8
Natural Language and Text Processing in Biomedicine

Carol Friedman and Stephen B. Johnson

After reading this chapter, you should know the answers to these questions:

- Why is natural language processing (NLP) important?
- What are the potential uses for NLP in the biomedical domain?
- What forms of knowledge are used in NLP?
- What are the principal techniques of NLP?
- What are the challenges for NLP in the clinical domain?
- What are the challenges for NLP in the biological domain?

8.1 Motivation for NLP

Natural language is the primary means of human communication. In biomedical areas, knowledge and data are disseminated in written form through articles in the scientific literature, technical and administrative reports, and patient charts used in health care (Johnson, 2000). Information is also disseminated verbally through scientific interactions in conferences, lectures, and consultations, although, in this chapter we focus on the written form. Increasingly, computers are being employed to facilitate this process of collecting, storing, and distributing biomedical information. *Textual data* are now widely available in an electronic format, through the use of transcription services, word processing, and speech recognition technology (see Chapter 5). Important examples include articles published in the biomedical literature (see Chapter 19) and reports describing particular processes of patient care (e.g., radiology reports and discharge summaries; see Chapter 12).

While the ability to access and review *narrative data* is highly beneficial to researchers, clinicians, and administrators, the information is not in a form amenable to further computer processing, for example, storage in a structured database to enable subsequent retrievals. Narrative text is difficult to access reliably because the variety of expression is vast; many different words can be used to denote a single concept and an enormous variety of grammatical structures can be used to convey equivalent information. At present, the most significant impact of the computer in medicine is seen in processing **structured data**, information represented in a regular, predictable form. This information is often numeric in nature (e.g., measurements recorded in a scientific study) or made up of discrete data elements (e.g., elements selected from a predefined list of biomedical terms, such as the names of diseases or genes). The techniques of NLP provide a means

to bridge the gap between textual and structured data, allowing humans to interact using familiar natural language while enabling computer applications to process data effectively.

8.2 Applications of NLP

NLP has a wide range of potential applications in the biomedical domain. NLP enables a new level of functionality for health care and research-oriented applications that would not be otherwise possible. NLP methods can help manage large volumes of text (e.g., patient reports or journal articles) by extracting relevant information in a timely manner. Some text-processing tasks are currently performed by humans, for example, human coders identify diagnoses and procedures in patient documents for billing purposes, and database curators extract genomic information on organisms from the literature. However, it is generally not feasible to perform these tasks manually because they are too costly and too time consuming. For example, an automated system could process enormous numbers of patient reports to detect medical errors, whereas it would not be possible for experts to check such large volumes. Because automated systems are based on rules determined by experts, it is possible to incorporate the best and most current knowledge into the rules. Such systems are generally more consistent and objective than humans. Another significant advantage of NLP is the ability to standardize information occurring in documents from diverse applications and institutions, representing the information uniformly with common output structure and vocabulary.

The following are important applications of NLP technology in biomedicine:

- **Information extraction** locates and structures important information in text, usually without performing a complete analysis. This is the most common application in biomedicine, and is the primary focus of this chapter. The technique may be limited to the identification of isolated terms in text (e.g., medications or proteins), which can then be mapped to canonical or standardized forms. A slightly more complex application may search for recognizable patterns in text, such as names of people or places, dates, and numerical expressions. More sophisticated techniques identify and represent the relations among the terms within a sentence. Such advanced methods are necessary for reliable retrieval of information in patient documents and the biomedical literature because the correct interpretation of a biomedical term typically depends on its relation with other terms. For example, `fever` has different interpretations in `no fever`, `high fever`, `fever lasted 2 days`, and `check for fever`.[1] In the biomolecular domain, an important use of this technology involves extracting interactions from individual journal articles, and then subsequently combining them in order to automatically generate pathways. Figure 8.1 is an example of a pathway in the form of a graph that was created by extracting some interactions from one journal article in *Cell* (Maroto et al., 1997).

[1]The natural language processing literature typically connotes narrative text in *italics*; however, in this chapter we will depict text from narrative reports using Courier font to distinguish it from important informatics terms that are italicized throughout this book.

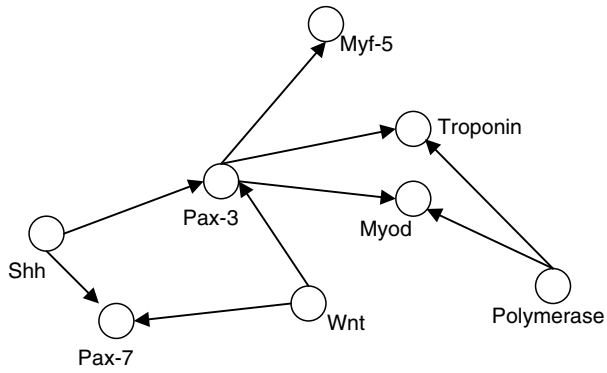

Figure 8.1. A graph showing interactions that were extracted from an article. A vertex represents a gene or protein, and an edge represents the interaction. The arrow represents the direction of the interaction so that the agent is represented by the outgoing end of the arrow and the target by the incoming end.

- **Information retrieval** helps users to access documents in very large collections, such as the scientific literature. This is a crucial application in biomedicine, due to the explosion of information available in electronic form. The essential goal of information retrieval is to match a user's query against the collection and return the most similar documents. Because matching is approximate, the process is usually iterative, requiring several rounds of refinement. The most basic form of *indexing* isolates simple words and terms. More advanced approaches use methods similar to those employed in information extraction, identifying complex noun phrases and determining their relationships in order to improve the accuracy of retrieval. For example, it is important to distinguish between a journal article that discusses the use of a drug to treat a medical condition from an article that discusses a medical condition being a side effect of a drug.
- **Text generation** formulates natural language sentences from a given source of information, usually structured data. These techniques can be used to generate text from a structured database, such as summarizing trends and patterns in laboratory data. Another important application is the generation of small summaries from large texts. This may involve summarization of a single document (e.g., a single clinical report such as a discharge summary), or of multiple documents (e.g., multiple journal articles).
- **User interfaces** (see Chapter 12) enable humans to communicate more effectively with computer systems. Tools that facilitate data entry are an important application in biomedicine. Data can be captured by keyboard (e.g., using templates or macros) or by speech recognition technology that enables users to enter words directly into computer systems by speaking. Additional examples (somewhat less common) include issuing commands or querying a database using natural language.
- **Machine translation** converts text in one language (e.g., English) into another (e.g., Spanish). These applications are important in multilingual environments in which human translation is too expensive or time consuming. Examples include translating medication instructions to assist patients, translating consent forms to enroll diverse subjects in a study, and translating journal articles to reach an international audience.

8.3 Knowledge Used in NLP

While current linguistic theories differ in certain details, there is broad consensus that linguistic knowledge consists of multiple levels: *morphology* (parts of words), *lexicography* (words and terms), *syntax* (phrases and sentences), *semantics* (words, phrases and sentences), and *pragmatics* (paragraphs and documents). Human language processing may appear deceptively simple, because we are not conscious of the effort involved in learning and using language. However, a long process of acculturation is necessary to attain proficiency in speaking, reading, and writing, with further intensive study to master the language of biological science or medicine. The sections below briefly describe the nature of the knowledge in each of these levels.

Morphology concerns the combination of **morphemes** (roots, prefixes, suffixes) to produce words. **Free morphemes** can occur as separate words, while **bound morphemes** cannot, e.g., `de-` in `detoxify`, `-tion` in `creation`, `-s` in `dogs`. **Inflectional morphemes** express grammatically required features or indicate relations between different words in the sentence, but do not change the basic syntactic category, thus `big`, `bigg-er`, `bigg-est` are all adjectives. **Derivational morphemes** change the part of speech or the basic meaning of a word, thus `-ment` added to a verb forms a noun (`judg-ment`); `re-activate` means activate again. Compared with other languages, English does not exhibit complex morphology, and therefore many NLP systems for general English do not incorporate morphological knowledge. However, biomedical language has a very rich morphological structure especially for chemicals (e.g., `Hydr-oxy-nitro-di-hydro-thym-ine`) and procedures (`hepatico-cholangio-jejuno-stom-y`). Recognizing morphemes enables an NLP system to handle words much more flexibly, especially in dealing with new words. However, determining the correct separation can be difficult. In the previous chemical example, the first split must be made after `hydr-` (because the `-o-` is part of `-oxy`) while the fifth split occurs after `hydro-`. In the procedure example, the system must distinguish `stom` (mouth) from `tom` (cut) in `-stom`.

Lexicography concerns the categorization of **lexemes**, the words and atomic terms of the language. Each lexeme belongs to one or more *parts of speech* in the language, such as noun (e.g., `chest`), adjective (e.g., `mild`), or tensed verb (e.g., `improves`), which are the elementary components of the English grammar. Lexemes may also have *subcategories*, depending on the basic part of speech, which are usually expressed by inflectional morphemes. For example, nouns have number (e.g., plural or singular as in `legs`, `leg`), person (e.g., first, second, third as in `I`, `you`, `he`, respectively), and case (e.g., subjective, objective, possessive as in `I`, `me`, `my`, respectively). Lexemes can consist of more than one word as in foreign phrases (`ad hoc`), prepositions (`along with`), and idioms (`follow up`, `on and off`). Biomedical lexicons tend to contain many multiword lexemes, e.g., lexemes in the clinical domain include `congestive heart failure` and `diabetes mellitus`, and in the biomolecular domain include the gene named `ALL1-fused gene from chromosome 1q`.

Syntax concerns the *structure* of the phrases and sentences. Lexemes combine (according to their parts of speech) in well-defined ways to form phrases such as noun phrases (e.g., `severe chest pain`), adjectival phrases (e.g., `painful to touch`),

or verb phrases (e.g., `has increased`). Each phrase generally consists of a main part of speech and modifiers, e.g., nouns are frequently modified by adjectives while verbs are frequently modified by adverbs. The phrases then combine in well-defined ways to form sentences (`he complained of severe chest pain`). General English imposes many restrictions on the formation of sentences, e.g., every sentence requires a subject, and count nouns (like `cough`) require an article (e.g., `a` or `the`). Clinical language is often *telegraphic*, relaxing many of these restrictions to achieve a highly compact form. For example, clinical language allows all of the following as sentences: `the cough worsened; cough worsened; cough`. Because the community widely uses and accepts these alternate forms, they are not considered ungrammatical but constitute a **sublanguage** (Kittredge and Lehrberger 1982; Grishman and Kittredge, 1986; Friedman, 2002). There is a wide variety of sublanguages in the biomedical domain, each exhibiting specialized content and linguistic forms.

