagerness on the part of the research communities to participate. We are now at a critical point in bringing the biology and bioinformatics community together with the computer science and natural language processing community. One outcome has been an increasing number of common challenge evaluations for text mining in biology, as well as resources and benchmark sets. The evaluations include:

- Knowledge Discovery and Data Mining (KDD) Challenge Cup 2002 Task 1 [18]: Document classification for FlyBase (http://flybase.bio.indiana.edu/), requiring identification of full text journal articles containing experimental evidence for specific genes and gene products. This evaluation was done under the auspices of the KDD Special Interest Group of the Association for Computing Machinery (ACM).
- TREC Genomics Track 2003, ongoing [5, 6]: Document retrieval and classification tasks for genomics. This evaluation is now a track of TREC, held on an annual basis.
- BioCreAtIvE 2003 [19]: Critical Assessment of Information Extraction in Biology, which focused on two tasks: gene mention identification and normalization, and functional annotation for genes using the GO [20].
- BioNLP [21]: Tagging of biological names in MEDLINE abstracts. This evaluation was done as part of the JNLPBA in 2004.

These evaluations are the subject of this chapter, which is organized as follows. In Section 9.2, we address the question of why to evaluate. We look at the stakeholders in a shared assessment, and argue that evaluation has many functions, including measuring progress, bringing together a community, and creating a shared infrastructure to support research. In Section 9.3, we talk about what to evaluate, and discuss applications of text mining in biology that can be used to develop biologically motivated tasks for assessment. In Section 9.4, we review the history of recent evaluations for text mining in biology and their results. Finally, in Section 9.5, we conclude with a look towards the future, and discuss where evaluation for text mining in biology should go.

We have chosen to focus on shared assessments, or challenge evaluations, in this chapter. These are the most visible, and often the most influential, assessments in a field. However, evaluation itself must be embedded in every piece of empirical work in the area of text mining. For each new approach or each new task, it is critical to assess how well the approach performs. Often this is difficult,

because valid assessments may require extensive infrastructure: annotated "gold standard" or ground truth data, or even user studies. Such data is often difficult or expensive to obtain. Common assessments can lead the way for a research community by creating a shared infrastructure that can drive a whole field forward.

9.2 Why Evaluate?

We now look at the first issue of why to evaluate. There are several aspects to this question that we now consider.

9.2.1 The Stakeholders

The first issue in evaluation is to identify the stakeholders. Who cares about the evaluation? Who will participate? Who will pay attention to the results? Who will pay for the evaluations, and why? In an interdisciplinary area, such as text mining applied to biology, there are various groups of stakeholders with different objectives:

- *The users.* In the case of biology, the biologists and bioinformaticians are the end users of the tools under development. Very importantly, they are also the people who have the problems and the biological datasets that are needed for realistic evaluations.
- *The technology developers.* In this case, the developers are the bioinformaticians and computer scientists who are creating the tools and systems to be evaluated. These are the people who will field systems to be evaluated. They may be from academia or commercial companies, which may affect their willingness to share insights and results.
- *The funders.* These are the people who fund the underlying research (e.g., biology, bioinformatics, natural language processing), and have an interest in demonstrating that their investment is paying off, and that the field is making progress. This group can be quite diverse, but, in general, they want to know what the state of the art is, what directions of research seem most promising, and whether their investment has paid off.

These are the stakeholders during the relatively early precommercial stage in the life cycle of tools. Once there are commercial products available, then the roles shift. The users become consumers, becoming the people who purchase tools to assist them in their work. Once the tools have become mature enough to

make the transition from research to deployed software, the role of developers shifts as well, moving from research (publication-driven) to development (production of commercial software). The role of challenge evaluations also changes. The stakes become higher, and less information is shared because of intellectual property concerns. This limits the free flow of information, and may even preclude commercial participation. However, some evaluations (e.g., TREC) allow commercial participants with a reduced requirement for disclosure of technical details.

9.2.2 Dimensions of a Successful Evaluation

Next, we look at the dimensions of a successful evaluation. The critical ingredients for a successful evaluation are: choice of a meaningful application; clear, simple, reproducible evaluation measures; and availability of training and test data. Stakeholders have their own requirements for a challenge evaluation, and, in some cases, the needs of these stakeholders may conflict.

9.2.2.1 Choice of Application

In defining an evaluation, there is a tension between focusing on general techniques versus solutions to specific problems. This may, in part, reflect the different backgrounds of the developer versus the consumer. For example, in text mining, the technology developer may be a computer scientist with limited background in biology, and the consumer typically will be a biologist or bioinformatician. The consumer/biologist wants a particular biological problem solved, but it may be more natural for the technology developer/computer scientist to abstract away from the biology. This creates a tension in selecting applications. If the application is too specific, then it becomes difficult to assess the wider applicability of the technology. If the application requires too much biology, then it may deter the computer scientists. If the application is too technology-focused, then the biologists will not be interested. As we will see below, the solution has been to create a range of evaluations: some defined to address significant biological problems, others defined to tackle generic technology issues and to attract researchers from other disciplines.

9.2.2.2 Evaluation Metrics

It is critical to select clear, reproducible, and easily understood evaluation metrics. When complex cognitive tasks are involved, it can be quite difficult to provide an easy-to-measure definition of success. Evaluation in the language understanding area has generally used automated calculations that compare a system response to a "gold standard," which is the result of consensus-expert human performance on a task [22]. For example, in speech understanding, the measure has been transcription word error rate, using a human expert

transcription as a gold standard. Ideally, such measurements should be accompanied by a measure of inter-annotator agreement; that is, how well multiple human experts agree when performing this task.

An alternative approach is the "Olympic judging" approach, where experts rate system performance according to some agreed-upon rules. This is done in TREC, where human judges determine whether a particular passage or document is relevant to a request. The disadvantage of the Olympic judging approach is that the measures are not automatically reproducible. Each evaluation requires the participation of a panel of judges. Automated comparison of system results against a gold standard has the advantage that it is repeatable and can be used as an integral part of the development cycle (e.g., for hill climbing or machine learning).

However, there may not always be a "ground truth" or gold standard, particularly in biology, in which all knowledge is partial, new knowledge is constantly being added, and whole subfields may be restructured. Nonetheless, we can still compare computational predictions to predictions from other sources, using both automated comparison and human judgment. For example, CASP uses protein structure as determined by X-ray crystallography as its gold standard for protein structure prediction. This protein structure represents only partial knowledge, since the crystalline state may be quite different from the biologically active state. Computational predictions are compared algorithmically to this gold standard, but final determinations are made by a panel of experts.

Lastly, repeated evaluation on the same tasks is important for the field to demonstrate progress over time. The speech community held annual evaluations, and was able to reduce the error rate by a factor of two every two years [23]. However, as the error rate on one task dropped below some critical threshold (often 10% word error rate), the community also defined a new, more difficult task.

9.2.2.3 Data and Resources

A successful evaluation contributes resources and infrastructure that enable a research community to tackle problems of importance. Biology provides a particularly rich environment for text mining, because many resources already exist (e.g., synonym lists, nomenclatures, and ontologies). Many tasks, such as curation of biological databases, are currently performed by domain experts, providing large sets of "naturally occurring" data, which provide examples (input/output pairs) to system developers. Use of such datasets for evaluations has advantages and risks. One advantage is that these are tasks that experts are already performing, ensuring that the task has real biological importance and is tractable, at least for humans. If a task is too hard for an expert biologist to do, or if there is significant disagreement among experts, then such a task is not a

good candidate for use in a challenge evaluation. A second advantage is the availability of large (cheap) training datasets. This is highly desirable for any statistical or machine learning approach. Typically, for text mining applications, corpora of millions of words are desirable. Work by Craven and Kumlien [24] demonstrated the use of existing biological resources as sources of training data.

However, there are also risks with respect to data quality. These naturally occurring datasets exist to meet the needs of biologists, not the needs of text mining researchers, making them less accessible to nonbiologists. The naturally occurring annotations are often coarse-grained. That is, an annotation will exist at a document level (e.g., a gene annotated with its function and a reference to a publication), while the task may require extraction at the sentence or phrase level. The annotation also may be incomplete, in that only some of the positive examples are annotated. Novel learning algorithms may be needed to handle these kinds of partial and imperfect annotations, in contrast to most current approaches that rely heavily on perfect and complete annotation. Finally, preparation of gold standard test data requires extensive collaboration with expert biologists, and careful additional checks on data quality, as well as inter-annotator agreement experiments, to assess how well experts agree (see Chapter 8).

9.2.3 What Can Evaluation Accomplish?

Evaluation can facilitate progress in a new field, including:

- Encouraging scientific progress by supporting a systematic comparison of different techniques applied to a common problem, allowing researchers to learn from each other's successes and failures;
- Demonstrating progress over time, so that the organizations funding the research can see that their funding has produced quantifiable progress;
- Building a research community; in the case of text mining for biology, a key issue has been to attract researchers from both natural language processing and bioinformatics and to encourage cross-training and the formation of teams;
- Addressing problems of importance to the biology and bioinformatics community, which is key to keeping the biologists actively involved;
- Creating a legacy infrastructure to support future researchers by creating repositories of open source component software modules, as well as providing training and test data suites that can be reused for development and benchmarking of future applications;

- Lowering the barrier to entry, ideally, so that high school students could build and integrate components for text mining in biology to do their biology experiments;
- Creating standards and, eventually, creating a market for successful tools. If an evaluation is successful, many systems will use the training resources as input, and will produce output in a standard format for evaluation. These formats may become de facto data exchange formats. This, in turn, encourages sharing, comparison, and integration of components.

9.3 What to Evaluate?

The preceding section outlined the ingredients of a successful evaluation. As discussed above, the task should be of importance to the biology community. Ideally, it should be a task that expert biologists perform, because this ensures the existence of a substantial pool of training data, resources, and experts to create gold standard data. The task also should use existing standards where possible, because this increases reusability of materials. The task also must be relevant to the technology under development, and should provide a succession of progressively harder subtasks that will push research in useful directions. These considerations constrain the choice of evaluation in three dimensions: by the needs of biologists; by the need to evaluate existing technologies; and by the requirement that there be sufficient (and sufficiently cheap) training and test material.

9.3.1 Biological Applications

Two classes of biology tasks have been used as the basis for recent text mining evaluations. The first task is a search of the literature that individual researchers carry out to find articles relevant to their current research. This class of task focuses on document retrieval, and draws heavily on research from the information retrieval community. The second task is biological database curation, which includes the creating and updating of a biological database through a process of expert review of the literature, followed by the encoding of key findings in a biological database. Curation often requires document retrieval and classification as initial steps, but it also requires information extraction. This requires the ability to extract information from free text, and map it to a standard representation in a biological database, such as gene or protein identifiers associated with concepts in an ontology, such as the GO [20]. The remainder of this section discusses these two types of tasks and the associated technologies.

9.3.1.1 Literature Search

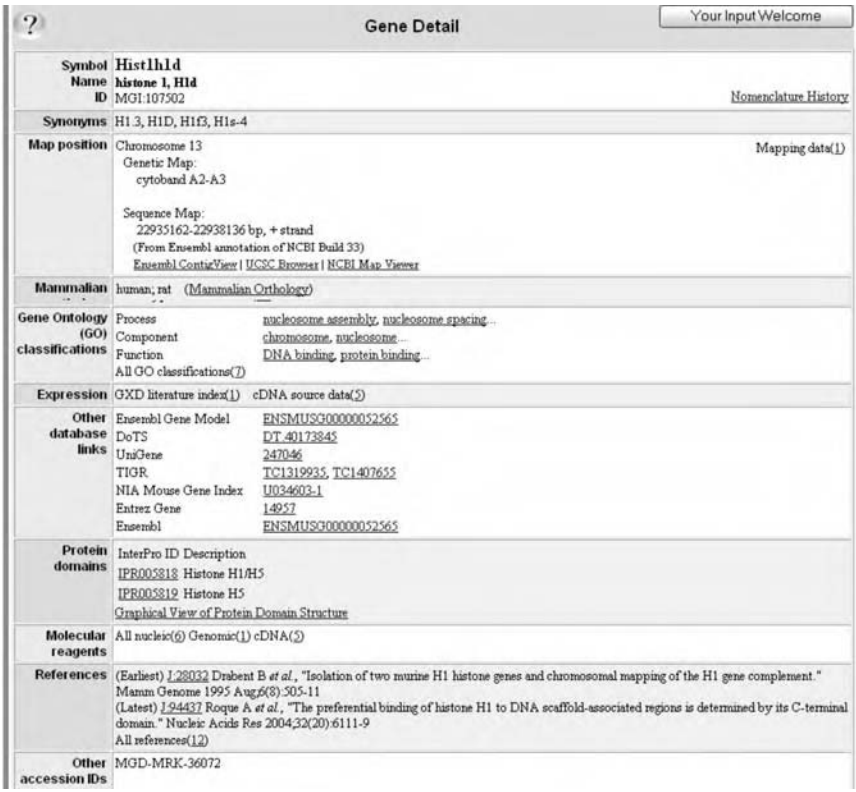
The individual biologist is now overwhelmed by the task of finding needed information from the increasing quantity of biological literature. The first step in a biological research project involves a background literature search, to identify who has done what on a particular problem. This may be followed by the creation of a personal database or annotated bibliography, to keep track of the information. Typically, initial references may point to other references, which also need to be included in the personal database or bibliography. Biologists would benefit from improved indexing into the biological literature, particularly indexing that automatically included the great number of synonyms for genes or proteins, or that allowed specification of search based on types of interactions or pathways.

Beyond document retrieval, users would like to get answers to their questions. TREC has had a track devoted to Question Answering for the past several years, encouraging the development of technology to return answers to natural language questions run against large document collections. The TREC Genomics Roadmap (http://ir.ohsu.edu/genomics/roadmap.html) includes question answering in its long-term goals. This would require some extension to the current question-answering technology to support answers to typical biological questions, such as, "Identify all known transcriptional targets for Smad4 expression and activation." Such questions require the return of lists of entities and associated references, rather than producing a single "factoid" answer.

Finally, literature search also can involve document classification, where the goal is to identify clusters of related documents. For some applications, clusters are predefined. For example, papers could be clustered by the experimental organism, so that papers about mice are in one cluster, while papers about rats are in another cluster. For other applications, such as searching on a topic mentioned in thousands of papers, the system could cluster the hits into thematic groupings to facilitate browsing.

9.3.1.2 Biological Database Curation

The second type of task focuses on a different community of biologists, namely curators of biological databases. A curated biological database captures critical information derived from the open literature, classified into biologically meaningful categories by expert biologists (the curators). There is a growing group of biologists involved in the creation of large systematic repositories of knowledge about a particular organism, as in the model organism databases (e.g., databases for flies [25], mice [26], yeast [27], collections of proteins [28], or binding relations [29]). The number of expert-curated biological databases is expanding. The 2005 edition of *Nucleic Acids Research* contains articles or pointers describing more than 800 biological databases and associated resources [30].



Gene Detail

Your Input Welcome

Symbol	Hist1h1d
Name	histone 1, H1d
ID	MGI:107502 Nomenclature History
Synonyms	H1.3, H1D, H1f3, H1s-4
Map position	Chromosome 13 Mapping data(1) Genetic Map: cytoband A2-A3 Sequence Map: 22935162-22938136 bp, + strand (From Ensembl annotation of NCBI Build 33) Ensembl ContigView | UCSC Browser | NCBI Map Viewer
Mammalian	human; rat (Mammalian Orthology)
Gene Ontology (GO) classifications	Process: nucleosome assembly, nucleosome spacing... Component: chromosome, nucleosome... Function: DNA binding, protein binding... All GO classifications(7)
Expression	GXD literature index(1) cDNA source data(5)
Other database links	Ensembl Gene Model: ENSMUSG00000052565 DoTS: DT.40173845 UniGene: 247046 TIGR: TC1319935, TC1407655 NIA Mouse Gene Index: U034603-1 Entrez Gene: 14957 Ensembl: ENSMUSG00000052565
Protein domains	InterPro ID Description IPR005818 Histone H1/H5 IPR005819 Histone H5 Graphical View of Protein Domain Structure
Molecular reagents	All nucleic(6) Genomic(1) cDNA(5)
References	(Earliest) J:28032 Drabent B *et al.*, "Isolation of two murine H1 histone genes and chromosomal mapping of the H1 gene complement." Mamm Genome 1995 Aug;6(8):505-11 (Latest) J:94437 Roque A *et al.*, "The preferential binding of histone H1 to DNA scaffold-associated regions is determined by its C-terminal domain." Nucleic Acids Res 2004;32(20):6111-9 All references(12)
Other accession IDs	MGD-MRK-36072

Figure 9.1 Screen shot from MGI showing gene names, functions, and references.