Semantics concerns the *meaning* or *interpretation* of words, phrases, and sentences. Each word has one or more meanings or *word senses* (e.g., `capsule,` as in `renal capsule` or as in `vitamin B12 capsule`), and the meanings of the words combine to form a meaningful sentence, as in `there was thickening in the renal capsule`). Representing the semantics of general language is an extremely difficult problem, and an area of active research. Biomedical sublanguages are easier to interpret than general languages because they exhibit highly restrictive semantic patterns that can be represented more easily (Harris et al., 1989, 1991; Sager et al., 1987). Sublanguages tend to have a relatively small number of **semantic types** (e.g., medication, gene, disease, body part, or organism) and a small number of **semantic patterns**: medication-treats-disease, gene-interacts with-gene.

Pragmatics concerns how sentences combine to form *discourse* (paragraphs, documents, dialogues, etc.), and studies how this context affects the interpretation of the meaning of individual sentences. For example, in a mammography report, `mass` generally denotes `breast mass`, in a radiological report of the chest it denotes `mass in lung` whereas in a religious journal it is likely to denote a ceremony. Similarly, in a health care setting, `he drinks heavily` is assumed to be referring to alcohol and not water. Another pragmatic consideration is the interpretation of pronouns and other referential expressions (`there, tomorrow`). For example, in `An infiltrate was noted in right upper lobe; it was patchy, it` refers to `infiltrate` and not `lobe.` Other linguistic devices are used to link sentences together, for example to convey a complex temporal sequence of events.

8.4 NLP Techniques

NLP involves three major tasks: (1) representing the various kinds of linguistic knowledge discussed in Section 8.3, (2) using the knowledge to carry out the applications described in Section 8.2, and (3) acquiring the necessary knowledge in computable form. The field of computer science provides a number of formalisms that can be used to represent the knowledge (task 1). These include symbolic or logical formalisms (e.g., **finite state machines** and **context-free grammars**) and statistical formalisms (e.g., **Markov**

models and **probabilistic context-free grammars**). The use of these representations to analyze language is generally called **text parsing**, while their use to create language is called **text generation**. Traditionally, the acquisition of linguistic knowledge (task 3) has been performed by trained linguists, who manually construct linguistically based rule systems, or grammars. This process is extremely time intensive. Increasingly, there is interest in using methods of machine learning to acquire the knowledge with less effort from linguists. However, machine learning generally requires the creation of training data, which also requires extensive manual annotation.

Most NLP systems are designed with separate modules that handle different functions. The modules typically coincide with the linguistic levels described in Section 8.3. In general, the output from lower levels serves as input to higher levels. For example, the result of lexical analysis is input to syntactic analysis, which in turn is input to semantic analysis. Each system packages these processing steps somewhat differently. At each stage of processing, the module for that stage regularizes the data in some aspect while preserving the informational content as much as possible.

8.4.1 Morphology

The first step in processing generally consists of reading the electronic form of a text (usually it is initially one large string), and separating it into individual units called tokens (the process is called **tokenization**), which include morphemes, words (really morpheme sequences), numbers, symbols (e.g., mathematical operators), and punctuation. The notion of what constitutes a word is far from trivial. The primary indication of a word in general English is the occurrence of white space before and after a word; however, there are many exceptions. A word may be followed by certain punctuation marks without an intervening space, such as by a period, comma, semicolon, or question mark, or may have a "-" in the middle. In biomedicine, periods and other punctuation marks can be part of words (e.g., `q.i.d.` meaning `four times a day` in the clinical domain or `M03F4.2A`, a gene name that includes a period), and are used inconsistently, thereby complicating the tokenization process. Chemical and biological names often include parentheses, commas, and hyphens, for example `(w)adh-2`.

Symbolic approaches to tokenization are based on pattern matching. Patterns are conveniently represented by the formalism known as a **regular expression** or equivalently, a **finite state automata** (Jurafsky and Martin, 2000, pp. 21–52). For example, the following regular expression will identify the tokens contained in the sentence `patient's wbc dropped to 12`:

```
[a-z]+('s)?|[0-9]+|[.]
```

The vertical bar (`|`) separates alternative expressions, which in this case specify three different kinds of tokens (alphabetic, numeric, and punctuation). Expressions in square brackets represent a range or choice of characters. The expression `[a-z]` indicates a lower case letter, while `[0-9]` indicates a digit. The plus sign denotes one or more occurrences of an expression. The question mark indicates an optional expression (apostrophe `-s`). Finally `[.]` indicates a period. This regular expression is very limited,

because it does not deal with capital letters (e.g., `Patient`), numbers with a decimal point (`3.4`), or abbreviations terminated by a period (`mg.`).

More complex regular expressions can handle many of the morphological phenomena described above. However, situations that are locally ambiguous are more challenging. For example, in the sentence "`5 mg. given.`" the period character is used in two different ways: (1) to signal an abbreviation, and (2) to terminate the sentence. There is also the significant issue that we may not have anticipated all the possible patterns. Probabilistic methods such as Markov models provide a more robust solution. Markov models can be represented as a table (**transition matrix**). For this simple example, the table might appear as shown in Table 8.1. The rows represent the current symbol in the sentence, and the columns represent the words that can follow. Each cell indicates the probability that a given word can follow another.

In mathematical notation this can be written as P(following|current). The probability of a given sequence of tokens can be approximated by multiplying the probabilities of the individual transitions. Thus,

$$P(5\ \text{mg. given}) = P(\text{mg.}|5)P(\text{given}|\text{mg.})P(\text{given}|.) = 0.9 \times 0.9 \times 0.7 = 0.567$$
$$P(5\ \text{mg. given}) = P(\text{mg}|5)P(.|\text{mg})P(\text{given}|\text{mg})P(.|\text{given}) = 0.8 \times 0.4 \times 0.8 \times 0.7 = 0.1792$$

To find the best tokenization of a given sequence of characters, it is necessary to determine all possible ways of dividing the tokens and then to select the one that yields the maximum probability. For long sequences, a more efficient method known as the **Viterbi algorithm** is used, which considers only a small proportion of the possible sequences (Jurafsky and Martin 2000, pp. 177–180). In practice, the transition matrix would be very large to accommodate the wide range of possible tokens found in biomedical text. The transition probabilities are typically estimated from training sets in which linguists have verified the correct tokenization. However, for accuracy, it is important that the training set be typical for the intended text and that the training set is sufficiently large.

8.4.2 Lexicography

Once text is tokenized, an NLP system needs to perform **lexical look up** to identify the words or multiword terms known to the system, and determine their categories and canonical forms. Many systems carry out tokenization on complete words and perform lexical look up immediately afterwards. This requires that the lexicon contains all the possible combinations of morphemes. Each lexical entry assigns a word to one or more parts of speech, and a canonical form. For example, `abdominal` is an adjective where the canonical form is `abdomen`, and `activation` is a noun that is the nominal form

Table 8.1. Transition probabilities for morphemes.

	5	mg	mg.	given	.
5	0.1	0.8	0.9	0.4	0.6
mg	0.3	0.1	0.1	0.9	0.4
mg.	0.3	0.1	0.1	0.9	0.2
given	0.7	0.6	0.6	0.2	0.7
.	0.6	0.4	0.4	0.8	0.1

of the verb `activate`. A few systems perform morphological analysis during tokenization. In that case, the lexicon only needs entries for roots, prefixes, and suffixes, with additional entries for irregular forms. For example, the lexicon would contain entries for the roots `abdomen` (with variant `abdomin-`) the adjective suffix `-al`, and `activat-`, verb suffix `-e`, and noun suffix `-ion`.

Lexical look up is not straightforward because a word may be associated with more than one part of speech. For example, `stay` may be a noun (as in `her hospital stay`) or a verb (as in `refused to stay`). Without resolution, these ambiguities could cause inaccuracies in parsing and interpretation, and must be addressed in subsequent stages of processing, using syntactic and semantic information. Alternatively, various methods for **part of speech tagging** may be used to resolve ambiguities by considering the surrounding words. For example, when `stay` follows `the` or `her` it is usually tagged as a noun, but after `to` it is usually tagged as a verb. A symbolic approach to this problem is the use of transformation rules that change the part of speech tag assigned to a word based on previous or following tags. The meaning of some part of speech tags are provided in Table 8.2.

The following are the rules that might be applied to clinical text.

Change NN to VB if the previous tag is TO
Change NN to JJ if the following tag is NN
Change IN to TO if the following tag is VB

Examples of applying these rules are shown in Table 8.3.

Table 8.2. Meanings of part of speech tags.

Tag	Meaning
NN	Singular noun
NNS	Plural noun
NNP	Proper noun singular
IN	Preposition
VB	Infinitive verb
VBD	Past-tense verb
VBG	Progressive verb form
VBN	Past participle
VBZ	Present-tense verb
JJ	Adjective
DT	Article
PP$	Possessive pronoun

Table 8.3. Application of transformation rules to part of speech tags.

Before rule application	After rule application
total/NN hip/NN replacement/NN	total/JJ hip/NN replacement/NN
a/DT total/NN of/IN four/NN units/NNS	(no change)
refused/VBD to/TO stay/NN	refused/VBD to/TO stay/VB
her/PP$ hospital/NN stay/NN	(no change)
unable/JJ to/IN assess/VB	unable/JJ to/TO assess/VB
allergy/NN to/IN penicillin/NN	(no change)

Rules for part of speech tagging can be created by hand or constructed automatically using **transformation-based learning**, based on a sample corpus where the correct parts of speech have been manually annotated (Jurafsky and Martin 2000, pp. 307–312). Statistical approaches to part of speech tagging are based on Markov models (as described above for morphology). The transition matrix specifies the probability of one part of speech following another (see Table 8.4):

The following sentence shows the correct assignment of part of speech tags: `Rheumatology/NN consult/NN continued/VBD to/TO follow/VB patient/NN.`

This assignment is challenging for a computer, because `consult` can be tagged VB (`Orthopedics asked to consult`), `continued` can be tagged VBN (`penicillin was continued`), and `to` can be tagged IN. However, probabilities can be calculated for these sequences using the matrix in Table 8.4 (these were estimated from a large corpus of clinical text). By multiplying the transitions together, a probability for each sequence can be obtained (as described above for morphology), and is shown in Table 8.5. Note that the correct assignment has the highest probability.

8.4.3 Syntax

Many NLP systems perform some type of **syntactic analysis**. A **grammar** specifies how the words combine into well-formed structures, and consists of rules where categories combine with other categories or structures to produce a well-formed structure with underlying relations. Generally, words combine to form phrases consisting of a head word and modifiers, and phrases form sentences or clauses. For example, in English there are noun phrases (NP) that contain a noun and optionally left and right modifiers,

Table 8.4. Transition probabilities for part of speech tags.

	NN	VB	VBD	VBN	TO	IN
NN	0.34	0.00	0.22	0.02	0.01	0.40
VB	0.28	0.01	0.02	0.27	0.04	0.39
VBD	0.12	0.01	0.01	0.62	0.05	0.19
VBN	0.21	0.00	0.00	0.03	0.11	0.65
TO	0.02	0.98	0.00	0.00	0.00	0.00
IN	0.85	0.00	0.02	0.05	0.00	0.08

Table 8.5. Probabilities of alternative part of speech tag sequences.

Part of speech tag sequence	Probability
NN NN VBD TO VB NN	0.001149434
NN NN VBN TO VB NN	0.000187779
NN VB VBN TO VB NN	0.000014194
NN NN VBD IN VB NN	0.000005510
NN NN VBN IN VB NN	0.000001619
NN VB VBD TO VB NN	0.000000453
NN VB VBN IN VB NN	0.000000122
NN VB VBD IN VB NN	0.000000002

such as definite articles, adjectives, or prepositional phrases (i.e., `the patient, lower extremities, pain in lower extremities, chest pain`), and verb phrases (VP), such as `had pain, will be discharged,` and `denies smoking.`

Simple phrases can be represented using **regular expressions** (as shown above for tokenization). In this case, syntactic categories are used to match the text instead of characters. A regular expression (using the tags defined in Table 8.2) for a simple noun phrase (i.e., a noun phrase that has no modifiers on the right side) is:

```
DT? JJ* NN* (NN|NNS)
```

This structure specifies a simple noun phrase as consisting of an optional determiner (i.e., `a, the, some, no`), followed by zero or more adjectives, followed by zero or more singular nouns, and terminated by a singular or plural noun. For example, the above regular expression would match the noun phrase `no/AT usual/JJ congestive/JJ heart/NN failure/NN symptoms/NNS` but would not match `heart/NN the/AT unusual/JJ`, because in the above regular expression `the` cannot occur in the middle of a noun phrase.

Some systems perform partial parsing using regular expressions. These systems determine local phrases, such as simple noun phrases (i.e., noun phrases without right adjuncts) and simple adjectival phrases, but do not determine relations among the phrases. These systems tend to be robust because it is easier to recognize isolated phrases than it is to recognize complete sentences, but typically they lose some information. For example, in `amputation below knee`, the two noun phrases `amputation` and `knee` would be extracted, but the relation `below` might not be.

More complex structures can be represented by **context-free grammars** (Jurafsky and Martin 2000, pp. 325–344). A complete noun phrase cannot be handled using a regular expression because it contains nested structures, such as nested prepositional phrases or nested relative clauses. A very simple grammar of English is shown in Figure 8.2.

Context-free rules use part of speech tags (see Table 8.2) and the operators found in regular expressions, for optionality (`?`), repetition (`*`), and alternative (`|`). The difference is that each rule has a nonterminal symbol on the left side (S, NP, VP, PP), which consists of a rule that specifies a sequence of grammar symbols (nonterminal, and terminal) on the right side. Thus, the S (sentence) rule contains an NP followed by a VP. Additionally, other rules may refer to these symbols or to the atomic parts of speech. Thus, the NP rule contains PP, which in turn contains NP.

Applying the grammar rules to a given sentence is called parsing, and if the grammar rules can be satisfied, the grammar yields a nested structure that can be represented

```
S  →NP  VP .
NP →DT? JJ* (NN|NNS) CONJN* PP* | NP and NP
VP →(VBZ |VBP) NP? PP*
PP →IN NP
CONJN →and  (NN|NNS)
```

Figure 8.2. A simple syntactic context-free grammar of English. A sentence is represented by the rule S, a noun phrase by the rule NP, a verb phrase by VP, and a prepositional phrase by PP. Terminal symbols in the grammar, which correspond to syntactic parts of speech, are underlined in the figure.

graphically as a **parse tree**. For example, the sentence `the patient had pain in lower extremities` would be assigned the parse tree shown in Figure 8.3.

Alternatively, brackets can be used to represent the nesting of phrases instead of a parse tree. Subscripts on the brackets specify the type of phrase or tag:

```
[ S [ NP [ DT the] [ NN patient]] [ VP [ VBD had]
  [ NP [ NN pain] [ PP [ IN in] [ NP [ JJ lower] [ NNS extremities]]]]]]
```

The following shows an example of a parse in the biomolecular domain for the sentence `Activation of Pax-3 blocks Myod phosphorylation`:

```
[ S [ NP [ NN Activation] [ PP [ IN of] [ NP [ NN Pax-3]]]]
  [ VP [ VBZ blocks] [ NP [ NNP Myod] [ NN phosphorylation]]]]
```

Grammar rules generally give rise to many possible structures for a parse tree (structural ambiguity). If a word has more than one part of speech, the choice of part of speech for the word can result in different structures for the sentence. For example, when `swallowing` occurs before a noun, it can be an adjective (JJ) that modifies the noun, or a verb (VBG) that takes the noun as an object:

```
Swallowing/JJ evaluation/NN showed/VBD no/DT dysphagia/NN
Swallowing/VBG food/NN showed/VBD no/DT dysphagia/NN
```

Additionally, the sequence of alternative choices of rules in the grammar can yield different groupings of phrases. For example, sentence 1a below corresponds to a parse based on the grammar rules shown in Figure 8.2, where the VP rule contains a PP (e.g., `denied in the ER`) and the NP rule contains only a noun (e.g., `pain`). Sentence 1b

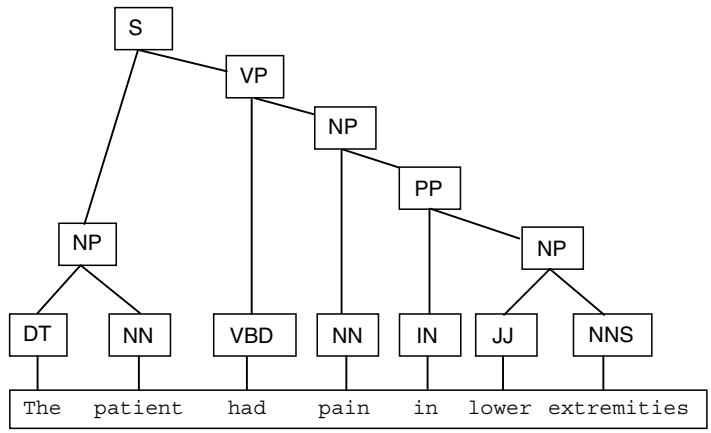

Figure 8.3. A parse tree for the sentence the patient had pain in lower extremities according to the context-free grammar shown in Figure 8.2. Notice that the terminal nodes in the tree correspond to the syntactic categories of the words in the sentence.

corresponds to the same atomic sequence of syntactic categories but the parse is different because the VP rule contains only a verb (e.g., `denied`) and the NP contains a noun followed by a PP (e.g., `pain in the abdomen`). Prepositions and conjunctions are also a frequent cause of ambiguity. In 2a, the NP consists of a conjunction of the head nouns so that the left adjunct (e.g., `pulmonary`) is distributed across both nouns (i.e., this is equivalent to an interpretation `pulmonary edema` and `pulmonary effusion`), whereas in 2b the left adjunct `pulmonary` is attached only to `edema` and is not related to `effusion`. In 3a, the NP in the prepositional phrase PP contains a conjunction (i.e., this is equivalent to `pain in hands` and `pain in feet`) whereas in 3b two NPs are also conjoined but the first NP consists of `pain in hands` and the second consists of `fever`.