Text mining tools can play a potentially important role in aiding curators with the identification and capture of information for the maintenance of biological databases. Currently, curators cannot keep up with the flood of new information appearing in the biological literature. Text mining tools could help to improve the currency, consistency, and completeness of biological databases. To support their manual curation activity, biologists have created biological nomenclatures and ontologies to organize the information in these databases. These turn out to be important resources in understanding and processing biological information (see Chapter 3).

The choice of a curation task ensures that the task will have biological relevance. It also makes it possible to involve the expert curators in defining and evaluating the task, since they have in-depth knowledge about the annotation process, and they have an interest in the development of tools to assist them in this process. Furthermore, curation tasks come with annotated data. Biological databases capture these expert annotations with links to the articles from which

the annotations are derived. Figure 9.1 is an excerpt from the Mouse Genome Informatics (MGI) database. It shows the association of a gene name, gene identifier, and gene synonyms (top), with GO functional information (middle), and literature references (bottom). This illustrates the kind of rich information available in expert curated biological databases.

Figure 9.2 shows a schematic version of a pipeline for model organism curation. Curation begins with the identification of relevant articles from the literature (e.g., the MGI database, and all the papers discussing a particular gene in mouse). Part of this task is to ensure that the papers contain experimental findings for the gene or gene product in the specific organism of interest. This document retrieval task was the basis for the KDD 2002 Challenge Cup Task 1 [18], and the categorization task in TREC Genomics Track 2004 [6].

A later step in the curation pipeline involves listing the genes or gene products that have sufficient experimental information in the article to warrant curation. This formed the basis for the BioCreAtIvE normalized gene list task.

A third step involves the actual curation—assignment of properties to the genes and gene products, based on the experimental findings reported in the literature. The adoption of a shared ontology across organisms, namely GO, has enabled the comparison of gene and protein function across organisms (comparative genomics). GO provides three separate hierarchies that allow annotators to describe molecular function, biological process, and cellular localization of genes and gene products. This step is the basis for BioCreAtIvE task 2, assigning functional annotation for specific genes based on evidence provided in the literature.

These curation tasks require different technologies, including document retrieval, information extraction and normalization, and fact or relationship extraction. They also cover different gradations of difficulty: from the relatively easy tasks, such as generation of lists of unique gene identifiers; to intermediate tasks, such as selection of relevant documents for curation; to the very difficult tasks, such as GO annotation of gene products based on information in the open literature.

9.4 Current Assessments for Text Mining in Biology

There have been four challenge evaluations to date for text mining in biology, listed in chronological order:

1. KDD Challenge Cup Task 1 (2002);
2. TREC Genomics Track (2003, 2004);
3. BioCreAtIvE (2003–2004);
4. BioNLP (2004).

We will describe each of these briefly in terms of the application, the technologies and the overall results.

9.4.1 KDD Challenge Cup

The first biology-based challenge problem was offered in the context of the KDD Challenge Cup, as task 1 in 2002 [18]. This task was created in collaboration with Gelbart and the FlyBase curation team. The task was selected to balance biological importance with tractability for the existing state of technology and availability of resources (e.g., full text articles). It was a task at the beginning of the Harvard Flybase curation pipeline (see Figure 9.2), specifically, identification of the papers to be curated for Drosophila gene expression. The criterion was that the paper must contain experimental evidence for gene products (mRNA transcripts, polypeptides, or proteins) associated with a given gene.

The KDD Challenge Cup Task was defined as follows:

- Collect a set of (full text) articles on genetics or molecular biology, and for each paper, list the genes mentioned in that paper;
- Determine whether the paper meets the FlyBase gene-expression curation criteria, and for each gene, indicate whether the paper has experimental evidence for its gene products (mRNA and/or protein).

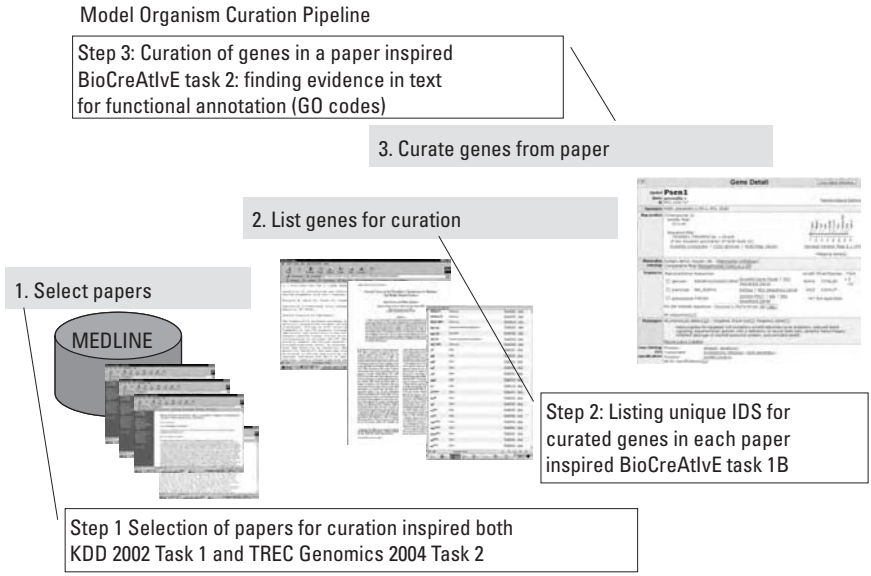


Figure 9.2 Curation pipeline for GO model organism databases.

The training set consisted of full text articles, cleaned to remove irregular typography that occurred in the full text papers. In addition to the articles, the participants received the FlyBase standardized nomenclature for the genes, including synonym lists for each gene. Participants also received the controlled vocabulary terms from FlyBase that indicated experimental methods, (e.g., 'Northern blot,' 'immunolocalization'). For the test, systems were presented with a new set of articles, together with unique identifiers for the genes mentioned in each article. Table 9.1 summarizes the size of the training and test sets.

Systems were required to return three things:

1. A ranked list of articles for curation, where papers containing experimental evidence of interest ranked higher than papers that did not contain such evidence;
2. A yes/no decision on whether to curate each article;
3. For each gene listed for an article, a yes/no decision about whether the article contained experimental evidence for that gene's products (e.g., RNA, protein/polypeptide).

Each subtask had a simple scoring procedure. For the ranking task, the metric was the area under the receiver operating characteristic curve (AROC). The ROC curve measures the trade-off between sensitivity (recall) and the probability of a false alarm. For the yes/no curation decisions on the papers, the metric was balanced F-measure[1] with precision and recall weighted equally. F-measure also was used at the gene level for the yes/no decisions on experimental evidence for each gene's products. The average of these three scores (equally

Table 9.1
KDD Datasets and Participants

Task	**Participants**	**Runs**	**Training Set**		**Test Set**
			Total	Positive Examples	Total
KDD Challenge Cup	18	32	862 articles	283 articles	213 articles

1. The balanced F-measure is (2 × precision × recall)/(precision + recall), where recall is number of correct "yes" decisions returned by the system, divided by the number of possibly correct "yes" decisions; precision is the number of correct "yes" decisions returned by the system, divided by the total number of "yes" decisions returned.

weighted) produced the overall system score. Each team was allowed to submit up to three runs. The results are shown in Table 9.2. The top performing team obtained both the highest overall score and the highest scores on the individual tasks.

Overall, the evaluation was a success. There was strong international participation, with 18 teams from 8 countries, and a steady stream of requests for the dataset following the evaluation. Many teams consisted of a biology partner and a computer science/data mining partner.

The preparation of the training and test data required significant effort from both the Harvard FlyBase team and the MITRE team, even though the task was designed to make use of existing data annotated in FlyBase. For example, the organizers provided explanations understandable to nonbiologists for decisions about curating (or not curating) genes in an article based on experimental evidence.

While the task did not involve returning the experimental evidence, it did require association of evidence with specific genes or gene products for the third subtask—deciding, for each gene, whether the article contained experimental evidence for that gene's products (RNA, protein/polypeptide). As a result, many groups reported that "bag-of-words" techniques alone were not very successful. The top system used information extraction and manually constructed rules to identify "interesting" patterns. Other high scoring systems generally combined identification of gene names with patterns associated with experimental evidence.

The results reflect the overall task difficulty. While the highest scoring results for returning a ranked list seemed promising, the real test is whether such a ranked list could significantly improve curator productivity, which would be an important dimension to add in future tests. At the fine-grained level of associating experimental results with specific gene products (see the last row in Table 9.2), the F-measure of 67% indicates that the techniques were not very accurate at the gene level.

Overall, the 2002 KDD Challenge Cup task made two major contributions:

Table 9.2
Results from KDD Challenge Cup 2002

Evaluation Task	Best	Median
1. Ranked-list	84%	69%
2. Yes/No curate paper	78%	58%
3. Yes/No gene products	67%	35%

1. It showed that there was significant interest across several communities in working on text mining in biology.
2. It showed that this kind of document selection task made an interesting and appropriate challenge task. Some of the lessons learned from KDD influenced the development of the Mouse Triage task for TREC Genomics Track 2004.

9.4.2 TREC Genomics Track

The TREC Genomics Track [5, 6] began with a trial run in 2003, and now has five-year funding from the National Science Foundation through 2008. The track, like many TREC tracks, has a core task, which is "ad hoc" search—thes earch for documents relevant to an arbitrary (biological) topic, from a data base of documents. It also has experimental tasks that may change each year.

The MEDLINE abstract collection is the starting point for biology literature searches, and constitutes the document collection for the track's ad hoc search. The evaluation issues for ad hoc search are well understood, since TREC has been running such evaluations for over a decade. The main requirements are a readily accessible document collection; a set of representative topics; and a method to determine, for a given topic/document pair, whether the document is relevant to the topic. The main TREC evaluations have used human assessors to make these determinations. The evaluation metrics are based on precision (i.e., number of relevant documents returned, divided by number of documents returned), and recall (i.e., number of relevant documents returned, divided by number of relevant documents in the collection). Computation of precision is straightforward, once a set of documents returned for a topic has been judged for relevance. However, computing recall is not straightforward for large collections, because the calculation requires the true number of relevant documents in the denominator, which only can be known by exhaustive inspection of a large collection for each topic. Therefore, relevance is computed against a much smaller collection, created by pooling the top *N*-ranked documents returned from a sufficient number of participants [31]. For example, if there are 25 participants, and each system provides its 100 top-ranked documents, this creates a pool of 2,500 documents for inspection. Pooling candidate document sets from multiple retrieval engines means that the most relevant documents will be in this set. It is then possible to estimate how many more documents would have been found by increasing the size of the pool, to give some estimate of the recall error from pooling.

Defining a set of representative topics for the biology domain is an area of ongoing experimentation. For TREC 2003, the track used Gene Reference into

Function (GeneRIFs)[2] from the NCBI LocusLink database (http://www.ncbi.nlm.nih.gov/LocusLink/) to generate a set of topics. The topics had the form: "For some gene X, find all MEDLINE references that focus on the basic biology of the gene or its gene products." A few sample GeneRIFs for the gene PSEN1 (Presenilin 1) are shown in Table 9.3.

The 2003 Genomics track used GeneRIFs as substitutes for relevance judgments, because of funding constraints. This turned out to be problematic, because there are many articles that discuss gene function but do not end up as GeneRIFs. An analysis of an OHSU training run [5] showed that more than 40% of the documents retrieved were relevant but were not GeneRIFs. Systems returning relevant documents that were not GeneRIFs would have suffered a penalty in their precision score. The 2003 evaluation used two metrics for the ad hoc task: mean average precision (i.e., the average of precision after each relevant document is retrieved for a given query); and number of relevant documents in the first *N* documents, averaged over all queries (*N* was 10 or 20). Table 9.4 provides information for the TREC Genomics Track for both 2003 and 2004. For the top scoring system on the 2003 ad hoc task, the highest mean average precision was slightly more than 40%, and slightly more than three of the first 10 retrieved documents were relevant, averaged over all queries. The top performing systems made use of medical terminology for species information and for synonyms.

The secondary task for 2003 was a sentence extraction task. The goal was to find the sentence(s) from the MEDLINE abstract or full text article that gave rise to the GeneRIF. Performance was measured by textual overlap between the GeneRIF and the passage, using several different ways of calculating overlap.[3] Preliminary analysis indicated that 95% of the GeneRIFs contained text from

Table 9.3
List of Some GeneRIFs for the Gene Presenilin-1 (PSEN1) in LocusLink

PubMed ID	Statement of Function
12058025	Interaction with GFAP epsilon
11876645	Inhibition of endoproteolysis by gamma-secretase inhibitors

2. GeneRIFs are short summaries of gene function, with a pointer to the source of the information, with the article and its abstract in MEDLINE.
3. The measure shown in Table 9.4 is the DICE coefficient for two documents D1 and D2, calculated as 2 times the number of words in both D1 and D2, divided by the total number of words in D1 plus the number of words in D2.

Table 9.4
TREC Genomics Track Results 2003–2004

TREC Genomics	TASK	Corpus	Training	Test	Participants	Runs	High Scores	
2003					29			
	Ad hoc	MEDLINE 525,000 Abstracts	50 topics	50 topics	25	49	MAP 0.417	3.2 relevant @ 10 docs
	Extract GeneRIF statement from document	Abstracts and articles for 139 GeneRIFs	None	139 GeneRIFs	14	24	DICE 57.8; Baseline: (Titles) 50.5	
2004					33	145		
	Ad hoc	MEDLINE 4.5 million Abstracts	50 topics		27	47	MAP 0.41	6.0 relevant @ 10 docs
	Triage	12,000 full text articles	6,000 articles; 375 positive examples	6,000 articles; 420 positive examples	18	59	Utility 0.65; Baseline return-all 0.33	F-score: 0.28; Baseline "Mice" 0.26
	Annotation of GO domain		504 documents; 1,418 documents/ gene pairs	378 documents; 877 documents/ gene pairs	10	36	F-score: 0.56; precision: 0.44; recall: 0.77	

Note: Doc = document.

the title or the abstract. A baseline system that simply used the title as the GeneRIF scored in the upper one-third of systems submitted.

Overall, the 2003 Genomics Track demonstrated that there was a large community of interested researchers—29 participating groups. While the results were not very meaningful due to the preliminary nature of the task definitions and evaluation procedures, the track provided experience in defining appropriate tasks, and laid the groundwork for the future Genomics tracks. It also exposed nonbiology groups to some of the available biological resources, and brought together bioinformatics researchers and computer scientists.

For TREC 2004, Hersh's group at OHSU, and volunteers from participating sites, conducted a study of user needs and candidate queries. These formed the basis of the ad hoc queries, and Figure 9.3 shows a sample query [6]. Funding made it possible to hire assessors to assess results using the traditional TREC pooled data approach. Two people with backgrounds in biology performed the assessment. Kappa statistics[4] showed "fair" agreement (0.51) between the assessors, which is consistent with other experiments on inter-assessor agreement in TREC.

Overall, 27 groups submitted 47 runs for the ad hoc task. The document collection consisted of a 10-year subset of MEDLINE, including approximately 4.5 million records. For the high-scoring run for TREC 2004, precision at 10 documents was slightly more than 6 documents, a definite improvement over the much lower results in TREC 2003. It is difficult to interpret the significance of the ad hoc retrieval results for working biologists, but for future evaluations, an interesting benchmark might be to compare these results to those achieved by a biologist using the standard MEDLINE tools to compose a query.