1a. `Denied [ pain] [ in the ER]`
1b. `Denied [ pain [ in the abdomen]]`

2a. `Pulmonary [ edema and effusion]`
2b. `[ Pulmonary edema] and anemia`

3a. `Pain in [ hands and feet]`
3b. `[ Pain in hands] and fever`

More complex forms of ambiguity do not exhibit differences in parts of speech or in grouping, but require determining deeper syntactic relationships. For example, when a verb ending in `-ing` is followed by `of`, the following noun can be either the subject or object of the verb.

```
Feeling of lightheadedness improved.
Feeling of patient improved.
```

Statistical approaches provide one method of addressing ambiguity. The essential idea is to exploit the fact that some choices in the grammar are more likely than others. This can be represented using a **probabilistic context-free grammar**, which associates a probability with each choice in a rule (Jurafsky and Martin 2000, pp. 448–458). The grammar above can be annotated with probabilities for each choice by placing a numerical superscript after each symbol. The number indicates the probability of including the given category in the parse tree. For example, the probability of having a determiner (DT) is 0.9, while not having one has a probability of 0.1. The probability of a present tense verb (VBZ) is 0.4, while a past tense verb (VBD) is 0.6.

```
S → NP VP.
NP → DT?·9 JJ*·8 (NN|·6 NNS) PP*·8
VP → (VBZ|·4 VBD) NP?·9 PP*·7
PP → IN NP
```

The probability of a given parse tree is the product of the probabilities of each grammar rule used to make it. For example, there are two ways to parse `X-ray shows`

`patches in lung` using this grammar (shown below). The first interpretation in which `shows` is modified by `lung` has probability 3.48×10^{-8}, while the second interpretation in which `patches` is modified by `lung` has probability 5.97×10^{-8}.

$$[_S\ [_{NP}\ \text{NN}\ 0.1 \times 0.2 \times 0.6 \times 0.2]\ [_{VP}\ \text{VBZ}\ [_{NP}\ \text{NN}\ 0.1 \times 0.2 \times 0.6 \times 0.2]\ [_{PP}\ \text{IN}\ [_{NP}\ \text{NN}\ 0.1 \times 0.2 \times 0.6 \times 0.2]]\ 0.4 \times 0.9 \times 0.7]]$$

$$[_S\ [_{NP}\ \text{NN}\ 0.1 \times 0.2 \times 0.6 \times 0.2]\ [_{VP}\ \text{VBZ}\ [_{NP}\ [_{PP}\ \text{IN}\ [_{NP}\ \text{NN}\ 0.1 \times 0.2 \times 0.6 \times 0.2]\ \text{NN}\ 0.1 \times 0.2 \times 0.6 \times 0.8]\ 0.4 \times 0.9 \times 0.3]]$$

8.4.4 Semantics

Semantic analysis involves steps analogous to those described above for syntax. First, semantic interpretations must be assigned to individual words. Then, these are combined into larger semantic structures (Jurafsky and Martin 2000, pp. 510–512). Semantic information about words is generally maintained in the lexicon. A **semantic type** is usually a broad class that includes many instances while a **semantic sense** distinguishes individual word meanings (Jurafsky and Martin 2000, pp. 592–601). For example, `aspirin`, `ibuprofen` and `Motrin` all have the same semantic type (medication), `ibuprofen` and `Motrin` have the same semantic sense (they are synonymous), which is distinct from the sense of `aspirin` (a different drug).

A lexicon may be created manually by a linguist, or be derived from external knowledge sources, such as the **Unified Medical Language System** (UMLS) (Lindberg et al., 1993; see Chapter 7) or GenBank (Benson et al., 2003). While external sources can save a substantial effect, the types and senses provided may not be appropriate for the text being analyzed. Narrow categories may be too restrictive, and broad categories may introduce ambiguities. Morphological knowledge can be helpful in determining semantic types in the absence of lexical information. For example, in the clinical domain, suffixes like `-itis` and `-osis` indicate diseases, while `-otomy` and `ectomy` indicate procedures. However, such techniques cannot determine the specific sense of a word.

As with parts of speech, many words have more than one semantic type, and the NLP system must determine which of these is intended in the given context. For example, `growth` can be either an abnormal physiologic process (e.g., for a tumor) or a normal one (e.g., for a child). The word `left` can indicate laterality (`pain in left leg`) or an action (`patient left hospital`). This problem is much harder than syntactic disambiguation because there is no well-established notion of word sense, different lexicons recognize different distinctions, and the space of word senses is substantially larger than that of syntactic categories. Words may be ambiguous within a particular domain, across domains, or with a general English word. Abbreviations are notoriously ambiguous. For example, the abbreviation `MS` may denote `multiple sclerosis` or `mitral stenosis` or it may denote the general English usage (i.e., as in `Ms White`). The ambiguity problem is particularly troublesome in the biomolecular domain because gene symbols in many model organism databases consist of three letters, and are ambiguous with other English words, and also with different gene symbols of different model organisms. For example, `nervous` and `to` are English words that are also the names of genes. When writing about a specific organism, authors use

alias names, which may correspond to different genes. For example, in articles associated with the mouse, according to the Mouse Genome Database (MGD) (Blake et al., 2003), authors may use the term `fbp1` to denote three different genes.

Semantic disambiguation of **lexemes** can be performed using the same methods described above for syntax. Rules can assign semantic types using contextual knowledge of other nearby words and their types. For example, `discharge from hospital` and `discharge from eye` can be disambiguated depending on whether the noun following `discharge` is an institution or a body location. As illustrated in Table 8.6, a rule may change the hospitalization action sense (e.g., HACT) that denotes `discharge` to the body substance sense `discharge` (e.g., BSUB) if the following semantic category is a body part (e.g., PART).

Statistical approaches, such as Markov models, can be used to determine the most likely assignment of semantic types (Jurafsky and Martin 2000, pp. 636–645). As with methods for morphology and syntax, large amounts of training data are required to provide sufficient instances of the different senses for each ambiguous word. This is extremely labor intensive because a linguist must manually annotate the corpus, although in certain cases automated annotation is possible.

Larger semantic structures consisting of **semantic relations** can be identified using regular expressions, which specify patterns of semantic types. The expressions may be semantic and look only at the semantic categories of the words in the sentence. This method may be applied in the biomolecular domain to identify interactions between genes or proteins. For example, the regular expression

```
[ GENE|PROT]  MFUN [ GENE|PROT]
```

will match sentences consisting of very simple gene or protein interactions (e.g., `Pax-3/GENE activated/MFUN Myod/GENE`). In this case, the elements of the pattern consist of semantic classes: GENE (gene), molecular function (MFUN), and PROT (protein). This pattern is very restrictive because any deviation from the pattern will result in a failed match. Regular expressions that skip over parts of the sentence when trying to find a match are much more robust, and can be used to detect relevant patterns for a broader variety of text, thus incurring some loss of specificity and precision while achieving increased sensitivity. For example, the regular expression

```
[ GENE|PROT]  .* MFUN .* [ GENE|PROT]
```

can be satisfied by skipping over intermediate tags in the text. The dot (**.**) matches any tag, and the asterisk (*****) allows for an arbitrary number of occurrences. For example, using the above expression, the interaction, `Pax-3 activated Myod` would

Table 8.6. Application of transformation rules to semantic tags. HACT denotes an action (e.g., *admission, discharge*), PART denotes a body part (e.g., *eye, abdomen*), and BSUB denotes a body substance (e.g., *sputum*).

Before rule application	After rule application
Discharge/*HACT from hospital*/HORG	(no change)
Discharge/*HACT from eye*/*PART*	*Discharge*/*BSUB from eye*/*PART*

be obtained for the sentence `Pax-3/GENE, only when activated/MFUN by Myod/GENE, inhibited/MFUN phosphorylation/MFUN`. In this example, the match does not capture the information correctly because the relation `only when` was skipped. The correct interpretation of the individual interactions in this sentence should be `Myod activated Pax-3`, and `Pax-3 inhibited phosphorylation`. Note that the simple regular expression shown above does not provide for the latter pattern (i.e., GENE-MFUN-MFUN), for the connective relation, or for the passive structure.

An alternate method of processing sentences with regular expressions, which is currently the most widely employed in general English because it is very robust, uses **cascading finite state automata (FSA)** (Hobbs et al., 1996). In this technique, a series of different FSAs are employed so that each performs a special tagging function. The tagged output of one FSA becomes the input to a subsequent FSA. For example, one FSA may perform tokenization and lexical look up, another may perform partial parsing to identify syntactic phrases, such as noun phrases and verb phrases, and the next may determine semantic relations. In that case, the patterns for the semantic relations will be based on a combination of syntactic phrases and their corresponding semantic classes, as shown below. The pattern for biomolecular interactions might then be represented using a combination of tags:

$$\mathrm{NP}_{[\,\mathrm{GENE}\,|\,\mathrm{PROT}\,]}\ .^*\ \mathrm{VP}_{\mathrm{MFUN}}\ .^*\ \mathrm{NP}_{[\,\mathrm{GENE}\,|\,\mathrm{PROT}\,]}$$

The advantage of cascading FSA systems is that they are relatively easy to adapt to different information extraction tasks because the FSAs that are domain independent (tokenizing and phrasal FSAs) remain the same while the domain-specific components (semantic patterns) change with the domain and or the extraction task. These types of systems have been used to extract highly specific information, such as detection of terrorist attacks, identification of joint mergers, and changes in corporation management (Sundheim 1991, 1992, 1994, 1996; Chinchor 1998). However, they may not be accurate enough for clinical applications.

More complex semantic structures can be recognized using a **semantic grammar** that is a context-free grammar based on semantic categories. As shown in Figure 8.4, a simple grammar for clinical text might define a clinical sentence as a Finding, which consists of optional degree information and optional change information followed by a symptom.

```
S →          Finding .
Finding → DegreePhrase?  ChangePhrase? SYMP
ChangePhrase → NEG? CHNG
DegreePhrase → DEGR | NEG
```

Figure 8.4. A simple semantic context-free grammar for the English clinical domain. A sentence S consists of a FINDING, which consists of an optional DEGREEPHRASE, an optional CHANGEPHRASE and a Symptom. The DEGREEPHRASE consists of a degree type word or a negation type word; the CHANGEPHRASE consists of an optional negation type word followed by a change type word. The terminal symbols in the grammar correspond to semantic parts of speech and are underlined.

This is particularly effective for domains where the text is very compact, and where typical sentences consist primarily of noun phrases because the subject (i.e., `patient`) and verb have been omitted. For example, `increased/CHNG tenderness/SYMP` is a typical sentence in the clinical domain where both the subject and verb are omitted. For the simple grammar illustrated in Figure 8.4, the parsed sentence would be a **FINDING** that consists of a **CHANGEPHRASE** (e.g., `increased`) followed by a **SYMPTOM** (e.g., `tenderness`). Note that ambiguity is possible in this grammar because a sentence such as `No/NEG increased/CHNG tenderness/SYMP` could be parsed in two ways. In the incorrect parse shown in Figure 8.5, the **DEGREEPHRASE** (e.g., `no`) and the **CHANGEPHRASE** (e.g., `increased`) both modify `tenderness`, whereas in the correct parse (see Figure 8.6) only the **CHANGEPHRASE** (e.g., `no increased`) modifies `tenderness`, and within the **CHANGEPHRASE**, `no` modifies **CHANGE** (e.g., `increased`); in this case only the change information is negated but not the symptom.

NLP systems can handle more complex language structures by integrating syntactic and semantic structures into the grammar (Friedman et al., 1994). In that case, the grammar would be similar to that shown in Figure 8.4, but the rules would also include syntactic structures. Additionally, the grammar rule may also specify the representational output form, which represents the underlying interpretation of the relations. For example, in Figure 8.4, the rule for FINDING would specify an output form denoting that SYMP is the primary finding and the other elements are the modifiers.

More comprehensive syntactic structures can be recognized using a broad-coverage context-free grammar of English, which is subsequently combined with a semantic

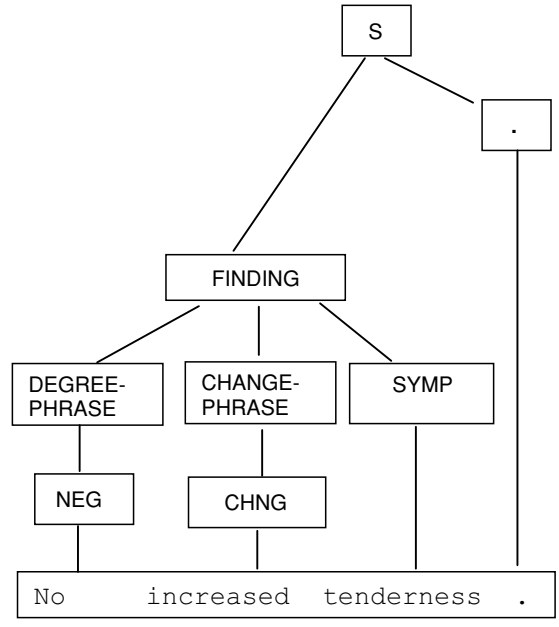

Figure 8.5. A parse tree for the sentence no increased tenderness according to the grammar shown in Figure 8.4. In this interpretation, which is incorrect, no and increased each modify tenderness.

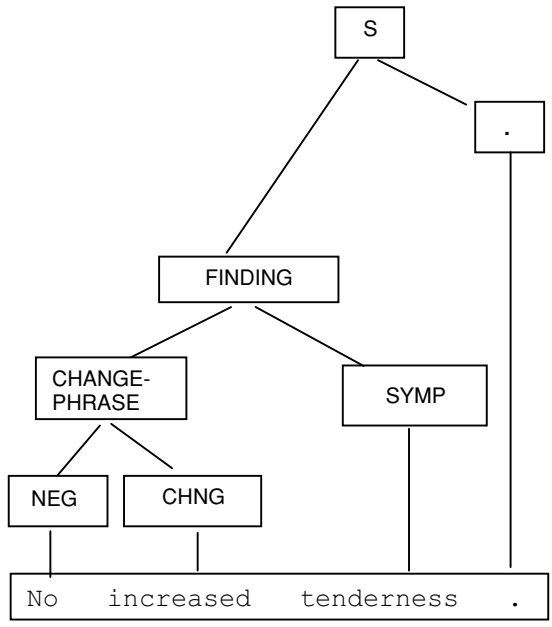

Figure 8.6. Another parse tree for the sentence no increased tenderness according to the grammar shown in Figure 8.4. This shows the correct interpretation because no modifies increased, which modifies tenderness.

component (Sager et al., 1987). After the syntactic structures are recognized, they are followed by syntactic rules that regularize the structures. For example, passive sentences, such as `the chest X-ray was interpreted by a radiologist`, would be transformed to the active form (e.g., `a radiologist interpreted the chest X-ray`). Another set of semantic rules would then operate on the regularized syntactic structures to interpret their semantic relations.

8.4.5 Pragmatics

Syntactic and semantic components of NLP systems evaluate each sentence in isolation. Complete analysis of a text (e.g., a clinical note or journal article) requires analysis of relationships between sentences and larger units of discourse, e.g., paragraphs and sections (Jurafsky and Martin, 2000, pp. 669–696). One of the most important mechanisms in language for creating linkages between sentences is the use of **referential expressions**, which include pronouns (`he, she, her, himself`), proper nouns (`Dr. Smith, Atlantic Hospital`), and noun phrases modified by the definite article or a demonstrative (`the left breast, this medication, that day, these findings`).

Each referential expression has a unique **referent** that must be identified in order to make sense of the text. The following text contains several examples. The proper noun `Dr. Smith` refers to the physician treating the patient. In clinical text, proper nouns can also refer to patients, family members, and departments and patient care institutions. In scientific discourse, proper nouns typically refer to scientists and research

institutions. In the first two sentences, `his` and `he` refer to the patient, while `he` refers to the physician in the fourth sentence. There are several definite noun phrases (e.g., `the epithelium, the trachea,` and `the lumen`), which have to be resolved. In this case, the referents are parts of the patient's body.

```
His laboratory values on admission were notable for a chest
X-ray showing a right upper lobe pneumonia. He underwent upper
endoscopy with dilatation. It was noted that his respiratory
function became compromised each time the balloon was dilated.
Subsequently, Dr. Smith saw him in consultation. He performed
a bronchoscopy and verified that there was an area of tumor.
It had not invaded the epithelium or the trachea. But it did
partially occlude the lumen.
```

Automatic resolution of referential expressions can draw on both syntactic and semantic information in the text. Syntactic information for resolving referential expressions includes:

- *Agreement* of syntactic features between the referential phrase and potential referents
- *Recency* of potential referents (nearness to referential phrase)
- *Syntactic position* of potential referents (e.g., subject, direct object, object of preposition)
- The *pattern of transitions* of topics across the sentences

Syntactic features that aid in resolution include such distinctions as singular/plural, animate/inanimate, and subjective/objective/possessive. For example, pronouns in the above text carry the following features: `he` (singular, animate, subjective), `his` (singular, animate, possessive), and `it` (singular, inanimate, subjective/objective). Animate pronouns (`he, she, her`) almost always refer humans. The inanimate pronoun `it` usually refers to things (e.g., `it had not invaded`), but sometimes does not refer to anything when it occurs in "cleft" constructions: `it was noted, it was decided to` and `it seemed likely that.`

Referential expressions are usually very close to their referents in the text. In `it had not invaded`, the pronoun refers to the immediately preceding noun phrase `area of tumor.` The pronoun in `it did partially occlude` has the same referent, but in this case there are two intervening nouns: `epithelium` or `trachea`. Thus, a rule that assigns pronouns to the most recent noun would work for the first case, but not for the second.

The syntactic position of a potential referent is an important factor. For example, a referent in subject position is a more likely candidate than the direct object, which in turn is more likely than an object of a preposition. In the fifth sentence of the text above, the pronoun `he` could refer to the patient or to the physician. The proper noun `Dr. Smith` is the more likely candidate, because it is the subject of the preceding sentence.

Centering theory accounts for reference by noting how the center (focus of attention) of each sentence changes across the discourse (Grosz et al., 1995). In the above text, the patient is the center of the first three sentences, the physician is the center of the fourth and fifth sentence, and the area of tumor is the center of the last sentence. In this approach, resolution rules attempt to minimize the number of changes in centers. Thus,

in the above text it is preferable to resolve he in sentence five as the physician rather than the patient because it results in smoother transition of centers.

Semantic information for resolving referential expressions involves consideration of the semantic type of the expression, and how it relates to potential referents (Hahn et al., 1999).

- Semantic type is the same as the potential referent.
- Semantic type is a subtype of the potential referent.
- Semantic type has a close semantic relationship with the potential referent.

For example, in the following text, the definite noun phrase the density must be resolved. If the phrase a density occurred previously, this would be the most likely referent. Instead, the phrase a spiculated nodule is selected since nodule and density have closely related semantic types. In the previous text, the noun phrase the balloon is also definite and requires resolution. Since there is no previous noun of similar type, it is necessary to establish a semantic relationship with a preceding noun. The word dilation is the best candidate because a balloon is a medical device used by that procedure.

The patient's gynecologist palpated a mass in the left breast on September 10, 1995. The patient had a mammogram, which showed a spiculated nodule at the two to three o'clock position in the left breast, which was not present in 1994. There were also microcalcifications medial and inferior to the density.

Temporal expressions are another important linguistic mechanisms for connecting the events in a discourse (Sager et al., 1987, pp. 175–194). For example, in the above text the mammogram occurs after the palpation event, and we are told that the nodule was not present before the palpation. There are many different kinds of temporal expressions. The dates in the above example locate an event at a point or interval in time. Other examples include at 7:30am, on Tuesday, August 1, 2003, in Fall 1998. Additional expressions can be used to position an event relative to a position in time, e.g., yesterday, this morning, last summer, and two years ago, a few days before admission, several weeks later and since age 12.

In the absence of temporal expressions, time in a narrative tends to flow forward. This rule enables one to determine that the mammogram occurred at or after the palpation in the above text. Temporal conjunctions (e.g., before, after, while, and when) are used to directly relate two events in time. Temporal modifiers are used to specify the duration of an event (e.g., 4 hours, all week, half a year), and frequency (e.g., twice, every hour, and 1 pack per day).

8.5 Challenges of Clinical Language

NLP is challenging for general language, but there are issues that are particularly germane in the clinical domain, which are discussed below.

Good performance: If the output of an NLP system is to be used to help manage and improve the quality of clinical care and to facilitate research, it must have high enough

sensitivity, accuracy, and specificity for the intended clinical applications. Different applications require varying levels of performance; however, generally, the performance should not be significantly worse than that of medical experts or the model organism database curators. This requirement means that before an application involving NLP can be used for a practical application, it would have to be evaluated so that the adequacy of performance can be measured. Additionally, since there is typically a trade-off between sensitivity and specificity, a system should be flexible to maximize the appropriate measure that is needed by the application.

Recovery of implicit information: Many health care reports are very compact and often omit information that can be assumed by other experts. An automated system may need to capture the implicit information in order to perform a particular task, necessitating that either the NLP system itself or the application using the output of the NLP system contains enough medical knowledge to make the appropriate inferences that are necessary in order to capture implicit information. For example, in one evaluation study we performed, medical experts when reading the sentence in an obstetrical report `she had been ruptured times 25 1/2 hours` inferred that `rupture` meant `rupture of membranes` because they related their domain knowledge to the context.