The second task for TREC Genomics 2004 was a "triage" task, using data from the Mouse Genome Informatics database [26]. This task was similar in design to the KDD Challenge Cup task for FlyBase [18]. Given a set of candidate documents, the task was to classify those documents containing experimental data on mouse genes that would warrant further curation for GO codes. For this task, a collection of full text articles was provided by Highwire Press, consisting of two years' worth of *Journal of Biological Chemistry*, *Journal of Cell Biology*, and the *Proceedings of the National Academy of Science*. This resulted in collections of roughly 10,000 articles for 2002, and the same for 2003. Querying for the terms "mouse," "mus," or "murine" produced subsets of roughly 6,000 documents for each year. The 2002 papers were used for training, and the 2003 papers were used for testing.

4. The kappa statistic measure tests the null hypothesis that there is no more agreement than might occur by chance, given random guessing; kappa ranges from 0 for chance agreement, to 1 for full agreement.

```
<TOPIC>
   <ID>51</ID>
   <TITLE>pBR322 used as a gene vector</TITLE>
   <NEED>Find information about base sequences and
restriction maps in plasmids that are used as gene
vectors.</NEED>
    <CONTEXT>The researcher would like to manipulate the
plasmid by removing a particular gene and needs the original
base sequence or restriction map information of the
plasmid</CONTEXT>
</TOPIC>
```

Figure 9.3 Sample ad hoc query from TREC genomics 2004.

To evaluate performance on the triage task, the track developed a utility measure, in addition to the usual measures of precision, recall, and F-measure. The utility measure was designed to weigh the utility of retrieving a relevant document versus the utility of retrieving a nonrelevant document. Perfect prediction had a utility of 1, completely imperfect prediction would have a value less than 0, and triaging everything was estimated to have an approximate value from 0.25 to 0.35.

The top performing system received a utility rating of 0.65, demonstrating that it classified documents better than the "triage everything" baseline of 0.33. However, the top F-score was just slightly more than 0.28. The systems had high recall. The top scoring systems had recall ranging from 0.88 to 0.96, but this came at the expense of poor precision, which ranged from 0.16 to 0.09. Perhaps the most unsettling result was a run performed by one group that classified documents solely based on the presence of the MeSH[5] term "mice". This system scored almost as well as the top performing system, with an F-score of 0.26, and a utility score of 0.64. There are several hypotheses that might account for these results:

1. The task may be so complex that it is difficult to select appropriate features, and systems were able to provide little added value, beyond telling whether an article was about mice.
2. The curation decisions may not be reproducible.

5. MeSH is the National Library of Medicine's controlled vocabulary used to index articles for MEDLINE. Each article in MEDLINE is indexed by a number of MeSH terms.

3. The metrics may be problematic, particularly in the weighting of true positives versus false positives.

These results are now being analyzed. It will be necessary to understand these results before designing another triage experiment for literature curation.

In addition to the triage task, there were two further annotation tasks. The first was to identify the top-level GO domains (e.g., biological process, molecular function, or subcellular location) that could be curated from the information in the article. The second task was to identify the correct GO code, but only two groups participated in this task, and it is not discussed here.

For the GO domain annotation task, systems were presented with documents and a set of genes associated with each document. The task was to determine which (if any) GO domains could be curated from the article for the gene. The top F-measure for this task was 0.56, with precision of 0.44 and recall of 0.77. This task, like the KDD gene-experimental evidence task, required association of information with a specific gene. The results are somewhat lower than the KDD task for genes, reinforcing the observation that it is difficult to associate specific information with individual genes. See also the discussion of BioCreAtIvE task 2 (Section 9.4.3.2), for a discussion of experiments assigning GO codes to genes.

The plan for the TREC Genomics Track is to refine the tasks and the metrics, while also exploring more complex tasks in the later years of the project. The user community, including the biological curation community, has been actively participating in definitions of these tasks. Through this process of continued feedback and refinement, the Genomics Track should be able to make substantial contributions to the field of document retrieval and access for biology.

9.4.3 BioCreAtIvE

The explicit goal of the Critical Assessment of Information Extraction in Biology (BioCreAtIvE) [19] was to assess the state of the art for information extraction in biology, based on problems of importance to the biology community. A team of computer scientists and biologists/bioinformaticians from CNB/CSIC (Madrid) and from MITRE, who were committed to evaluation of important tasks for the biology community, developed the evaluation.

BioCreAtIvE consisted of two tasks: gene name extraction and normalization (task 1), run by the group at MITRE [32, 33]; and extraction of information for assignment of GO codes to proteins (task 2), run by the group at CNB/CSIC [34]. The structure of BioCreAtIvE is summarized in Table 9.5.

Table 9.5
BioCreAtIvE Evaluation Tasks

BioCreAtIvE	Task	Data Type	Data Source	Training	Test	# Participants	# Runs
Task 1a	Identify all mentions of gene-or-protein in sentence	Sentences from abstracts	MEDLINE	10,000 sentences; 200,000 words	5,000 sentences	15	40 official + 4 unofficial
Task 1b	Return list of unique gene identifiers mentioned in text	Full text	MEDLINE; gene lists from Fly, Yeast, Mouse DBs	5,000 abstracts per organism	250 abstracts per organism	8	Yeast 15; Mouse 16; Fly 11
Task 2.1	Find evidence passage supporting protein/GO code	Full text	JBC; also for training Nature Medicine, Nature Genetics, Oncogene	803	113 articles; 1,076 PMID/gene-product/GO triples	9	21
Task 2.2	Return GO codes and supporting evidence passages	Full text	As for Task 2.1	803	99 articles	7	18

9.4.3.1 BioCreAtIvE Task 1: Extracting Genes

Task 1 focused on identification and normalization of gene names. There were two subtasks, with task 1a focused on identification of gene mentions in running text, and task 1b focused on creation of lists of unique gene identifiers. Different groups with somewhat different aims constructed these tasks. The data for task 1a was provided by Tanabe et al. (NCBI) [35], and was derived by annotation of single sentences selected from MEDLINE abstracts. The biological rationale for this task was that identification of gene-or-protein callouts in running text serves as a critical first step to tasks, such as indexing of articles by gene name, or extraction of gene and protein interactions. Task 1a was very close to the named entity tagging task used extensively in the natural language processing community (see Chapter 6). One objective in inclusion of this task was to facilitate participation for groups whose main expertise was in natural language processing, since the task required relatively little expert knowledge of biology. This was the most heavily subscribed BioCreAtIvE subtask, with 15 teams participating. An example sentence is shown here.

> "Furthermore, as in the human gene, the 3′ end of the Cacna1f gene maps within 5 kb of the 5′ end of the mouse synaptophysin gene in a region orthologous to Xp11/23."

In this example, the system must identify the gene/protein names "Cacna1f gene" (or "Cacna1f") and "mouse synaptophysin gene" (or minimally, "synaptophysin"), which are underlined. However, a phrase like "the human gene" is not marked because it is not the name of a particular gene. The answer key provides for alternative forms (e.g., "Cacna1f gene" or "Cacna1f").

Table 9.6
Gene Mention Scores for BioCreAtIvE Task 1 and BioNLP

			High F-Score		
Evaluation	**Task**	**# Classes**	**F-score**	**R**	**P**
BioCreAtIvE Task 1a	Gene-or-protein mentions	1 class	0.83	0.84	0.83
BioCreAtIvE Task 1b	Normalized gene list	1 class			
		Fly	0.82	0.80	0.83
		Mouse	0.79	0.82	0.77
		Yeast	0.92	0.89	0.95
BioNLP	Mentions of protein, DNA, RNA, cell line, cell type	5 classes	0.73	0.76	0.69

Participants were given 10,000 annotated training sentences, and were tested on an additional 5,000 blind test sentences. The main findings from task 1a [32] were that 4 of the 15 teams, using techniques such as Hidden Markov Models and Support Vector Machines, were able to achieve F-measures higher than 0.80, with a high of 0.83 balanced precision and recall (see Table 9.6). These results are somewhat lower than for similar tasks from the newswire domain. Yeh et al. [32] provide an analysis of these differences: extraction of organization names has been reported at F-measures higher than 0.90. Approximately one-half of the difference in F-measure can be attributed to longer names found for genes, compared to organization names. The remaining discrepancy may be attributable to annotation inconsistencies due to fuzziness in the definition of what constitutes a gene-or-protein name.

Task 1b created lists of gene identifiers for genes appearing in a given article or abstract. This is a task that is currently performed manually by curators for various model organism databases. This meant that there were datasets potentially available for both training and testing. Three model organism databases were selected to serve as sources of gene lists associated with papers: fly [25], mouse [26], and yeast [27]. By choosing several model organisms, the idea was to encourage approaches that could be readily adapted to the different vocabularies associated with the different organisms.

The curation process for these model organism databases was done on full text articles. However, it is difficult to obtain large quantities of full text articles, and the decision was made to use abstracts from MEDLINE instead. It was necessary to edit the gene lists to make them correspond to genes mentioned in the abstract, rather than the genes curated from the full text article. For the training data, a simple matching program removed genes that were not mentioned in the abstract, generating a large quantity of noisy annotated data. For the final test data, the lists were carefully hand-corrected to fix discrepancies [36]. This turned out to be a labor-intensive and somewhat error-prone process.

Eight groups participated in task 1b. The organizers provided synonym lists for each organism, consisting of the unique gene identifier and its alternate names, as listed in the lexical resources derived from each model organism database. The results [33] are shown in Figure 9.4 and summarized in Table 9.6. Performance varies by organism. The top score for yeast was an F-score higher than 0.90; the top scores were F-scores around 0.80 for both fly and mouse. Detailed analysis showed that the differences among organisms could be attributed to a variety of factors, including: extensive ambiguity in names and overlap of gene names with English terms (fly); complex multiword gene names (mouse); and quality of the training data, especially for mouse, where recall on the training data was estimated at only 55%. Training data quality is shown as open symbols for fly, mouse, and yeast in Figure 9.4.

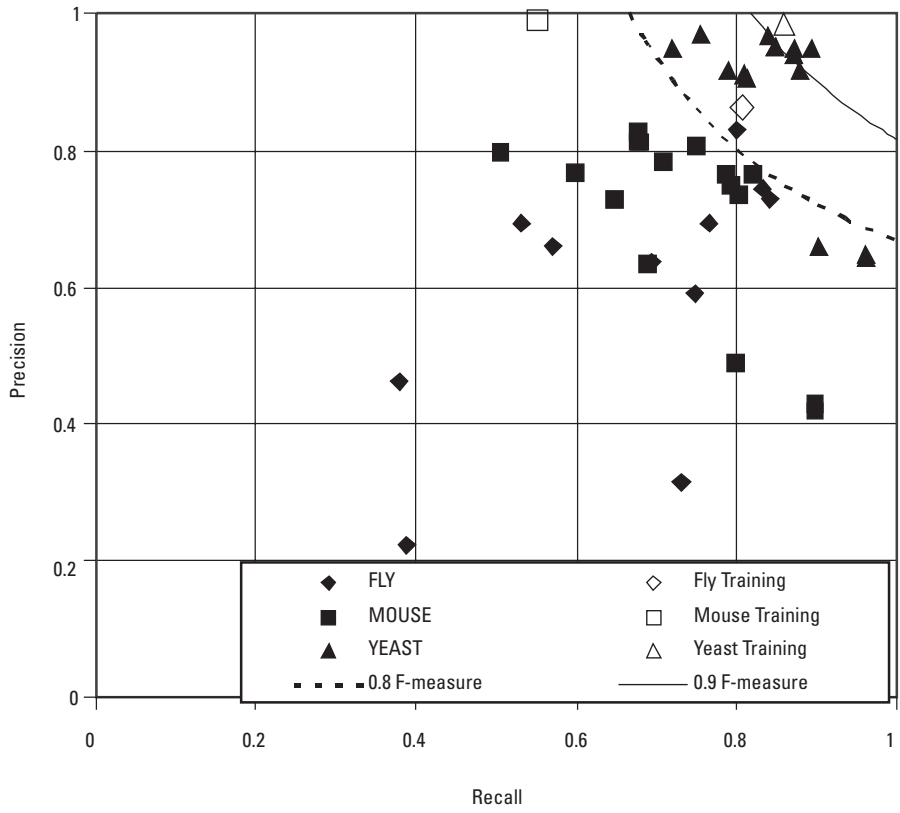


Figure 9.4 Performance results for BioCreAtIvE task 1b: Normalized gene lists.

Task 1b posed two major difficulties: recognizing the many synonyms and variants of gene names; and associating ambiguous names with the correct gene identifier (disambiguation). The general approach was to identify gene mentions, associate these mentions with the appropriate unique identifier, and then collapse these into a list of unique identifiers. Most systems used the lexical resources for recognizing the synonyms, in addition to some kind of flexible pattern matching and/or lexicon augmentation to take into account the many variations in case, punctuation, and spacing that occur in gene name nomenclature. Several high-performing systems also relied heavily on editing the lexical resources to remove ambiguous gene names and abbreviations (*NO*). Most groups hand-tailored the system for each organism, although one group used a generic "tag-everything-then-prune" approach that involved liberal pattern matching, based on the unedited lexical resources to generate candidate gene identifiers, followed by a classifier to remove incorrectly tagged candidates.

9.4.3.2 BioCreAtIvE Task 2

BioCreAtIvE task 2 focused on the automatic assignment of GO annotations to human proteins using full text articles [34] (see Table 9.5). The construction of task 2 was motivated by the needs of curators who wanted help in identifying passages from the full text article that provided evidence for the GO annotations of a specific protein. The task 2 organizers made a conscious decision to provide data "as is," to reflect the realities of a biological application. This meant that there were no special lexical resources supplied. The training set consisted of approximately 800 full text journal articles and their associated annotations (e.g., protein and GO code) at the article level, taken from GOA (http://www.ebi.ac.uk/GOA/). The training set was released to participants with no further annotations; that is, the participants had to find the evidence passages that supported the GO annotations. The test set consisted of approximately 200 articles that were curated by the GOA team specifically for the assessment. Participants also had to find their own lexical resources, such as synonyms for GO terms, as well as protein name synonyms.

The GOA curators at the EBI agreed to evaluate the results from the systems that took part in the BioCreAtIvE task 2 [37]. GOA contains manually extracted associations of proteins to GO terms, together with the identifier of the article that provided the evidence, as well as the type of evidence supporting those annotations.

For instance, the following example corresponds to a single GOA entry.

> P41220 RGS2 HUMAN GO:0005096 PMID:10747990 TAS F Regulator of G-protein signaling 2 IPI00013177.

Here, the human protein RGS2 with the accession number P41220 has been annotated 'Regulator of G-protein signaling 2 (GO ID 0005096)', using information derived from the article with the PubMed ID 10747990.

Task 2 had two subtasks. For task 2.1, the goal was to return a short text passage that provided evidence for the GO code assigned to that protein, given the paper, the protein, and the GO code. The ideal passage contained both a mention of the protein and the evidence for the GO code assignment. Figure 9.5 shows an example of an "easy" evidence passage for a protein and its GO code. The evidence passage illustrates one aspect of associating text mentions with GO codes. The description of the protein uses the term "inhibiting cell proliferation", whereas the GO concept is described as 'negative regulation of cell proliferation'. Many of the text mentions required much more sophisticated processing, including coreference resolution to handle information distributed across multiple sentences. Task 2.2 was similar, except that the system had to predict the GO codes for the protein, in addition to returning the supporting evidence passages. This further increased the difficulty of the task.

Protein	p21waf/cip1
GO: 0008285	negative regulation of cell proliferation
PMID	10692450
Evidence Passage:	The p21waf/cip1 protein is a universal inhibitor of cyclin kinases and plays an important role in inhibiting cell proliferation

Figure 9.5 Sample triple for BioCreAtIvE task 2.1 with evidence passage.