Intraoperability: In order to be functional in a clinical environment, an NLP system has to be seamlessly integrated into a clinical information system, and generate output that is in a form usable by other components of the system. This generally means that:

- The system will have to handle many different interchange formats (i.e., Extensible Markup Language (XML), HL7).
- The system will have to handle different formats that are associated with the different types of reports. Some reports often contain tables with different types of configurations. For example, Figure 8.7 shows portions of a cardiac catheterization report. Some of the sections contain text (i.e., procedures performed, comments, general conclusions), some consist of structured fields (i.e., height, weight) that are separated from each other by white space, and some consist of tabular data (i.e., pressure). The structured fields are easy for a human to interpret but are problematic for a general NLP program, because white space and indentation rather than linguistic structures determine the format of the table.
- The NLP system has to generate output that can be stored in an existing clinical repository. However, the output often has complex and nested relations, and it may be complicated or impossible to map the output to the database schema without substantial loss of information. Depending on the database schema loss of information may be unavoidable. An alternative approach would involve designing a complex database that can store nested data and data with a wide range of modifiers to accompany the NLP system. Such a database has been in use at Columbia Presbyterian Medical Center (CPMC) since the early 1990s (Friedman et al., 1990; Johnson et al., 1991), and has been critical for the effective use of NLP technology at CPMC.
- The underlying clinical information system may require a controlled vocabulary for data that are used for subsequent automated applications. This necessitates that the output of the NLP system be mapped to an appropriate controlled vocabulary;

Procedures performed: Right Heart Catheterization
Pericardiocentesis

Complications: None
Medications given during procedure: None
Hemodynamic data
Height (cm.): 180 Weight (kg): 74.0
Body surface area (sq. m 93): 1. Hemoglobin (gm/dl):
Heart rate: 102

Pressure (mmHg)

	Sys	Dias	Mean	Sat
RA	14	13	8	
RV	36	9	12	
PA	44	23	33	62%
PCW	25	30	21	

Conclusions: Postoperative cardiac transplant
Abnormal hemodynamics
Pericardial effusion
Successful pericardiocentesis
General comments:
1600cc of serosanguinous fluid were drained from the pericardial sac with improvement in hemodynamics.

Figure 8.7. A portion of an actual cardiac catheterization report.

sometimes the vocabulary is homegrown and sometimes it is a standard vocabulary, such as the UMLS (Lindberg et al., 1993), *Systematized Nomenclature of Medicine* (SNOMED) (Côté, et al., 1993), or *International Classification of Diseases* (ICD-9), ninth edition (World Health Organization, 1990). Since natural language is very expressive and varied, most likely, there will be important terms that will have no corresponding controlled vocabulary concept, and a method for handling this type of situation will have to be designed. Additionally, the NLP system has to be capable of mapping to different controlled vocabularies, depending on the application.

Interoperability: NLP systems are time consuming and difficult to develop, and in order to be operational for multiple institutions and diverse applications, they would minimally have to generate output containing a controlled vocabulary. It would be ideal if the controlled vocabulary were one of the "standard" vocabularies. Otherwise, explicit definitions of the controlled vocabulary terms would be needed for each institution or application. In addition to a controlled vocabulary, a standard representational model for medical language is needed in order to represent important relations, such as negation, certainty, severity, change, and temporal information that are associated with the clinical terms. Since there is no standardized language model currently, an understanding of the model generated by each NLP is necessary in order for automated applications to use NLP output appropriately. An effort to merge different representational

models of medical language to create a widely used model for medical language was undertaken by a large number of researchers called *The Canon group* (Evans et al., 1994). That effort resulted in a common model for radiological reports of the chest (Friedman et al., 1995), but the model was not actually utilized by the different researchers.

Training sets for development: Development of NLP systems is based on analysis (manual or automated) of samples of the text to be processed. In the clinical domain, this means that large collections of online patient records in textual form must be available for training the NLP systems. This is very problematic because many NLP researchers do not have direct ties to clinical information systems. Access to online patient records is confidential, requires the approval of **institutional review boards** (IRB), and generally necessitates removal of identifying information. Removal of identifying information from structured fields is straightforward; however, identifying information occurring in the text itself, such as names, addresses, phone numbers, unique characteristics (i.e., Mayor of New York) make this task extremely difficult. Ideally, for transferability among different health care institutions, data from a large number of different institutions is desirable so that the NLP system is not trained for a particular institution, but because of patient confidentiality, the data is difficult, if not impossible, to obtain. The problem is slightly different when processing the literature. For example, the abstracts can be obtained through the Medline database and are available to the public. Additionally, Pubmed Central (Wheeler et al., 2002) and other electronic journals provide full text articles.

Evaluation: Evaluation of an NLP system is critical but difficult in the health care domain because of the difficulty of obtaining a **gold standard** and because it is difficult to share the data across institutions. A fuller discussion on evaluation of NLP systems can be found in (Friedman et al., 1997; Hripcsak and Wilcox, 2002). Generally, there is no gold standard available that can be used to evaluate the performance of an NLP system. Therefore, for each evaluation, recruitment of subjects who are medical experts is generally required to obtain a gold standard for a test set. There are several ways an evaluation can be carried out. One way involves having experts determine if all the information and relations are correctly extracted and encoded based on the test set of text reports. Obtaining a gold standard for this type of evaluation is very time consuming and costly, since medical experts would have to structure and encode the information in the reports manually. For example, in a study that was performed associated with SNOMED encoding, it took a physician who was experienced in coding 60 hours to encode all the clinical information in one short emergency room report (Lussier et al., 2001).

Another way to carry out an evaluation would be to evaluate performance of a clinical application that uses the output generated by an NLP system. This type of evaluation would not only evaluate the information and relations captured by the system, but would also evaluate the accessibility and practical utility of the structured information for use with other clinical applications. An advantage of this type of evaluation is that it is generally easier for experts to provide a gold standard for this type of evaluation because they would not have to encode all the information in the report, but would only be required to detect particular clinical conditions in the reports. This is a much less time-consuming task than encoding all the data, and generally does not necessitate

special training because medical experts routinely read patient reports and interpret the information in them. To perform this type of evaluation, knowledge engineering skills as well as clinical expertise would be required in order to formulate the queries that will be needed to retrieve the appropriate information generated by the NLP system, and to make the appropriate inferences. Formulation of the query can be relatively straightforward or complex depending on the particular task. For example, to determine whether a patient experienced a change in mental status based on information in the discharge summary, many different terms associated with that concept must be searched for, such as `hallucinating, confusion, Alzheimer's disease, decreased mental status`. In addition, over 24 different modifier concepts, such as `rule out, at risk for, family history, negative, previous admission for,` and `work up` may modify the relevant terms, signifying that the change in mental status should be disregarded because the modifiers denote that the patient did not currently experience a change in mental status, but may have in the past, a family member may have experienced it, or a work up was being performed.

When evaluating a particular application using NLP output, the performance measurements would be associated with the particular application, and performance would constitute the performance of both the NLP system and the automated query. For example, if the NLP system correctly extracted the finding `confusion` from the report but the query did not include that condition, there could be a loss of sensitivity; similarly, if the query did not filter out modifier conditions, such as `negative`, there could be a loss of precision. When analyzing results for this type of evaluation study, it would be important to determine whether errors occurred within the NLP system or by the query that retrieved the reports. Additionally, it would be important to ascertain how to fix the error and how much effort would be involved. In the above examples, simple corrections would be involved: one correction would involve adding a new term, `confusion`, to the query; the second would involve adding a modifier term to the filter. Corrections to the NLP system could involve adding entries to a lexicon, which is also very straightforward. However, a more complex change would involve revising the grammar rules.

In order to obtain a better understanding of the underlying methods used by different NLP systems, an evaluation effort that is carried out by a third party, in which NLP systems can participate, is needed to allow for comparison of performance across the different systems. In the general English domain, this was accomplished for a number of years by the Message Understanding Conferences (Sundheim, 1991, 1992, 1994, 1996; Chinchor, 1998), which were supported with funding from DARPA. These intersystem evaluations not only allowed for comparison of systems but also substantially fostered the growth, improvement, and understanding of NLP systems in the general English domain. Presently, similar efforts are occurring for NLP systems in the biological community, as evidenced by the KDD Challenge, the TREC Genomics Track, the Bio**C**re**A**t**I**v**E** Assessment of Information Extraction systems in Biology (e.g., the folowing web sites are associated with NLP evaluation efforts within the bioinformatics community—http://www.biostat.wisc. edu/~craven/kddcup/, http://ir.ohsu.edu/genomics/, and http://www.pdg.cnb.uam.es/ BioLINK/BioCreative. eval.html).

Determining types of information to capture: Determining which information an NLP system should capture is an important decision. Some NLP systems may process partial

information in the report, such as admission diagnoses or chief complaints, but not the complete report. Other NLP systems may be highly specialized and may also include expert medical knowledge (i.e., knowledge to determine whether a patient has community-acquired pneumonia).

Granularity of the information: NLP systems may capture the clinical information at many different levels of granularity. One level of coarse granularity consists of classification of reports. For example, several systems (Aronow et al., 1999) classified reports as positive or negative for specific clinical conditions, such as breast cancer. Another level of granularity, which is useful for information retrieval and indexing, captures relevant terms by mapping the information to a controlled vocabulary, such as the UMLS (Aronson et al., 2001; Nadkarni et al., 2001), but modifier relations are not captured. A more specific level of granularity also captures positive and negative modification (Mutalik et al., 2001; Chapman et al., 2001), but not other types of modification (e.g., severity, time of event, duration, and frequency). An even more specific level of granularity captures all modifiers associated with the term, facilitating reliable retrieval.