Three expert GOA annotators were responsible for the assessment of the submitted evidence passages and GO predictions. This turned out to be a very labor-intensive process, and would need to be greatly streamlined in any future evaluation. The GOA team also provided an important inter-annotator agreement study on consistency of the expert annotation [37]. This study showed that the annotators were generally consistent, but that they annotated to different depths in the GO hierarchy, depending on their specific areas of expertise. Overall, the pairwise inter-annotator precision was higher than 90%, but recall was approximately 70%. The study also revealed differences in annotation due to changes in the experimental setup. Curators did "annotation by paper" for task 2, as opposed to the normal curation process of "annotation by protein," where many papers are used to come up with the most accurate GO code assignment.

For task 2.1 (evidence passage retrieval), the best performing systems returned evidence passages judged correct for approximately 300 out of 1,000 cases. Task 2.2 was much harder, because it required prediction of GO terms, in addition to returning the evidence passage. In this case, performance dropped significantly. The answers of the top performing system were correct for approximately 80 of 640 answers returned, out of 1,200 possible article/protein/GO code tuples (see Table 9.7). There were a number of approaches to task 2. One approach created a set of (weighted) terms associated with each GO code, and then searched for regions containing these terms. Several systems that returned passages for each GO code used this approach, achieving recall and precision of approximately 30% for task 2.1. A second approach used machine learning approaches to identify relevant regions of the text. This approach was made difficult by lack of real training exemplars. A third approach used a more information extraction–based approach, which led to higher precision at the expense of low recall. For one system, 125 correct answers were returned out of 251 in task 2.1.

BioCreAtIvE task 2 represents an ambitious end goal for text mining, requiring the ability to map complex concepts expressed in free text to

Table 9.7
BioCreAtIvE Task 2 High Scores

BioCreAtIvE Task 2	Task	Test	High Scores
Task 2.1	Evidence passage	113 articles; 1,076 PMID/protein/GO triples	303 correct out of 1,050 submitted
Task 2.2	GO code + evidence passage	99 articles; 1,227 PMID/protein/GO triples	78 correct out of 634 submitted

ontological concepts from the GO. The semantic distance between a simple concept name in GO and its expression in text made this particularly challenging. However, this task is also difficult for humans, as the inter-annotator agreement experiments [37] showed. Absence of training data was a major bottleneck for this task. For subsequent experiments, the evidence judgments collected during this experiment will be made available, to provide training data. It would be feasible to collect much more training data if there were a tool that allowed curators to capture the association between evidence text and the extracted biological facts as they curated. Development of such a tool would speed curation, as well as provide significant quantities of annotated training data.

9.4.4 BioNLP

The JNLPBA held an evaluation focused on a shared task in bioentity recognition [21]. This work built on the GENIA-annotated corpus that is being prepared at Tokyo University [37]. Training data came from GENIA version 3.02, which contains 2,000 annotated MEDLINE abstracts (more than 18,000 sentences) that were selected based on keyword hits for the MeSH terms 'human', 'blood cells', and 'transcription factors'. GENIA has been annotated for 36 classes of biological named entities. However, for the JNLPBA shared task, only five classes were used: 'protein', 'DNA', 'RNA', 'cell line', and 'cell type'. To create the test data, some 400 new abstracts were annotated for these classes. One half of the abstracts were chosen from the same domain as the training data, using the same set of keywords, while the other half were chosen from a more general domain, using only the MeSH keywords 'blood cells' and 'transcription factors'. The test set was also subdivided according to year of publication. These sets contained somewhat different proportions of the five classes of named entities. For example, the oldest set contained fewer instances of 'protein', and more terms relating to 'cell line'.

There were a total of eight participating groups. The results were measured as F-scores. In addition to a score for the strict match, right boundary match (allowing extra or missing words in front of the entity mention), and left boundary match (extra or missing words at the end of the name) also were scored. These scores take into account some fuzziness in whether various words should be part of a name. Should the word *human* be included as part of the name in *human interleukin-2 gene expression*? Should the word *gene* be included as part of the name?

The best F-measure for the exact-match task was around 0.73 (see Table 9.6). This was considerably lower than the top F-measure of 0.83 reported for BioCreAtIvE task 1a. There are obvious differences in task difficulty. For example, BioCreAtIvE task 1a required named entity identification for only one class (gene-or-protein), compared to the five classes for the BioNLP evaluation. However, Dingare et al. [39] provide a detailed analysis of some of the differences in the evaluations, since their team participated in both evaluations. Their top F-score in BioCreAtIvE was 0.83 [40]. To make the BioNLP results comparable, they combined the 'DNA', 'RNA', and 'protein' categories, and eliminated 'cell line' and 'cell type'. They then evaluated their system, and achieved an F-measure of 0.74 on the "single-class" BioNLP data, which is still significantly lower than their BioCreAtIvE results. They attribute this to annotation quality in the two corpora. They estimated that, of 50 system errors examined, from 34 to 35 could be attributed to inconsistent annotation in the training or evaluation data for BioNLP, often related to left and right modifiers.

The evaluation results may be extremely sensitive to the quality of the test and training data. The results from [39] indicate that there may be as much as a 30% difference in error rate attributable to annotation inconsistencies. This is also interesting with respect to interpreting BioCreAtIvE task 1b, where the data was derived via approximate techniques from biological databases. The noise in the data also may have placed a threshold on the performance of BioCreAtIvE systems.

9.5 What Next?

We have presented a snapshot in time of a rapidly growing field. If we compare the landscape for the evaluation of text mining in biology to either bioinformatics or natural language processing, then it is clear that this is the beginning of a decade-long process. From the BioCreAtIvE and BioNLP evaluations, we can conclude that gene/protein name identification is a problem that is almost solved. There are systems able to obtain accuracy from 80% to 90%, depending on the specific task. The broader task of biological entity identification needs more work, but should be tractable, provided that there are lexical

resources comparable to those used for gene/protein name identification. The KDD and TREC experiments indicate that selection or ranking of papers by biologically relevant criteria is still an open research problem. It may be that classic information retrieval is not fine-grained enough to make the distinctions necessary to decide whether or not to curate a paper.

In addition to these specifics, we have learned a number of general lessons from the evaluations done to date for text mining in biology.

1. Biology presents a rich set of resources that provide opportunities for defining evaluation tasks, and for creating significant training corpora based on real biological applications.
2. There is a trade-off in quantity versus quality of training data, and, therefore, in the cost of the evaluation. On the one hand, the "found" datasets often are only partially annotated, or are noisily annotated. The BioCreAtIvE datasets for tasks 1b and 2 illustrate this well. Despite significant effort spent in cleaning the data (particularly for the development test and test sets for task 1b), the training datasets were noisy (task 1b) or minimal (task 2). On the other hand, specially created datasets also may be limited in quantity and quality, because it is expensive to create high quality, consistently annotated datasets for a natural language processing task. Consistency checks on the BioNLP data and on BioCreAtIvE task 1a data revealed a number of inconsistent annotations with respect to boundaries of gene or protein names. In addition, the underlying definitions of biological entities (e.g., gene, protein, RNA, and so forth) are often quite fuzzy, even to biologists.
3. System performance may be limited by quantity of training data (BioCreAtIvE task 2), but also by quality, as was the case for the mouse data in BioCreAtIvE task 1b.

These lessons can inform future evaluations, which should continue to probe biologically motivated applications. This is still largely unexplored. There are obvious application classes that could form the basis for future evaluations, including:

- Question answering on MEDLINE;
- Summarization (i.e., some users of biological databases would prefer natural language summaries rather than tables);
- Interactive curation tools that support the curator, and eventually learn from the curator;
- Functional annotation tools, combining linguistic and bioinformatics data sources;

- High-throughput data interpretation tools that extract information about genes and pathways from the literature and biological databases;
- A cross reference tool that checks articles for submission (e.g., cross checking gene names, Genbank identifier, nomenclature, and model organism gene identifiers).

There are two important additional dimensions that need to be addressed in future evaluations. The first is for user interfaces that support end-to-end processing. The experience in the natural language community has been that stand-alone named entity identification tools are not particularly useful. If the developers can make it easy to embed these tools in a larger application, and make it easy for the user to tailor the application to their specific needs, then there may be a market for such tools. However, for applications that involve user interaction, assessment becomes much harder. Such evaluations cannot be run "off-line," and require having subjects directly involved. This requires the collection of larger quantities of data, to better control subject variability. It is possible that challenge evaluations are not the right venue for evaluation of such integrated tool sets and end-to-end systems. It may be that the marketplace "evaluates" these by buying or not buying them.

The second related area that needs to be addressed is the rapid tailoring to new applications. To date, most commercial systems require some tinkering or consulting with a developer to get optimal performance. Systems for text mining in biology will present similar challenges, and failure to address this issue will prevent tools from being commercially successful. No two applications are ever quite the same, and the data also changes over time. Easy adaptability is critical to the success of text mining tools in any domain.

The following resources are necessary to continue progress in these areas:

- Funding to run the assessments.
- Infrastructure (e.g., test sets, evaluation protocols, lexical resources), which would be contributed by the biology community, who would see the value in having a large community of researchers work on *their* problems.
- Evaluation Services. If utilities provided by the research community can be made readily available, these can be used to provide ongoing evaluation services, which can act as a beneficial source of feedback. In addition, this would provide the opportunity for tool vendors to gain visibility and market share.
- An active research community to advocate for, and participate in, such assessments.

One critical question is the extent to which the commercial market will do this on its own. In a mature market, consumers can evaluate products and make their own cost/benefit trade-offs, although there is still a strong role for independent "consumer report" evaluations. In an emerging technology area, if past experience is any guide, the commercial tools developers will not allocate scarce resources to evaluation. However, based on experience with TREC, commercial companies may be willing to participate if they can learn about the state of the art (and the state of the competition). This underscores the role that evaluation plays in the creation of de facto standards, the creation of a knowledgeable set of consumers and systems integrators, and, eventually, the creation of a market for text mining tools, particularly as they are embedded in biologically useful end-to-end applications.

References

[1] Johnson, S., (ed.), *Proceedings of the ACL-02 Workshop on Natural Language Processing in the Biomedical Domain*, Philadelphia, PA, 2002.

[2] Hirschman, L., and J. Pustejovsky, (eds.), *Proceedings of the HLT/NAACL-2004 Workshop on Linking Biological Literature, Ontologies and Databases: Tools for Users*, Boston, MA, 2004.

[3] Ananiadou, S., and J. Tsujii, (eds.), *Proceedings of the ACL-03 Workshop on Natural Language Processing in Biomedicine*, Sapporo, Japan, 2003.

[4] Voorhees, E., "Overview of TREC 2003," *Proc. 12th Text Retrieval Conference (TREC 2003)*, Gaithersburg, MD, NIST Special Publication: SP 500-255, 2003, pp. 54–68.

[5] Hersh, W., and R. T. Bhupatiraju, "TREC Genomics Track Overview," *Proc. 12th Text Retrieval Conference (TREC 2003)*, Gaithersburg, MD, NIST Special Publication: SP 500-255, 2003, pp. 14–23.

[6] Hersh, W., et al., "TREC 2004 Genomics Track Overview," *Proc. 13th Text Retrieval Conference (TREC 2004)*, Gaithersburg, MD, NIST Special Publication, 2004, http://trec.nist.gov/pubs/trec13/papers/GEO.OVERVIEW.pdf, accessed June 2005.

[7] Blaschke, C., L. Hirschman, and A. Valencia, "BioLINK," 2005, http://www.pdg.cnb.uam.es/BioLink/, accessed June 2005.

[8] Joshi, A., (ed.), *Proceedings of the Workshop on Language Modeling of Biological Data*, University of Pennsylvania, 2001. http://www.ircs.upenn.edu/modeling2001/attendee.shtml, accessed June 2005.

[9] Tsujii, J.-I., (ed.), *Proceedings of the Workshop on Natural Language Processing and Ontology Building*, Tokyo, University of Tokyo, 2002.

[10] Joshi, A., and L. Hirschman, (eds.), "Proceedings of the Special Session on Language Processing and Biological Data," *Human Language Technology Conference*, San Diego, CA, Association of Computational Linguistics, 2002.

[11] Hirschman, L., et al., "Accomplishments and Challenges in Literature Data Mining for Biology," *Bioinformatics*, Vol. 18, No. 12, 2002, pp. 1553–1561.

[12] Venclovas, C., et al., "Assessment of Progress over the CASP Experiments," *Proteins*, Vol. 53, Suppl. 6, 2003, pp. 585–595.

[13] Johnson, K. F., and S. M. Lin, "Critical Assessment of Microarray Data Analysis: The 2001 Challenge," *Bioinformatics*, Vol. 17, No. 9, 2001, pp. 857–858.

[14] Wodak, S. J., and R. Mendez, "Prediction of Protein-Protein Interactions: The CAPRI Experiment, Its Evaluation and Implications," *Curr. Opin. Struct. Biol.*, Vol. 14, No. 2, 2004, pp. 242–249.

[15] Reese, M. G., et al., "Genome Annotation Assessment in Drosophila Melanogaster," *Genome Res.*, Vol. 10, No. 4, 2000, pp. 483–501.

[16] Almasy, L., et al., "Genetic Analysis Workshop 13: Introduction to Workshop Summaries," *Genet. Epidemiol.*, Vol. 25, Suppl. 1, 2003, pp. S1–S4.

[17] Helma, C., and S. Kramer, "A Survey of the Predictive Toxicology Challenge 2000–2001," *Bioinformatics*, Vol. 19, No. 10, 2003, pp. 1179–1182.

[18] Yeh, A. S., L. Hirschman, and A. A. Morgan, "Evaluation of Text Data Mining for Database Curation: Lessons Learned from the KDD Challenge Cup," *Bioinformatics*, Vol. 19, Suppl. 1, 2003, pp. i331–i339.

[19] Hirschman, L., et al., "Overview of BioCreAtIvE: Critical Assessment of Information Extraction for Biology," *BMC Bioinformatics*, Vol. 6, Suppl. 1, 2005, p. S1.

[20] Ashburner, M., et al., "Gene Ontology: Tool for the Unification of Biology. The Gene Ontology Consortium," *Nat. Genet.*, Vol. 25, No. 1, 2000, pp. 25–29.

[21] Kim, J., et al., "Introduction to the Bio-Entity Recognition Task at JNLPBA," *Proc. Joint Workshop on Natural Language Processing in Biomedicine and Its Applications*, Geneva, Switzerland, 2004, pp. 70–75.

[22] Hirschman, L., and I. Mani, "Evaluation," in *Oxford Handbook of Computational Linguistics*, R. Mitkov, (ed.), Oxford: Oxford University Press, 2003, pp. 415–430.

[23] Pallett, D., J. Garofolo, and J. Fiscus, "Measurements in Support of Research Accomplishments," *Communications of the ACM*, Vol. 43, No. 2, 2000, pp. 75–79.

[24] Craven, M., and J. Kumlien, "Constructing Biological Knowledge Bases by Extracting Information from Text Sources," *Proc. 7th International Conference on Intelligent Systems for Molecular Biology*, 1999, pp. 77–86.

[25] Gelbart, W. M., et al., "FlyBase: A Drosophila Database, The FlyBase Consortium," *Nucleic Acids Res.*, Vol. 25, No. 1, 1997, pp. 63–66.

[26] Blake, J. A., et al., "MGD: The Mouse Genome Database," *Nucleic Acids Res.*, Vol. 31, No. 1, 2003, pp. 193–195.

[27] Christie, K. R., et al., "Saccharomyces Genome Database (SGD) Provides Tools to Identify and Analyze Sequences from Saccharomyces Cerevisiae and Related Sequences from Other Organisms," *Nucleic Acids Res.*, Vol. 32, Database Issue, 2004, pp. D311–D3114.

[28] Apweiler, R., et al., "UniProt: The Universal Protein Knowledgebase," *Nucleic Acids Res.*, Vol. 32, Database Issue, 2004, pp. D115–D119.

[29] Bader, G. D., D. Betel, and C. W. Hogue, "BIND: The Biomolecular Interaction Network Database," *Nucleic Acids Res.*, Vol. 31, No. 1, 2003, pp. 248–250.

[30] *Nucleic Acids Research*, Vol. 33, Suppl. 1, Database Issue, Oxford, England: Oxford University Press, 2005.

[31] Harman, D., "Overview of the Third Text REtrieval Conference (TREC-3)," Gaithersburg, MD, NIST Special Publication SP 500-225, 1995, pp. 1–20.

[32] Yeh, A., et al., "BioCreAtIvE Task 1A: Gene Mention Finding Evaluation," *BMC Bioinformatics*, Vol. 6, Suppl. 1, 2005, p. S2.

[33] Hirschman, L., et al., "Overview of BioCreAtIvE Task 1B: Normalized Gene Lists," *BMC Bioinformatics*, Vol. 6, Suppl. 1, 2005, p. S11.

[34] Blaschke, C., et al., "Evaluation of BioCreAtIvE Assessment of Task 2," *BMC Bioinformatics*, Vol. 6, Suppl. 1, 2005, p. S16.

[35] Tanabe, L., et al., "GENETAG: A Tagged Corpus for Gene/Protein Named Entity Recognition," *BMC Bioinformatics*, Vol. 6, Suppl. 1, 2005, p. S3.

[36] Colosimo, M., et al., "Data Preparation and Interannotator Agreement: BioCreAtIvE Task 1B," *BMC Bioinformatics*, Vol. 6, Suppl. 1, 2005, p. S12.

[37] Camon, E., et al., "An Evaluation of GO Annotation Retrieval for BioCreAtIvE and GOA," *BMC Bioinformatics*, Vol. 6, Suppl. 1, 2005, p. S17.

[38] Kim, J. D., et al., "GENIA Corpus—A Semantically Annotated Corpus for Bio-Text Mining." *Bioinformatics*, Vol. 19, Suppl. 1, 2003, pp. i180–i182.

[39] Dingare, S., et al., "A System for Identifying Named Entities in Biomedical Text: How Results from Two Evaluations Reflect on Both the System and the Evaluation," *Comparative and Functional Genomics*, Vol. 6, No. 1–2, 2005, pp. 77–85.

[40] Finkel, J., et al., "Exploring the Boundaries: Gene and Protein Identification in Biomedical Text," *BMC Bioinformatics*, Vol. 6, Suppl. 1, 2005, p. S5.

10

Integrating Text Mining with Data Mining

See-Kiong Ng

10.1 Introduction: Biological Sequence Analysis and Text Mining

Today's biologists and bioinformaticians must operate effectively in a dual environment that is composed of, on the one hand, a data world, which encompasses high-throughput biological data, and, on the other hand, a knowledge world, which contains a tremendous amount of domain information encased in free text. Data mining provides a computational bridge, leading from the ever-expanding biological data world into the biological knowledge world, while text mining provides the computational means to navigate within the vast biological knowledge world currently encapsulated by the ever-growing unstructured scientific literature. Integrating these two structurally disparate worlds for biological data analysis could be advantageous. The analysis of datasets produced by large-scale biological assays would be greatly facilitated if the data mining system were cognizant of the relevant knowledge encoded in the biological literature. However, there is an apparent digital divide between data mining and text mining. The data mining methods that were developed for mining nontext biological data typically expect the source data to be highly structured, while the text mining methods have focused largely on dealing with the unstructuredness of the texts in the scientific literature, as evident from the discussions in previous chapters. In this chapter, we describe various attempts to bridge the gap by employing text as an integrated knowledge source to help guide the mining of nontext biological data. We focus on the analysis of two representative classes of

nontext biological data, namely biological sequences and genetic expression profiles.

The genome sequencing technologies developed during the genome era have allowed us to read our "book of life," or genome, letter by letter. The genome of an organism refers to the entire complement of DNA in any of its cells. The DNA contains the genetic recipes for making proteins that are responsible for carrying out most biological functions in the cells.

A common set of DNA molecules is found in the brain or nucleus of each of the cells in our body. Together, these DNA molecules encode the genetic instructions needed by the cell to carry out its biological functions. These instructions are encoded in a deceptively simple genetic alphabet that contains only four biological letters (nucleotides): A (adenine), C (cytosine), G (guanine), and T (thymine). For computational purposes, we normally would be dealing with the DNA as a long string made up of the four letters (A, G, C, and T). We often refer to the segment on the DNA that encodes the recipes for a protein, namely a gene. Proteins are large, complex biomolecules composed of long chains of smaller subunits, called amino acids, from a fixed alphabet of 20 different kinds of amino acids. As such, proteins also can be represented by strings of biological letters for computational purposes, just like the DNA.[1] Protein sequences are systematically translated from the genetic sequences in the DNA. A triplet of DNA letters is used to code for each amino acid. As such, biologists often use the terms *gene* and *protein* interchangeably. Figure 10.1 gives an example of this coding.

In the cell, the genetic recipes encoded in the DNA are not directly translated into their respective proteins. Instead, a two-step transcription-translation procedure, also known as the "Central Dogma of Molecular Biology," applies (see Figure 10.2). The gene, containing the recipe for making a protein, is first transcribed from the DNA into intermediary molecules called mRNA (messenger ribonucleic acid). The mRNA molecules are biochemically similar to the DNA. Like the DNA, they also are composed of four different nucleotides (A, C, G, and U), in which the nucleotide U (uracil) in RNA replaces the T (thymine) in the DNA. The primary role of the mRNAs is to facilitate the transfer of genetic information from the nucleus, where the DNA resides, into the cytoplasm of the cell, where the protein molecules are manufactured. This intermediary transcriptional step also provides for the cell to regulate the expression of different genes as it desires. More copies of mRNA are made from the DNA, when a cell needs a higher concentration of a particular protein for its function.

1. Unlike in DNA, the structure and shape of protein molecules (i.e., how the long chain of amino acids folds in three-dimensional space) play an important role in how they function biologically. It is thus important to note that just knowing a protein's linear amino acid sequence is often not enough to understand how the protein works.

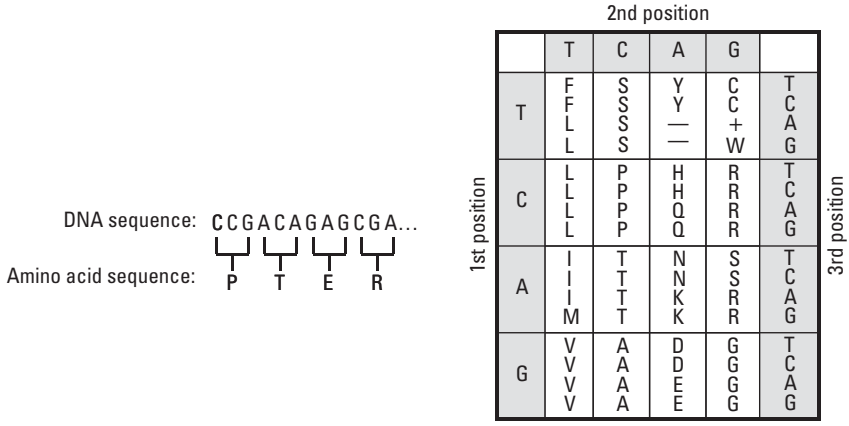


Figure 10.1 The DNA sequence is translated into a sequence of amino acids. Three DNA bases translate to one amino acid, according to the Genetic Code shown in the table on the right. ATG, also known as the "start codon" that signals the initiation of translation, is labeled as "+," while the "stop codons" are labeled as "–."

Through this transcriptional mechanism, the cells in our body are able to switch on different subsets of genes, at different levels and in different cells. While each of the cells contains the same copy of DNA, it can selectively generate different kinds of proteins needed to serve its particular biological function. In the postgenome era (i.e., after many genomes have been completely sequenced, including the human genome), DNA microarray technology allows scientists to simultaneously measure the expression levels of different genes in the cells. These genetic snapshots of cells in different conditions and localizations can provide great insights about the response of various genes to different needs in different situations. We will describe how text mining can be employed to help in the data mining of gene expression profiles in Section 10.2. In this section, we focus on the data mining of biological sequences.

Genetic sequences were the first major class of biological data in the genome era for which computer science—or rather, bioinformatics—had played a major role in their analyses. The sequence information generated by the various large-scale genome sequencing projects offered great opportunities for biological investigations at the molecular level. The previously unreadable book of life can now be scanned for new genes and proteins. The trophy challenge is functional genomics, which involves deciphering the biological function of the new genes or proteins identified.

One sensible way to hypothesize the biological function of a new gene is to find another gene, possibly from another organism, whose function is already known, and to which the new gene has high sequence similarity. This is called

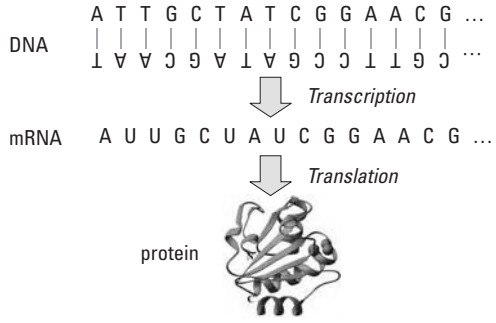


Figure 10.2 The central dogma of biology: Information flows from DNA to RNA to proteins. (Protein structure image from: [1].)

the sequence homology approach. Another way for predicting the biological functions of new genes and proteins is to apply supervised machine learning approaches, to learn about the characteristic features in the biological sequences in each functional class, which can be used to infer the functional classes for unknown genes and proteins. In Section 10.1.1, we show how text mining can be incorporated into sequence homology searches to yield improved accuracies. In Section 10.1.2, we show how text mining can be incorporated with machine learning approaches to achieve better performance in sequence-based functional classification.

10.1.1 Improving Homology Searches

Two sequences are homologous if they share a common evolutionary ancestry. Given that nature is a "tinkerer" and not an "inventor," as aptly put by Jacob [2], new biological sequences are often adapted from preexisting sequences, rather than newly invented. Homology searches can therefore be implemented by looking for significant similarity between the sequences of genes or proteins, to infer their biological relatedness,[2] which in turn reduces into the classic sequence alignment problem in computer science.

In practice, sequence homology searches are possible for two reasons.

1. The global research community has been sharing its sequence data by making them available online via GenBank [3] and other databases.
2. Sequence alignment programs, such as FASTA [4], BLAST [5], and Position Specific Iterated BLAST (PSI-BLAST) [6], have been

2. It is important to note that the assumption that "sequence similarity implies functional similarity" does not always hold, and that not all homologues exhibit sequence similarity.

developed specifically for detecting biological sequence similarity, allowing any newly discovered DNA or protein sequence to be efficiently compared with all known sequences.

Homology searches have been used for function prediction [7, 8], genome annotation [9, 10], and even structure prediction [11, 12]. In fact, sequence homology is perhaps the most widely used bioinformatics tool. For example, the NCBI BLAST server for homology search is estimated to be queried more than 100,000 times per day, and this rate is growing from 10% to 15% per month [12].

In integrating text mining with data mining, two groups [13, 14] have recently attempted to improve homology search, by applying text analysis to database annotations and MEDLINE references, alongside PSI-BLAST. PSI-BLAST has been recognized as one of the most powerful tools for detecting remote evolutionary relationships by sequence considerations only, while the original BLAST program could find only local alignments without gaps. As the name suggests, PSI-BLAST is an iterative version of BLAST. Through iterations, PSI-BLAST increases the sensitivity of the homology searches, so that distant similarity to a query sequence can be detected. Figure 10.3 shows the flowchart for PSI-BLAST. In the first iteration, a BLAST search obtains an initial family of related sequences that are significantly similar to the query sequence. A statistical model is then constructed to represent this family of sequences, by processing the multiple alignment from the BLAST output data into a position-specific score matrix (PSSM). In subsequent iterations, such probabilistic profiles are used to search the sequence database, to add new homologues for output. In this way, more diverse sequences can be incorporated into the query, improving the sensitivity of the homology search with respect to the remote homologues. However, the iterative procedure in PSI-BLAST also may lead to the "profile drift" effect. Since PSI-BLAST allows more diverse sequences to be included as seed sequences for the homology search in its iterations, the possibility of some nonhomologous sequences being introduced into the query set also increases over time. These errors are magnified as the iteration continues, and they dilute the signal from the original query sequence.

MacCallum et al. [15] were one of the first groups to incorporate text mining with PSI-BLAST for improved sequence homology searches. Their algorithm, known as Structure Assignment with Text Description (SAWTED), tapped into the textual information in the expert annotations available in the curated protein database SWISS-PROT [16]. This allowed a search for likely remote homologues, while filtering away potentially false positives that have no such textual support. In SAWTED, the literature of a protein sequence is defined as the concatenation of words found in the corresponding SWISS-PROT database entry for the protein (or a close homologue) under the

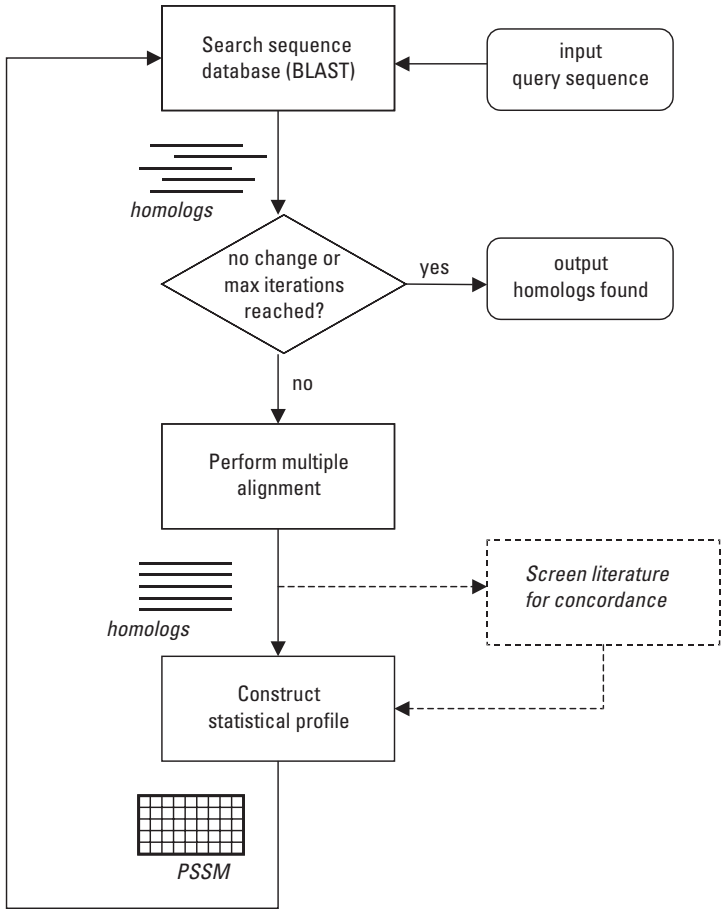


Figure 10.3 Computational steps of PSI-BLAST. The additional integrative text mining step depicted in dashed line was proposed by Chang et al. [14].

database's annotation fields, such as the Reference Title (RT), Comments (CC), and Keywords (KW). The basic vector-cosine model from text retrieval is then used to compute the literature similarity (c_{AB}) between two proteins P_A and P_B with database textual annotations A and B:

$$c_{AB} = \cos\theta = \frac{\nu^{A}\nu^{B}}{\left|\nu^{A}\right\|\nu^{B}\right|} \tag{10.1}$$

Using the literature similarity scores as a postfilter for PSI-BLAST, MacCallum et al. found fewer errors in their benchmark homology searches. Encouraged by this improvement, they also attempted to integrate text mining

within PSI-BLAST's search process, instead of using text mining merely as a postfilter, by combining the literature scores c_{AB} with the *E*-values used by PSI-BLAST to rank the homologous sequences. Unfortunately, in their case, the combined score did not perform as well as the simple use of text scores for the PSI-BLAST hits.