Expressiveness versus ease of access: Natural language is very expressive. There are often several ways to express a particular medical concept and also numerous ways to express modifiers of the concept. For example, severity information may be expressed in more than 200 different expressions, with terms such as `faint, mild, borderline, 1+, 3rd degree, severe, extensive, and mild to moderate`. These modifiers make it more complex to retrieve reports based on NLP-structured output since such a wide variety has to be accounted for. In addition, nesting of information also adds complexity. For example, a change type of modifier, such as `improvement` (as in `no improvement in pneumonia`), would be represented using nesting: the change modifier `improvement` would modify `pneumonia` and the negation modifier, `no,` would modify `improvement`. In this situation, a query that detects changes concerned with `pneumonia` would have to look for primary findings associated with pneumonia, filter out cases not associated with a current episode, look for a change modifier of the finding, and, if there is one, make sure there is no negation modifier on the change modifier. Another form of representation would facilitate retrieval by flattening the nesting. In this case, some information may be lost but ideally only the information that is not critical. For example, `slightly improved` may not be clinically different from `improved` depending on the application. Since this type of information is fuzzy and imprecise, the loss of information may not be significant. However, the loss of a modifier `no` would be significant, and those cases should be handled specially.

Heterogeneous formats: There is no standardized structure for clinical reports, or for the format of the text within the report. Frequently, there is no period (i.e., ".") to demarcate the end of a sentence, but a new line or a tabular format is used instead. This is easy for humans to manually interpret but difficult for computers. In addition, the sections and subsections of the reports are not standardized. For example, in CPMC, there are many different section headers for reporting diagnostic findings (e.g., Diagnosis, Diagnosis on admission, Final Diagnosis, Preoperative Diagnosis, and Medical Diagnosis). In addition, section headers are frequently omitted or several sections are merged into one. For example, past clinical history and family history may

be reported in the history of present illness section. In addition, there is a lack of uniformity for specifying subsections. For example, surgical pathology reports often refer to findings for different specimens, which are mentioned throughout the report, but are not uniformly identified (e.g., the same specimen may be called `specimen A, slide A,` or just `A` within the same report).

Lack of a standardized set of domains: Knowledge of the domain being processed is important for NLP systems since a domain provides context that is often needed by the NLP system. For example, knowledge of the domain would facilitate recovery of implicit information (e.g., `mass` in a mammogram denotes `mass in breast`), or to resolve an ambiguous word or abbreviation (e.g., `pvc` in a chest X-ray denotes `pulmonary vascular congestion` whereas in an electrocardiogram it denotes `premature ventricular complexes`). Currently, there are no standard domains for naming different types of clinical documents. For example, at CPMC, there is a domain called `cardiology report`, which can correspond to an echocardiogram, catheterization diagnostic report, electrocardiography report, or stress test. Additionally, although individual radiology reports are coded, the different areas within radiology (e.g., abdomen, musculoskeletal system, etc.) have not been classified.

Large number of different clinical domains: There are a large number of different clinical domains, and a lexicon has to be developed for each domain. Each domain may be associated with its own lexicon but maintaining separate lexicons would be inefficient and error prone, since there is a significant amount of overlap among the terms. However, if one lexicon is maintained for all the domains, ambiguity increases because more terms become associated with multiple senses. For example, the term `discharge` may refer to `discharge from institution, discharge from eye, or electrical discharge from lead` (seen in a few electrocardiogram reports).

Interpreting clinical information: The interpretations of the findings may vary according to the type of report. For example, when retrieving information from the `Diagnosis Section` of a discharge summary, the interpretation will generally be more straightforward than when extracting information from radiological reports. Radiological reports generally do not contain definitive diagnoses, but contain a continuum of findings that range from patterns of light (e.g., `patchy opacity`), to descriptive findings (e.g., `focal infiltrate`) to interpretations and diagnoses (e.g., `pneumonia`). In some types of clinical reports, the descriptive findings may be expressed without further interpretation (e.g., a finding `pneumonia` may not be present in a radiological report; instead, findings consistent with `pneumonia`, such as `consolidation` or `infiltrate` may occur), or the interpretation may be included along with the descriptive findings. Therefore, in order to use an NLP system to detect pneumonia based on chest X-ray findings, the NLP system or application using the system would have to contain medical knowledge. The knowledge needed to detect a particular condition may be quite complex. In order to develop such a component, machine learning techniques could be used that involve collecting training instances, which would then be used to develop rules automatically (Wilcox and Hripcsak, 1999) or to train a Bayesian network (Christensen et al., 2002), but this may be costly since performance is impacted by sample size (McKnight et al., 2002), and, for many conditions, a large number of instances would have to be obtained for satisfactory performance. An

alternative would involve manually writing the rules by observing the target terms that the NLP system can generate along with sample output. For that situation, the rules will generally consist of combinations of **Boolean operators** (e.g., `and`, `or`, and `not`) and findings. For example, a rule, which detects a comorbidity of neoplastic disease based on information in a discharge summary, could consist of a Boolean combination of over 200 terms (Chuang et al., 2002).

Compactness of text: Generally, clinical reports are very compact, contain abbreviations, and often omit punctuation marks. Some abbreviations will be well known but others may be defined uniquely. An example of a typical resident sign-out note, which is full of abbreviations and missing punctuation, is shown in Figure 8.8. Lack of punctuation means that sentence boundaries are poorly delineated, thereby making it more difficult for NLP systems because they generally depend on recognition of well-defined sentences. Abbreviations cause problems because they are highly ambiguous and not well defined.

Interpretation depends on context: Contextual information must be included in the findings since it often affects the interpretation. The section of the report and the type of report is important for the interpretation. For example, `pneumonia` in the Clinical Information Section of a chest X-ray report may mean `rule out pneumonia` or `patient has pneumonia`, whereas the occurrence of `pneumonia` in the Diagnosis Section is not ambiguous. Similarly, `history of asthma` in the Family History Section does not mean that the patient has asthma.

Rare events: Natural language systems require a sufficient number of training examples, which are needed to refine or test the system. Certain occurrences of interest, such as medical errors and adverse events, are not always reported frequently. Thus, it may be difficult to find a large number of reports necessary for training and testing an NLP system for certain applications. For those cases, terminological knowledge sources may be helpful for providing lexical knowledge related to rare terms that may occur in text associated with the events of interest.

Occurrence of typographic and spelling errors: Clinical reports occasionally contain typographic errors, which may cause the system to lose information or to misinterpret information. Automated correction of spelling errors is difficult and could create additional errors. For example, a typographic error `hyprtension` will cause a loss of clinical information; it is not trivial to correct this error automatically without additional knowledge because it may refer to `hypertension` or `hypotension`. A particularly serious error could involve substitution of a similar sounding medical term. For

```
Admit 10/23
71 yo woman h/o DM, HTN, Dilated CM/CHF, Afib s/p embolic event,
chronic diarrhea, admitted with SOB.  CXR pulm edema.  Rx'd Lasix.
All:  none
Meds Lasix 40mg IVP bid, ASA, Coumadin 5, Prinivil 10, glucophage
850 bid, glipizide 10 bid, immodium prn
Hospitalist = Smith  PMD = Jones  Full Code, Cx >101
```

Figure 8.8. An example of a resident sign-out note.

example, the drug Evista may be misspelled E-Vista, which is a different drug. This type of error is troublesome not only for automated systems but also for medical experts when reading the information manually.

Limited availability of electronic records: Not all clinical documents are in electronic form. At many hospitals daily clinical notes (such as nursing notes and progress notes) are recorded in the paper chart but are not available online; however, the information they contain is critical for patient care. NLP systems must have the documents available electronically in textual form in order to process them. A scanner could be used to obtain the documents in electronic form as image files, but then **optical character recognition** (OCR) technology would have to be used to obtain textual versions of the documents. However, OCR technology is generally not accurate enough for this purpose, especially since human experts often find the documents difficult to read.

8.6 Challenges for Biological Language Processing

Dynamic nature of domain: The biomolecular domain is very dynamic, continually creating new names for biomolecular entities and withdrawing older names. For example, for the week ending July 20, 2003, the Mouse Genome Informatics Web site (Blake et al., 2003) reported 104 name changes, representing changes related only to the mouse organism. If the other organisms being actively sequenced were also considered, the number of name changes during that week would be much larger.

Ambiguous nature of biomolecular names: Short symbols consisting of two to three letters are frequently used that correspond to names of biomolecular entities. Since the number of different combinations consisting of only a few letters is relatively small, it is highly likely that this would lead to names that correspond to different meanings. For example, to, which is a very frequent English word, corresponds to two different Drosophila genes and to the mouse gene tryptophan 2,3-dioxygenase. Another situation that contributes to the amount of ambiguity in gene names is that the different model organism groups name genes and other entities independently of each other, leading to names which are the same but which represent different entities. The ambiguity problem is actually worse if the entire biomedical domain is considered. For example, cad represents over 11 different biomolecular entities in Drosophila and the mouse but it also represents the clinical concept coronary artery disease. Another contributing factor to the ambiguity problem is due to the different naming conventions for the organisms. These conventions were not developed for NLP purposes but for consistency within the individual databases. For example, Flybase states that "Gene names must be concise. They should allude to the gene's function, mutant phenotype or other relevant characteristic. The name must be unique and not have been used previously for a Drosophila gene." This rule is fairly loose and leads to ambiguities.

Large number of biomolecular entities: The number of entities in this domain is very large. For example, there are about 70,000 genes when considering only humans, fly, mouse, and worm, and the number of corresponding proteins is over 100,000. Additionally, there are over 1 million species as well as a large number of cell lines and small molecules. Having such a large number of names means the NLP system has to

keep a very large knowledge base of names or be capable of dynamically recognizing the type by considering the context. When entities are dynamically recognized without use of a knowledge source, it would be very difficult to identify them within an established nomenclature system.

Variant names: Names are created within the model organism database communities, but they are not always exactly the same as the names used by authors when writing articles. There are many ways authors may vary the names (particularly long names), which leads to difficulties in name recognition. This is also true in the medical domain, but the problem is exacerbated in the biomolecular domain because of the frequent use of punctuation, and other special types of symbols. Some of the more common types of variations are due to punctuation and use of blanks (`bmp-4, bmp 4, bmp4`), numerical variations (`syt4, syt IV`), variations containing Greek letters (`iga, ig alpha`), and word order differences (`phosphatidylinositol 3-kinase, catalytic, alpha polypeptide, catalytic alpha polypeptide phosphatidylinositol 3-kinase`).