In more recent work, Chang et al. [14] reexamined the problem of integrating text mining within PSI-BLAST searches. Instead of combining the two mining approaches at the confidence scoring level, their modification to PSI-BLAST involved an additional step of throwing out sequences that have poor literature similarity to the query sequence in the iterative process, as shown by the dotted lines in Figure 10.3. They also employed a literature richer than SAWTED for each protein sequence. In addition to using the descriptions, comments, and keywords found in the respective SWISS-PROT database records, they downloaded the corresponding literature citations with the MeSH headings, subheadings, and abstracts from MEDLINE, to form a more complete concatenated literature corpus for each protein. At each iteration of their modified PSI-BLAST search, the significant hits are ranked according to a literature similarity score that is similar to c_{AB} above. The lowest scoring fractions are discarded, excluding them from the query profile. Potential false positives are eliminated from the profile in this way, and the resulting effect of profile drift is reduced.

Chang et al. evaluated their modified algorithm, by comparing to standard PSI-BLAST in searching for homologous proteins. They reported that the performance of their modified algorithm achieved 32% recall with 95% precision, while the standard PSI-BLAST achieved 33% recall with 84% precision. Their results indicate that incorporating the literature similarity requirement into the homology search process improves the precision of the PSI-BLAST algorithm, without compromising the desired sensitive characteristic of the PSI-BLAST algorithm.

10.1.2 Improving Sequence-Based Functional Classification

In sequence homology searches, the problem of determining the biological functions of a new gene or protein is formulated as a sequence matching problem. Functional information from well-studied genes or proteins is transferred to uncharacterized gene products, based on sequence similarity. In this section, we treat the problem of determining the biological function of new genes as a classification problem in machine learning. We can apply supervised machine learning methods to predict the biological functions of new genes and proteins.

We focus here on the sequence-based classification of subcellular locations of proteins. In order to carry out its physiological role, a protein usually must be proximal to other biological components involved in that process. A protein's

localization in the cell is tightly bound to its biological function. Knowing the subcellular location where a protein resides can give important insights as to its possible function, especially in the case where functional assignment by sequence homology has failed because there was no clear homology to existing proteins of known function.

Biologists have long observed that there is a significant correlation between the amino acid composition of a protein and its subcellular localization [17, 18]. The subcellular localization of a protein can be predicted with reasonable accuracy from its amino acid composition [19–21].[3] Stapley et al. [22] attempted to further improve on this problem by integrating text mining with data mining. They experimented with the prediction of the subcellular localization for yeast proteins, based on both sequence and literature information about the proteins.

- *Sequence information.* The amino acid composition for a protein is computed as the fractional composition of the 20 amino acids in the protein sequence.
- *Literature information.* For each yeast protein, MEDLINE is scanned for literature abstracts that mention the protein name or a synonym. After applying the necessary stop word removal, stemming, and the removal of stemmed terms that occurred in few documents (less than 5) to clean up the corpus, a word vector, based on a variant of inverse document frequency (IDF), is created for each protein.

A supervised machine learning method—support vector machines (SVM) [23]—is then trained, using the word vector, together with the amino acid compositions for each protein in the training set. The resulting SVM was found to give predictions on subcellular localizations of proteins that are more accurate than by using SVMs trained with either the amino acid composition information or the text information alone [see Figure 10.4(a)]. This improvement was noted for 5 out of the 11 subcellular localization classes studied by Stapley et al. Interestingly, for another 3 of the 11 classes studied, combining text with sequence data had showed no improvement in subcellular localization classification, while for the remaining 3 classes, the joint approach resulted in classification results that were actually worse. These results suggest that a better way to combine text data with sequence data for supervised machine learning must be

3. It is also possible to predict protein subcellular location based on the existence of sorting signals in *N*-terminal sequences [20], such as signal peptides, mitochondrial targeting peptides, and chloroplast transit peptides. In fact, Nakai et al. [21] constructed an expert system for subcellular localization, using both sorting signals and amino acid compositions.

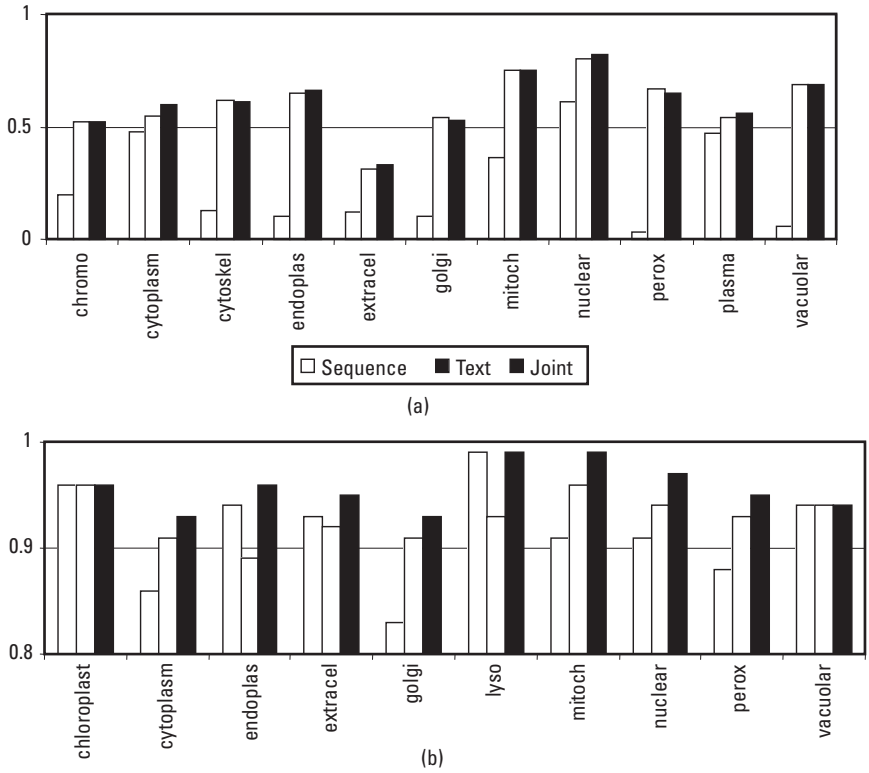


Figure 10.4 Improving sequence-based subcellular localization of proteins with text mining. (a) Classification results in terms of F1 scores by Stapley et al. [22]. (b) Classification results in terms of ROC_{50} scores by Eskin and Agichtein [24].

employed, in order to fully exploit the benefit of including text as the additional knowledge source for sequence data mining.

In building an SVM, input sequences are implicitly mapped into a higher n-dimensional vector space, where the coordinates are given by feature values, using a kernel function $K(x_i, x_j) = \phi(x_i)^T \phi(x_j)$, where ϕ is a function that maps the input vectors x_i into the higher dimensional feature space. The SVM then produces an optimal linear decision boundary in the high-dimensional feature space, which discriminates between the positive and negative classes. This kernel model in SVM provides an excellent way for integrating data mining and text mining. In their recent work, Eskin and Agichtein [24] proposed an approach in which the text and sequence data are intimately integrated at the kernel level of SVMs.

Eskin and Agichtein devised a text-only SVM classifier that used a text-based kernel $K_{text}(x, y)$, and a sequence-only SVM classifier that used a sequence-based kernel $K_{seq}(x, y)$, as follows.

- *Text Kernel.* To represent the literature associated with a protein, they used a bag-of-words model, where the text annotation x obtained from the corresponding SWISS-PROT entry is mapped to a feature vector $\phi_{text}(x)$, containing the frequency of each word. For two text annotations x and y, the text kernel is denoted as $K_{text}(x, y) = \phi_{text}(x) \cdot \phi_{text}(y)$.
- *Sequence Kernel.* Instead of using amino acid composition, as in the work by Stapley et al., Eskin and Agichtein represented each of their input protein sequences as a collection of substrings of a fixed length k (i.e., k-mers), obtained by sliding a window of length k across the length of the input sequence. The sequence feature vector $\phi_{seq}(x)$ is of a high dimension of 20^k and contains the frequency of each possible k-mer that occurs in the input protein sequence x. For two input protein sequences x and y, the sequence kernel is denoted as $K_{seq}(x, y) = \phi_{seq}(x) \cdot \phi_{seq}(y)$.

To combine text and sequences in the analysis, Eskin and Agichtein then simply defined a combined kernel as follows:

$$K_{combined}(x, y) = K_{text}(x, y) + K_{seq}(x, y) + \left[K_{text}(x, y) + K_{seq}(x, y)\right]^2 \quad (10.2)$$

An SVM then can be trained over the combined feature space using this combined kernel, allowing the resulting classifier to effectively learn from both sequence and text, and the interactions between them.

To verify whether such an integration of text mining with data mining is indeed advantageous for sequence-based functional classification of proteins, Eskin and Agichtein ran three experiments, by leaving out 20% of the original annotated sequence data as a test set, and using the remaining data as a training set. They trained three models on the training set: a text-only classifier, a sequence-only classifier, and a joint sequence-text classifier, using $K_{text}(x, y)$, $K_{seq}(x, y)$, and $K_{combined}(x, y)$, respectively. Their results, as depicted in Figure 10.4(b), showed that their joint classifier was able to achieve better results in 7 out of the 10 subcellular localization classes that they have investigated, indicating that better results can indeed be obtained by more careful and deeper integration of text mining with the specific data mining processes.

10.2 Gene Expression Analysis and Text Mining

Another source of biological data that has recently become widely available, and which provides genome-scale answers to functional genomics questions, is expression data. As mentioned earlier, gene expression analysis measures the

relative levels of expression or production of mRNA in the cell. Since different genes may be active in different cell types of an organism at different times and under different conditions, the level of mRNA molecules for the various genes in a cell can provide valuable information about the biochemical activity in the cell.

Gene expression data are typically collected using microarrays [25]. A microarray is a tiny chip made of a solid support, such as glass or silicon, onto which DNA molecules are immobilized at fixed spots. With the help of robotic technologies, tens of thousands of spots, each containing a large number of identical DNA molecules, can be placed on a small array (e.g., less than 1 in^2). When the mRNA from a sample cell is passed over the chip, it will bind to the appropriate spots, due to the complementary base pairing. This makes it possible to perform a systematic whole-genome scan on a single chip, where each of these spots would contain a unique DNA fragment that identifies a gene in the genome. In this way, DNA microarray technology allows us to take high resolution snapshots of the genetic expression of different genes in our cells, in different conditions and localization. Such concurrent monitoring of gene expression helps reveal how various genes respond to different needs.

Table 10.1 shows a fragment of the data table of gene expression measurements from a microarray experimental study to identify the regulatory genes in a yeast cell cycle [26]. Here, the expression levels of the genes in yeast (approximately 6,000) were measured using microarrays at 21 successive time points, by taking samples at seven-minute intervals from a population of synchronized yeast cells. The resulting data is a multivariate matrix, with 6,000 rows (entries) and 21 columns (variables). The expression profile of a gene refers to the set (row) of expression measurements for that gene in the microarray study.

To detect groups of genes that are potentially involved in a common biological process, clustering algorithms can be employed to group genes with similar expression profiles. Most array clustering algorithms either use the statistical correlation coefficient (ranging from −1 to +1), or the Euclidean distance (the square root of the sum of the squared differences in corresponding features

Table 10.1
Example Gene Expression Data

Gene	0 min	7 min	14 min	21 min	28 min	...
YER150W	0.41	1.47	1.80	0.81	0.03	...
YGR146C	0.78	0.37	0.09	0.07	0.03	...
YDR461W	2.36	2.35	2.30	2.11	1.75	...
:	:	:	:	:	:	

values), as a distance metric to compute the similarity between two profiles. Traditional clustering methods based on hierarchical clustering [27] or self-organizing maps [28], as well as more advanced stochastic clustering techniques [29], have been used to derive putative functional clusters of genes from expression profile data [30]. By applying the guilt-by-association principle, expression profile clustering also can be used for inferring the biological functions of new genes. If an uncharacterized gene is clustered with a group of genes known to participate in a specific biological process (e.g. cell cycle regulation), then it is assumed that the uncharacterized gene also participates in this process.

However, genes sharing similar expression profiles do not always share a common function. For example, in the work reported by Spellman et al. [27], clustering by expression profiles grouped the genes *CHS2*, *BUD8*, *IQG1*, *ACE2*, *ALK1*, and *HST3* into a single CLB2 cluster, even though they are involved in distinct cellular functions.[4] The reverse is also true: not all the genes in the same function group necessarily exhibit simultaneous expression. For example, the members of a signaling pathway often play antagonistic roles, resulting in anticorrelated expression levels in microarray experiments. The gene expression clustering approach should not be used as a stand-alone analysis tool for functional genomics. In this section, we show how text mining can help address this need, by being integrated with gene expression data mining methods to provide the necessary biological knowledge in intelligent expression profile analyses.

10.2.1 Assigning Biological Explanations to Gene Expression Clusters

With microarray technologies, scientists can now measure the expression of thousands of genes simultaneously. With data mining algorithms, they can organize the massive data generated by clustering subsets of genes whose expression changes with correlated intensities. However, clustering analysis does not reveal the underlying biological significance of the gene groupings. In fact, as we have mentioned earlier, it does not always lead to biologically meaningful gene clusters. Interpretation of the biological basis for the observed genetic expression patterns is therefore left to the biologists, who must try to provide reasonable biological explanations for the statistically-derived gene expression clusters, which is best accomplished by manually inspecting the literature for relevant information. Given the tremendous amount of literature currently available, this is equivalent to "attempting to drink from a fire hose" [31].

In this aspect, text mining can be integrated with gene expression profile mining, to associate the resulting gene clusters with published literature. A good illustrative example is the Gene Expression Information System for Human

4. *CHS2*, *BUD8*, and *IDG1* are involved in cell wall maintenance, while *ACE2*, *ALK1*, and *HST3* are involved in nuclear events, a very disparate cellular process.

Analysis (GEISHA), developed by Blaschke et al. [32], shown in Figure 10.5. For each cluster of similarly expressed genes, MEDLINE abstracts that mentioned at least one gene in the cluster are collected to generate an associated literature cluster. The words in the various literature clusters are then extracted, with the necessary text analysis steps taken to account for morphological variations and composite word terms. The frequency of the terms in the MEDLINE abstracts associated to each cluster is then compared to the frequency of these terms in the other clusters. A term is considered significant to a particular gene cluster if it appears more frequently in the associated literature cluster than in any other clusters, based on its Z-score. In this way, the system can automatically assign a set of significant terms to each gene expression cluster, which is extracted from its underlying literature corpus as the biological explanation of the gene cluster. A similar procedure also is applied in GEISHA to extract complete sentences specific to the various gene clusters, providing more readable biological explanations for the clusters. The authors have extensively compared the results of the GEISHA system to the annotations provided in biological databases, and human experts have verified that this approach was able to extract relevant information about the gene clusters in many cases.

This use of literature annotation to address the analytical challenge of interpreting gene expression clusters has been adopted in many published systems [33–39]. Like GEISHA, many of these systems employ literature to

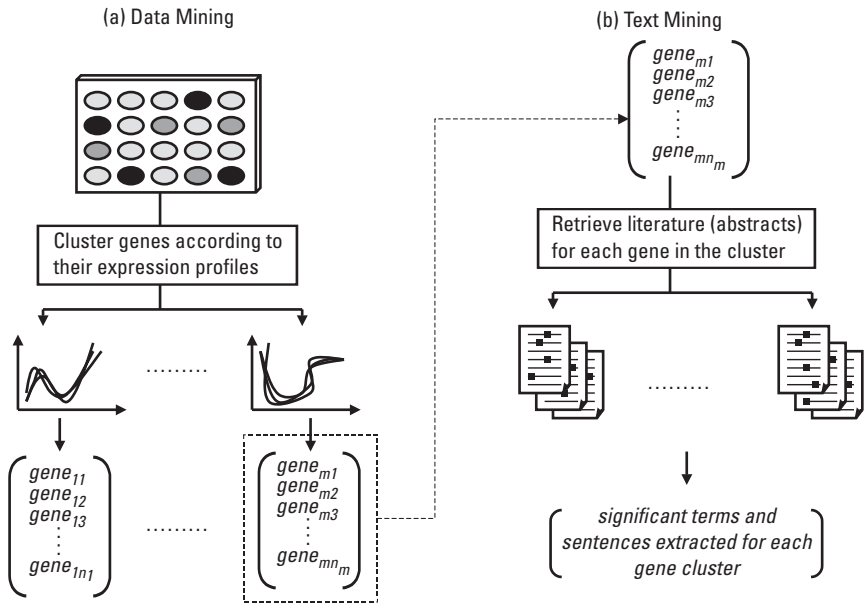


Figure 10.5 Combining text mining with data mining in GEISHA [32].

examine sets of genes only after they already have been grouped by some unsupervised clustering algorithms. In other words, the use of literature is incorporated as a retrospective procedure in the data mining of the gene expression profiles. While this approach can help biologists make sense of some of the expression-based gene clusters, it does not improve the biological coherence of the gene clusters originally generated by the expression clustering algorithms. To directly influence the quality of the gene clusters produced from gene expression data, a data mining algorithm must have the guidance from a direct access to the external background knowledge about the genes.

10.2.2 Enhancing Expression Data Analysis with Literature Knowledge

Several researchers have recently undertaken the challenge of incorporating the literature information as a knowledge base into data mining algorithms to intelligently interpret gene expression data. Raychaudhuri et al. [40] devised a novel clustering algorithm, which employs a literature-based functional coherence metric as the objective function to search for patterns in gene expression data that correlate with meaningful biology. Their literature-based metric is called the neighbor divergence per gene (NDPG) [38, 41]. A semantic network of articles is constructed, based on document distance metrics. An article's relevance to a given gene group is then measured by counting the number of semantic neighbors that referred to genes in the group. If the group represents a coherent biological function, then articles that discuss that function will have many referring neighbors, and therefore score highly under this scheme. Each gene's relative functional relevance to a group is scored by comparing its article scores to an expected random distribution of article scores. The NDPG measure of a gene group is then computed as the mean divergence of all of the genes in the group, which represents the likelihood that the genes share a biological function. NDPG was reported to be 95% sensitive and 100% specific at identifying functional yeast gene groups. Raychaudhuri et al. [42] also showed that, for hierarchical clustering, NDPG can be used to determine effectively which level of the tree to cut to form biologically relevant cluster boundaries. However, like the GEISHA system, this approach still does not fully exploit the knowledge in the associated literature to directly guide the clustering process, since the gene clustering is still done independent of the literature information. As we have seen in the case of PSI-BLAST in Section 10.1.1, where text mining was employed to improve homology searches, a clustering algorithm that is iterative in nature would be quite convenient for integrating text mining into data mining. As such, Raychaudhuri devised a new clustering algorithm called optimizing separating projections (OSP), which iteratively searches for gene expression rules (or criteria) that optimally separate functionally related groups of genes from the rest of the genes, using the NDPG score as the objective function in the iterative

mining process. When this integrative clustering approach was applied to a well-known published yeast gene expression dataset [26], it was found that OSP was able to generate—without any human input—biologically meaningful gene clusters that were comparable to gene clusters carefully screened by human experts. The deep integration of literature information in the data mining process is indeed a promising approach for enhancing gene expression profile clustering.

In more recent work, Glenisson et al. [43] showed that it is possible to fuse at the data level the two disparate worlds of gene expression and biological literature, so that standard clustering algorithms, such as hierarchical clustering methods, can be applied. In their approach, Glenisson et al. transformed the textual domain knowledge from biological literature into a suitable numerical format that can be mathematically combined with gene expression data, for intelligent gene clustering (see Figure 10.6). They used the bag-of-words representation of

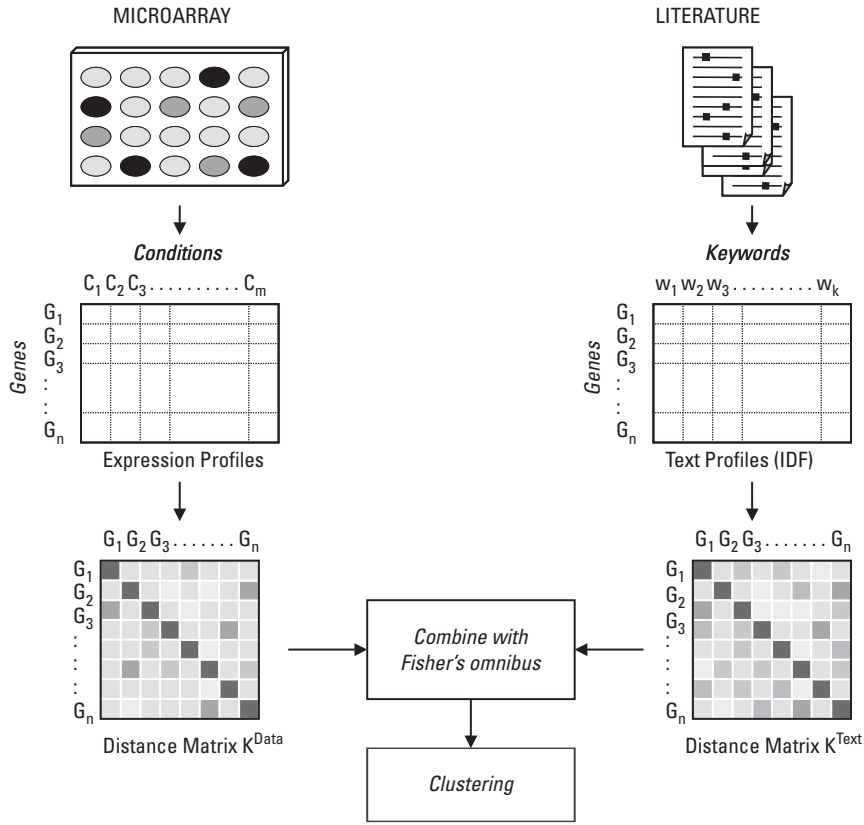


Figure 10.6 Integrating expression and text data in vector space model by mathematically combining the corresponding distance matrices for gene clustering.

literature that has been proven to be highly effective for many biomedical applications. For example, it was used by Eskin et al. to improve sequence-based functional classification by SVM, as described in Section 10.1.2. Here, all the text documents linked to each gene (e.g., as query results from PubMed) are summarized into a document index. This index is based on inverse document frequencies in a vast term space, consisting of a tailored vocabulary of 15,057 (possibly multiword) terms, based on the Gene Ontology [44]. Using this vector space model [45, 46], the associated literature corpus is effectively reduced into a matrix that is structurally similar to the expression data matrix.

Next, the corresponding distance matrices D^{data} and D^{text} for the genes in the expression and literature space are computed from the data matrices (see Figure 10.6). This facilitates a straightforward way to merge the information—the distance information for each pair of genes can be added together in a combined matrix.

$$D^{combined} = (1-\lambda)D^{data} + \lambda D^{text} \tag{10.3}$$

However, the use of the parameter λ to account for the relative importance of the two heterogeneous data sources does not adequately address the scaling issue between the heterogeneous data sources. To address the issue at a more fundamental level, the authors proposed using Fisher's omnibus to statistically combine the data and literature evidence.

First, the entries in the distance matrices are transformed to their respective *p*-values, by computing one-sided cumulative distribution function values for each distance value in D^{data} and D^{text}. This frees the subsequent steps from the bias due to the underlying distributions from which the data was generated. The *p*-values derived for each expression-based and text-based distance are then mathematically combined with Fisher's omnibus method, using the following:

$$S = -2\log p^{data} - 2\log p^{text} \tag{10.4}$$

The combined statistic *S* follows a χ^2-distribution. For effective joint analysis of gene expression data with literature information, standard clustering then can be applied on the combined distance matrix, using the resulting *p*-values. Glenisson et al. applied their combined mining method on published yeast expression data [26, 46], and verified that such augmentation of expression data with literature information did positively affect the biological significance of the overall clustering results. In addition, the authors found that their combined approach was able to reveal biologically meaningful clusters not identified when using microarray data alone.

10.3 Conclusion

Technological advances during the past decades have drastically changed genomic research. At this opportune time, computing in the form of bioinformatics has accelerated life sciences research, by enabling unprecedented throughput in organized data generation and analysis. The next rate-limiting step is in converting this voluminous data and information, which is stored in both structured and unstructured on-line databases, such as GenBank, SWISS-PROT, and MEDLINE, into useful knowledge that can be used for understanding the underlying mechanisms of diseases, as well as discovering drugs to treat them. The previous chapters in this book focus on text processing methods for mining the wealth of knowledge accumulated by decades of biological research, which is currently embedded in the literature databases. In this chapter, we discuss how literature can be properly incorporated as a knowledge source in the analysis of biological data. Rather than using or linking the data sources independently, the great challenge here lies in facilitating a deep integration of the textual knowledge into the data mining algorithms. By leveraging the valuable information from the literature in the data mining process, more biologically significant knowledge can be intelligently mined from the biological data.

Indeed, as the dual fields of biological text mining and data mining continue to progress, we can expect more cross-marriages between the two in the future. For example, a novel literature database mining approach, which handles text and chemistry in an integrated fashion, was recently proposed to address the problem of knowledge-based chemical structural similarity searches [48]. Textual medical records also may be mined, together with patients' genomic information, for pharmacogenomic applications, as suggested by Chang et al. [14]. Through the integration of text mining with data mining, the opportunity for text analysis to benefit biology is particularly compelling, since it facilitates a much more involved role for text mining in the various postgenome processes of biological knowledge discovery.

References

[1] National Institute for General Medical Sciences, http://www.nigms.nih.gov/psi/image_gallery/structures.html#.

[2] Jacob, F., "Evolution and Tinkering," *Science*, Vol. 196, No. 4295, 1977, pp. 1161–1166.

[3] Benson, A. D., et al., "Genbank: Update," *Nucleic Acids Res.*, Vol. 32, 2004, Database Issue, D23–26.

[4] Pearson, W. R., and D. J. Lipman, "Improved Tools for Biological Sequence Comparison," *Proc. Natl. Acad. Sci. USA*, Vol. 85, No. 8, 1988, pp. 2444–2448.

[5] Altschul, S. F., et al., "Basic Local Alignment Search Tool," *J. Mol. Biol.*, Vol. 215, No. 3, 1990, pp. 403–410.

[6] Altschul, S. F., et al., "Gapped Blast and Psi-Blast: A New Generation of Protein Database Search Programs," *Nucleic Acids Res.*, Vol. 25, No. 17, 1997, pp. 3389–3402.

[7] Gabaldon, T., and M. A. Huynen, "Prediction of Protein Function and Pathways in the Genome Era," *Cell Mol. Life Sci.*, Vol. 61, No. 7–8, 2004, pp. 930–944.

[8] Hannenhalli, S. S., and R. B. Russell, "Analysis and Prediction of Functional Subtypes from Protein Sequence Alignments," *J. Mol. Biol.*, Vol. 303, No. 1, 2000, pp. 61–76.

[9] Gopal, S., et al., "Homology-Based Annotation Yields 1,042 New Candidate Genes in the Drosophila Melanogaster Genome," *Nat. Genet.*, Vol. 27, No. 3, 2001, pp. 337–340.

[10] Wilson, C. A., J. Kreychman, and M. Gerstein, "Assessing Annotation Transfer for Genomics: Quantifying the Relations Between Protein Sequence, Structure and Function Through Traditional and Probabilistic Scores," *J. Mol. Biol.*, Vol. 297, No. 1, 2000, pp. 233–249.

[11] Huynen, M., et al., "Homology-Based Fold Predictions for Mycoplasma Genitalium Proteins," *J. Mol. Biol.*, Vol. 280, No. 3, 1998, pp. 323–326.

[12] McGuffin, L. J., K. Bryson, and D. T. Jones, "The PSIPRED Protein Structure Prediction Server," *Bioinformatics*, Vol. 16, No. 4, 2000, pp. 404–405.

[13] Brown, D. G., M. Li, and B. Ma, "Homology Search Methods," in *The Practical Bioinformatician*, L. S. Wong, (ed.), Singapore: World Scientific, 2004, pp. 217–244.

[14] Chang, J. T., S. Raychaudhuri, and R. B. Altman, "Including Biological Literature Improves Homology Search," *Proc. Pacific Symp. on Biocomputing*, 2001, pp. 374–383.

[15] MacCallum, R. M., L. A. Kelley, and M. J. Sternberg, "SAWTED: Structure Assignment with Text Description–Enhanced Detection of Remote Homologues with Automated Swiss-Prot Annotation Comparisons," *Bioinformatics*, Vol. 16, No. 2, 2000, pp. 125–129.

[16] Bairoch, R., and R. Apweiler, "The Swiss-Prot Protein Sequence Database and Its Supplement Tremble in 2000," *Nucleic Acids Res.*, Vol. 28, No. 1, 2000, pp. 45–48.

[17] Cedano, J., et al., "Relation Between Amino Acid Composition and Cellular Location of Proteins," *J. Mol. Biol.*, Vol. 266, No. 3, 1997, pp. 594–600.

[18] Nishikawa, K., and T. Ooi, "Correlation of the Amino Acid Composition of a Protein to its Structural and Biological Characters," *J. Biochem. (Tokyo)*, Vol. 91, No. 5, 1982, pp. 1821–1824.

[19] Reinhardt, A., and T. Hubbard, "Using Neural Networks for Prediction of the Subcellular Location of Proteins," *Nucleic Acids Res.*, Vol. 26, No. 9, 1998, pp. 2230–2236.

[20] Nakai, K., "Protein Sorting Signals and Prediction of Subcellular Localization," *Adv. Protein Chem.*, Vol. 54, 2000, pp. 277–344.

[21] Nakai, K., and M. Kanehisa, "A Knowledge Base for Predicting Protein Localization Sites in Eukaryotic Cells," *Genomics*, Vol. 14, No. 4, 1992, pp. 897–911.

[22] Stapley, B. J., L. A. Kelley, and M. J. Sternberg, "Predicting the Sub-Cellular Location of Proteins from Text Using Support Vector Machines," *Proc. Pacific Symp. on Biocomputing*, 2002, pp. 374–385.

[23] Vapnik, V. N., *Statistical Learning Theory*, Berlin, Germany: Springer-Verlag, 1998.

[24] Eskin, E., and E. Agichtein, "Combining Text Mining and Sequence Analysis to Discover Protein Functional Regions," *Proc. Pacific Symp. on Biocomputing*, 2004, pp. 288–299.

[25] Gerhold, D., T. Rushmore, and C. T. Caskey, "DNA Chips: Promising Toys Have Become Powerful Tools," *Trends Biochem. Sci.*, Vol. 24, No. 5, 1999, pp. 168–173.

[26] Eisen, M. B., et al., "Cluster Analysis and Display of Genome-Wide Expression Patterns," *Proc. Natl. Acad. Sci. USA*, Vol. 95, No. 25, 1998, pp. 14863–14868.

[27] Spellman, P. T. , et al., "Comprehensive Identification of Cell Cycleregulated Genes of the Yeast Saccharomyces Cerevisiae by Microarray Hybridization," *Mol. Biol. Cell*, Vol. 9, No. 12, 1988, pp. 3273–3297.

[28] Tamayo, P., et al., "Interpreting Patterns of Gene Expression with Self-Organizing Maps: Methods and Application to Hematopoietic Differentiation," *Proc. Natl. Acad. Sci. USA*, Vol. 96, No. 6, 1999, pp. 2907–2912.

[29] Ben-Dor, A., R. Shamir, and Z. Yakhini, "Clustering Gene Expression Patterns," *J. Comput. Biol.*, Vol. 6, No. 3–4, 1999, pp. 281–297.

[30] Sherlock, G., "Analysis of Large-Scale Gene Expression Data," *Briefings in Bioinformatics*, Vol. 2, No. 4, 2001, pp. 350–362.

[31] Waldrop, M. M., "Learning to Drink from a Fire Hose," *Science*, Vol. 248, No. 4956, 1990, pp. 674–675.

[32] Blaschke, C., J. C. Oliveros, and A. Valencia, "Mining Functional Information Associated with Expression Arrays," *Funct. Integr. Genomics*, Vol. 1, No. 4, 2001, pp. 256–268.

[33] Chaussabel, D., and A. Sher, "Mining Microarray Expression Data by Literature Profiling," *Genome Biol.*, Vol. 3, No. 10, 2002, RESEARCH0055.

[34] Glenisson, P., et al., "TXTgate: Profiling Gene Groups with Text-Based Information," *Genome Biol.*, Vol. 5, No. 6, 2004, p. R43.

[35] Hu, X., "Integration of Cluster Ensemble and Text Summarization for Gene Expression Analysis," *Proc. 4th IEEE Symp. of Bioinformatics and Bioengineering*, Taiwan, 2004, pp. 251–258.

[36] Korotkiy, M., et al., "A Tool for Gene Expression–Based PubMed Search Through Combining Data Sources," *Bioinformatics*, Vol. 20, No. 12, 2004, pp. 1980–1982.

[37] Masys, D. R., et al., "Use of Keyword Hierarchies to Interpret Gene Expression Patterns," *Bioinformatics*, Vol. 17, No. 4, 2001, pp. 319–326.

[38] Raychaudhuri, S., et al., "Associating Genes with Gene Ontology Codes Using a Maximum Entropy Analysis of Biomedical Literature," *Genome Res.*, Vol. 12, No. 1, 2002, pp. 203–214.

[39] Shatkay, H., et al., "Genes, Themes, and Microarrays: Using Information Retrieval for Large-Scale Gene Analysis," *Proc. 8th Int. Conf. on Intelligent Systems for Molecular Biology*, 2000, pp. 317–328.

[40] Raychaudhuri, S., H. Schutze, and R. B. Altman, "Inclusion of Textual Documents in the Analysis of Multidimensional Data Sets: Application to Gene Expression Data," *Machine Learning*, Vol. 52, 2003, pp. 119–145.

[41] Raychaudhuri, S., and R. B. Altman, "A Literature-Based Method for Assessing the Functional Coherence of a Gene Group," *Bioinformatics*, Vol. 19, No. 3, 2003, pp. 396–401.

[42] Raychaudhuri, S., et al., "The Computational Analysis of Scientific Literature to Define and Recognize Gene Expression Clusters," *Nucleic Acids Res.*, Vol. 31, No. 15, 2003, pp. 4553–4560.

[43] Glenisson, P., J. Mathys, and B. de Moor, "Meta-Clustering of Gene Expression Data and Literature-Based Information," *ACM SIGKDD Explorations*, Vol. 5, No. 2, 2003, pp. 101–112.

[44] Harris, M. A., et al., "The Gene Ontology (GO) Database and Informatics Resource," *Nucleic Acids Res.*, Vol. 32, 2004, Database issue: D258–261.

[45] Manning, C. M., and H. Schutze, *Foundations of Statistical Natural Language Processing*, Cambridge: MIT Press, 1999.

[46] Baeza-Yates, R., and B. Ribeiro-Neto, *Modern Information Retrieval*, New York: ACM Press/Addison-Wesley, 1999.

[47] Cho, R. J., et al., "A Genome-Wide Transcriptional Analysis of the Mitotic Cell Cycle," *Mol. Cell.*, Vol. 2, No. 1, 1998, pp. 65–73.

[48] Singh, S. B., R. D. Hull, and E. M. Fluder, "Text Influenced Molecular Indexing (TIMI): A Literature Database Mining Approach that Handles Text and Chemistry," *J. Chem. Inf. Comput Sci.*, Vol. 43, No. 3, 2003, pp. 743–752.

Acronyms

ACL	Association for Computational Linguistics
ACM	Association for Computing Machinery
ADJP-PRD	predicate adjectival phrase
API	application programming interface
ARGH	acronym resolving general heuristics
AROC	area under the receiver operator characteristic curve
ATM	automatic terminology management
ATR	automatic term recognition
ATRACT	Automatic Term Recognition and Clustering of Terms
ATS	automatic term structuring
BioCreAtIvE	Critical Assessment of Information Extraction in Biology
BMC	BioMed Central
CAMDA	Critical Assessment of Microarray Data Analysis
CAPRI	Critical Assessment of PRediction of Interactions
CASP	Critical Assessment of Protein Structure Prediction
CC	comments
CCG	Combinatory Categorical Grammar

COLING	International Conference on Computational Linguistics
CR	coreference
CSS	cascading style sheet
CVS	Concurrent Versions System
DCG	Definite Clause Grammar
DL	description logic
DMT	Dictionary of Medical Terms
DRT	Discourse Representation Theory
DTD	document type definition
EBI	European Bioinformatics Institute
EMPathIE	Enzyme and Metabolic Pathways Information Extraction
FMA	Foundational Model of Anatomy
GASP	Genome Annotation Assessment Project
GAW	Genome Access Workshop
GEISHA	Gene Expression Information System for Human Analysis
GeneRIF	Gene Reference into Function
GENIA	Gene Expression Information System for Human Analysis
GO	Gene Ontology
GOA	Gene Ontology Annotation
HGNC	Human Gene Nomenclature Committee
HMM	hidden Markov model
HPSG	Head-driven Phrase Structure Grammar
HUGO	Human Genome Organization
ICD	International Classification of Diseases
IDE	integrated development environment
IDF	inverse document frequency
IE	information extraction

IOB	Inside/Outside/Begin
IPI	International Protein Index
IR	information retrieval
ISMB	Intelligent Systems in Molecular Biology
ITM	intelligent term manager
JNLPBA	International Joint Workshop on Natural Language Processing in Biomedicine and its Applications
KDD	knowledge discovery and data mining
KW	keywords
LCS	Longest Common Substring
MCA/MR	Multiple Congenital Anomaly/Mental Retardation
ME	maximum entropy
MEDLINE	Medical Literature, Analysis, and Retrieval System Online
MeSH	Medical Subject Headings
MGI	Mouse Genome Informatics
ML	machine learning
MMTx	MetaMap
MUC	Message Understanding Conference
NCBI	National Center for Biotechnology Information
NDPG	neighbor divergence per gene
NE	named entity
NER	named entity recognition
NLM	National Library of Medicine
NLP	natural language processing
NP	noun phrase
OBO	Open Biomedical Ontologies
OSP	optimizing separating projections
PASTA	Protein Active Site Template Acquisition

PLoS	Public Library of Science
PMC	PubMed Central
PMID	PubMed unique identifier
POS	part-of-speech
PP	prepositional phrase
PSI-BLAST	Position Specific Iterated BLAST
PSSM	position-specific score matrix
PTB	Penn Treebank II
PTC	Predictive Toxicology Challenge
RLP	robust linear programming
ROC	receiver operating characteristic
RT	reference title
SAWTED	Structure Assignment with Text Description
SBAR	subordinate clause marker
SIB	Swiss Institute of Bioinformatics
SIG	special interest group
SNOMED	Systematized Nomenclature of Medicine
SNOMED CT	Systematized Nomenclature of Medicine Clinical Terms
ST	scenario template
SVM	support vector machine
TE	template element
TEI	Text Encoding Initiative
TEM	transmission electron microscopy
TR	template relation, text retrieval
TREC	Text Retrieval Conference
TSV	tab separated values
UMLS	Unified Medical Language System

UniProt	Universal Protein Resource
VBP	verbal base form in the present tense
VP	verbal phrase
VSP	verb selectional pattern
W3C	Word Wide Web Consortium
WSJ	Wall Street Journal
XML	Extensible Markup Language

About the Authors

Sophia Ananiadou is a reader in text mining at the University of Manchester, United Kingdom. She is also a deputy director of the U.K. National Centre for Text Mining. She has worked on various projects related to sublanguage knowledge acquisition, machine translation, and lexical and terminological acquisition, funded by industry, the European Union, and U.K. research councils. Her main interests are biotext mining, natural language processing, automatic terminology management, and knowledge acquisition from biomedical texts. Dr. Ananiadou's e-mail address is Sophia.Ananiadou@manchester.ac.uk.

John McNaught is a lecturer in natural language processing in the School of Informatics, University of Manchester, United Kingdom. He is also an associate director of the U.K. National Centre for Text Mining. He has worked on numerous E.C.-funded collaborative projects involving text mining, information extraction, lexical and terminological standards, and machine translation. He was cochief editor of the *European Advisory Group on Language Engineering Standards (EAGLES)*. His interests lie in the areas of ontology-driven information extraction, sublanguage analysis, corpus linguistics, computational lexicography, and terminology. His e-mail address is John.McNaught@manchester.ac.uk.

William Black is a senior lecturer in informatics at the University of Manchester, where he researches in applied natural language processing. He is also a member of the U.K. National Centre for Text Mining. He has been working on various applications related to information management, particularly information extraction and document summarization. The work on information extraction, now expanding into text mining, has been supported by a succession of E.U. Framework IV and V projects, and the National Centre for Text Mining, while the work on summarization was supported by the British Library. Prior to his present appointment, he was in the Department of Language Engineering at

UMIST, which he headed for more than two years. His e-mail address is William.Black@manchester.ac.uk.

Christian Blaschke is the chief scientific officer for information extraction and knowledge representation at Bioalma (http://www.bioalma.com). He obtained his degree at the University Autónoma in Madrid, Spain, for bioinformatics applications in text mining. After working in biomedical text mining and information extraction in an academic environment for a few years, he joined Bioalma, and is now responsible for product development of biomedical text mining systems. His e-mail address is blaschke@bioalma.com.

Olivier Bodenreider is a staff scientist in the cognitive science branch of the Lister Hill National Center for Biomedical Communications at the National Library of Medicine. His research interests include terminology, knowledge representation, and ontology in the biomedical domain, both from a theoretical perspective and in their application to natural language understanding, reasoning, information visualization, and interoperability. His e-mail address is olivier@nlm.nih.gov.

Jeffrey Chang is an American Cancer Society Fellow in the Institute of Genome Sciences and Policy at Duke University. He is interested in the extraction and application of knowledge from primary literature towards biological problems, functional genomics, machine learning, and their applications to understanding the cell cycle and oncogenesis. His e-mail address is jeffrey_chang@stanfordalumni.org.

Udo Hahn currently holds a chair in computational linguistics at Friedrich-Schiller University in Jena, Germany, where he heads the Jena University Language and Information Engineering (JULIE) Lab. He was previously affiliated with various German universities (Freiburg, Passau, Constance). His research focuses on the content-oriented automatic analysis of texts, with main interests in the fields of biomedical text mining, information extraction, document retrieval, and text summarization. His e-mail address is hahn@coling.uni-freiburg.de.

Lynette Hirschman is the director of biomedical informatics for the Information Technology Center at the MITRE Corporation in Bedford, Massachusetts. Dr. Hirschman is responsible for leading MITRE's activities in biotechnology, including research in bioinformatics and text mining for the biomedical domain. She has been active in bringing together both the bioinformatics and natural language processing communities, and has served as an organizer of the Text Mining SIG for the International Society for Computational Biology (ISCB) since 2002. She was also a lead organizer of the Critical Assessment of Information Extraction in Biology (BioCreAtIvE), Granada, 2004. Her e-mail address is lynette@mitre.org.

Jin-Dong Kim is a research associate on the GENIA project at the University of Tokyo, Japan. He graduated in 1994 and received his Ph.D. from Korea

University in 2000. By applying techniques of natural language processing and machine learning to text mining in biology, he is interested in discovering the basic nature and principles of human language processing. His e-mail address is jindong.kim@gmail.com

Jung-jae Kim received his B.E. and M.E. in computer science from the Korea Advanced Institute of Science and Technology (KAIST), Korea, in 1988 and 2000, respectively. He is a Ph.D. candidate in the Computer Science Division at KAIST. His academic interests include biotext mining and computational linguistics. His e-mail address is jjkim@nlp.kaist.ac.kr.

Goran Nenadic is a lecturer in the School of Informatics, University of Manchester, United Kingdom, and is an associated academic member of the U.K. National Centre for Text Mining. His main research interests are in literature mining in the domain of life sciences, in particular in terminology extraction, acquisition, and classification. He also investigates the combination of textual and nontextual resources to support knowledge discovery in biomedicine. His e-mail address is G.Nenadic@manchester.ac.uk.

See-Kiong Ng is the manager of the Knowledge Discovery Department at the Institute for Infocomm Research, Singapore. He is also an adjunct associate professor at the Nanyang Technological University, Singapore. His current research interest in bioinformatics is to unravel the underlying mechanisms of biological pathways using computational methods that include data mining and text mining. His e-mail address is skng@i2r.a-star.edu.sg.

Jong C. Park received his B.E. and M.S.E. in computer engineering from Seoul National University, Korea, and his Ph.D. in computer and information science from the University of Pennsylvania, Philadelphia, Pennsylvania. He is currently an associate professor in the Computer Science Division at Korea Advanced Institute of Science and Technology. His research interests include bioinformatics, natural language processing, and computational linguistics. His e-mail address is park@cs.kaist.ac.kr.

Hinrich Schütze is a professor of theoretical computational linguistics at the Institute of Natural Language Processing of the University of Stuttgart, Germany. His main interests are text mining, statistical natural language processing, text classification, and information retrieval. His e-mail address is hinrich@stanfordalumni.org.

Jun-ichi Tsujii is a professor of computer science at the University of Tokyo, Japan, and at the School of Informatics, University of Manchester, United Kingdom. He is also the director of the U.K. National Centre for Text Mining. His main interests are grammar formalisms, adaptable natural language processing, text and knowledge management systems for biology, and multilingual natural language processing. His e-mail address is tsujii@is.s.u-tokyo.ac.jp.

Joachim Wermter is a researcher at the Jena University Language and Information Engineering (JULIE) Lab, Germany. His main research interests

include biomedical natural language processing, automatic terminology discovery, biomedical information extraction, text mining, and knowledge acquisition. His e-mail address is Joachim.Wermter@uni-jena.de.

Index

Related Artech House Titles

Achieving Software Quality Through Teamwork, Isabel Evans

Agent-Based Software Development, Michael Luck, Ronald Ashri, and Mark d'Inverno

Agile Software Development: Evaluating the Methods for Your Organization, Alan S. Koch

Building Reliable Component-Based Software Systems, Ivica Crnkovic and Magnus Larsson

Discovering Real Business Requirements for Software Project Success, Robin F. Goldsmith

Implementing and Integrating Product Data Management and Software Configuration Management, Ivica Crnkovic, Ulf Asklund, and Annita Persson

Microfluidics for Biotechnology, Jean Berthier and Pascal Silberzan

Open Systems and Standards for Software Product Development, P. A. Dargan

A Practitioner's Guide to Software Test Design, Lee Copeland

Practical Guide to Software Quality Management, *Second Edition*, John W. Horch

Practical Insight into CMMI®, Tim Kasse

The Requirements Engineering Handbook, Ralph R. Young

Software Configuration Management Handbook, *Second Edition*, Alexis Leon

Systematic Software Testing, Rick D. Craig and Stefan P. Jaskiel

Testing and Qualilty Assurance for Component-Based Software, Jerry Zeyu Gao, H.-S. Jacob Tsao, and Ye Wu

Text Mining for Biology and Biomedicine, Sophia Ananiadou and John M^{c}Naught

For further information on these and other Artech House titles, including previously considered out-of-print books now available through our In-Print-Forever® (IPF®) program, contact:

Artech House Publishers
685 Canton Street
Norwood, MA 02062
Phone: 781-769-9750
Fax: 781-769-6334
e-mail: artech@artechhouse.com

Artech House Books
46 Gillingham Street
London SW1V 1AH UK
Phone: +44 (0)20 7596 8750
Fax: +44 (0)20 7630 0166
e-mail: artech-uk@artechhouse.com

Find us on the World Wide Web at: www.artechhouse.com