Nesting of names: The names of many biomolecular entities are long and contain substrings that are also names. For example, `caspase recruitment domain 4` and `caspase` both correspond to gene names; if a variant form of `caspase recruitment domain 4` occurs in an article and the entire name is not recognized by the NLP system, the substring `caspase` would be recognized in error.

Lack of a standard nomenclature: The different model organism communities have different nomenclatures, each of which are standard for a particular organism. Each of the communities maintains a database that names the entities, provide unique identifiers, and list synonyms and preferred forms. However, each community maintains different databases that have different schemas and taxonomies; therefore, an NLP system has to obtain the knowledge needed from a diverse set of resources. Although Gene Ontology (GO) (Gene Ontology Consortium, 2003) is a consortium that aims to produce a uniform controlled vocabulary that can be applied to all organisms (even as knowledge of gene and protein roles in cells accumulates and changes), it applies only to biological functions, processes, and structures.

Heterogeneity of the text: Many abstracts can be obtained from Medline. These are easy to process because they can be obtained in the form of plain text. However, a substantial portion of biomolecular information occurs only in the full journal articles, which have different file formats. They may occur as Portable Document Format (PDF), Hypertext Markup Language (HTML), or XML files, which must first be converted to plain text. PDF file formats cannot be easily converted to text; although there is software that is commercially available to perform the conversion, it is currently error prone. Additionally, HTML files are suitable for presentation of the file in a browser, but cannot be relied on for specifying the semantics of the information. For example, it may be possible to recognize a section such as "**Introduction**" because it is enclosed in a tag consisting of a large bold font. An additional problem is that some of the important information may be in a figure which is in graphic format, and therefore not accessible as text. For example, in chemical journals, the names of chemical compounds often appear as a single letter followed by the full name and the diagram. In the text of the article, the single letter appears in place of the name, causing a loss of information. An

additional problem is that the journals are often available only through subscriptions, which can be costly.

Complexity of the language: The structure of the biological language is very challenging. In clinical text, the important information is typically expressed as noun phrases, which consists of descriptive information such as findings and modifiers. In biomolecular text, the important information usually consists of interactions and relations, which are expressed as verbs or noun phrases that are frequently highly nested. Verb phrases are generally more complex structures than noun phrases. The arguments of a verb are important to capture as well as the order of the arguments (e.g., `Raf-1 activates Mek-1` has a different meaning than `Mek-1 activates Raf-1`). A typical sentence usually contains several nested interactions. For example, the sentence `Bad phosphorylation induced by interleukin-3 (IL-3) was inhibited by specific inhibitors of phosphoinositide 3-kinase (PI 3-kinase)` consists of four interactions (and also two parenthesized expressions specifying abbreviated forms). The interaction and the arguments are illustrated in Table 8.7. The nested relations can be illustrated more clearly as a tree (see Figure 8.9). Notice that the arguments of some interactions are also interactions (i.e., the second argument of

Table 8.7. Nested interactions extracted from the sentence *Bad phosphorylation induced by interleukin-3 (IL-3) was inhibited by specific inhibitors of phosphoinositide 3-kinase (PI 3-kinase)*. A "?" denotes that the argument was not present in the sentence.

Interaction	Argument 1 (agent)	Argument 2 (target)	Interaction id
Phosphorylate	?	Bad	1
Induce	Interleukin-3	1	2
Inhibit	?	Phosphoinositide 3-kinase	3
Inhibit	3	1	

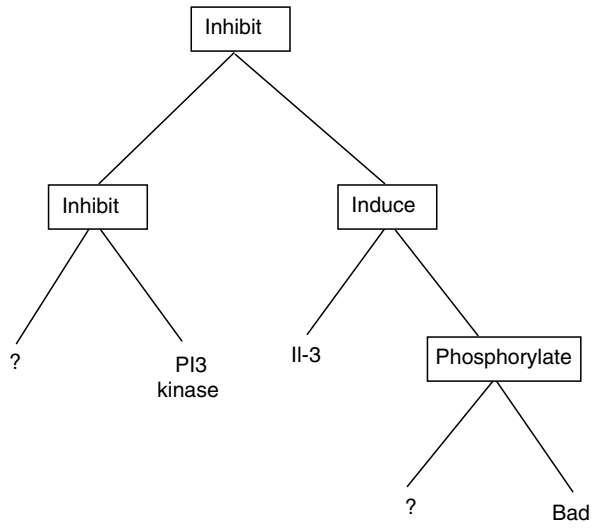

Figure 8.9. A tree showing the nesting of biomolecular interactions that are in the sentence Bad phosphorylation induced by interleukin-3 (IL-3) was inhibited by specific inhibitors of phosphoinositide 3-kinase (PI 3-kinase).

induce is phosphorylate). Also note that an argument which is not specified in the sentence is represented by a "?".

Multidisciplinary nature: In order for NLP researchers to work on biological text to extract the appropriate information, they need some knowledge of the domain. This is a big challenge because the understanding requires knowledge of biology, chemistry, physics, mathematics, and computer science.

8.7 Biomedical Resources for NLP

A number of controlled vocabularies provide terminological knowledge for NLP systems in the biomedical domain:

- UMLS (including the Metathesaurus, Semantic Network, the Specialist Lexicon; see Chapter 7)—can be used as a knowledge base and source for a medical lexicon. The Specialist Lexicon provides detailed syntactic knowledge for words and phrases, and includes a comprehensive medical vocabulary. It also provides a set of tools to assist in NLP, such as a lexical variant generator, an index of words corresponding to UMLS terms, a file of derivational variants (e.g., abdominal, abdomen), spelling variants (e.g., fetal, foetal), and a set of neoclassical forms (e.g., heart, cardio). The UMLS Metathesaurus provides the concept identifiers, and the Semantic Network specifies the semantic categories for the concepts. The UMLS also contains the terminology associated with various languages (e.g., French, German, Russian).
- Other controlled vocabularies (e.g., SNOMED, ICD-9, *Laboratory Observations, Identifiers, Names and Codes* (LOINC)) can also be used as sources of lexical knowledge for NLP. These vocabularies are also valuable as multilingual resources. For example, SNOMED was used as a lexical resource for French (Zweigenbaum and Courtois, 1998), and ICD was used as a resource for development of an interlingua (Baud et al., 1998).
- Biological databases. These include Model Organism Databases, such as Mouse Genome Informatics (Blake et al., 2003), the Flybase Database (Flybase Consortium, 2003), the WormBase Database (Todd et al., 2003), and the Saccharomyces Database (Issel-Tarver et al., 2001), as well as more general databases GenBank (Benson et al., 2003), Swiss-Prot (Boeckmann et al., 2003), LocusLink (Pruitt et al., 2001).
- GENIA corpus (Ohta et al., 2002). This corpus currently contains over 2,500 abstracts taken from Medline, which are related to transcription factors in human blood cells. It has over 100,000 hand-annotated terms marked with syntactic and semantic information appropriate for the biological domain, and is valuable for use as a gold standard for evaluation and training data for machine learning techniques. It also has an accompanying ontology.

Acknowledgments

The material in this chapter is derived in part from work sponsored by the U.S. Government for the National Academies. Any opinions, findings, conclusions, or

recommendations expressed in the material are those of the authors and do not necessarily reflect the views of the U.S. Government, or the National Academies.

Suggested Readings

Allen J. (1995). *Natural Language Understanding* (2nd ed.). Redwood City, CA: Benjamin Cummings.
This textbook, which is intended for computer scientists and computational linguists, provides a description of the theories and techniques used in natural language understanding. The focus is on the use of symbolic rule-based methods, and syntactic, semantic, and discourse levels of processing.

Charniak E. (1993). *Statistical Language Learning*. Cambridge: MIT Press.
This is the first textbook covering statistical language processing, which is intended for computer scientists. It is brief, but it is clearly written.

Friedman C. (Ed.) (2002). Special issue: Sublanguage. *Journal of Biomedical Informatics*, 35(4).
This special issue on sublanguage includes six articles by leading researchers on current work in sublanguage processing in the biomedical domain.

Harris Z., Gottfried M., Ryckmann T., Mattick Jr. P., Daladier A., Harris T.N., Harris S. (1989). *The Form of Information in Science: Analysis of an Immunology Sublanguage*. Reidel, Dordrecht: Boston Studies in the Philosophy of Science.
This book offers an in-depth description of methods for analyzing the languages of biomedical science. It provides detailed descriptions of linguistic structures found in science writing and the mapping of the information to a compact formal representation. The book includes an extensive analysis of 14 full-length research articles from the field of immunology, in English and in French.

Jurafsky D., Martin J.H. (2000). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition*. New York: Prentice-Hall.
This is an excellent textbook, which comprehensively covers in-depth methods in natural language processing, computational linguistics, and speech recognition. The natural language-processing methods include symbolic and statistical models, and also covers a broad range of practical applications.

Manning C.D., Schütze H. (1999). *Foundations of Statistical Natural Language Processing*. Cambridge: MIT Press.
This textbook contains a comprehensive introduction to statistical natural language processing, and includes the theories and algorithms needed for building statistical NLP systems.

Sager N., Friedman C., Lyman M.S. (1987). *Medical Language Processing: Computer Management of Narrative Data*. New York: Addison-Wesley.
This book describes early techniques used by the Linguistic String Project, a pioneering language processing effort in the biomedical field, explaining how biomedical text can be automatically analyzed and the relevant content summarized.

Questions for Discussion

1. Develop a regular expression to regularize the tokens in lines 4–9 of the cardiac catheterization report shown in Figure 8.7 (`Complications` through `Heart Rate`).
2. Create a lexicon for the last seven lines of the cardiac catheterization report shown in Figure 8.7 (`Conclusions` through the last sentence). For each word, determine all the parts of speech that apply, using the tags in Table 8.2. Which words have more than one part of speech? Choose eight clinically relevant words in that section of the report, and suggest appropriate semantic categories for them that would be consistent with the SNOMED axes and with the UMLS semantic network.
3. Using the grammar in Figure 8.3, draw all possible parse trees for each of the sample sentences 1a, 2a, and 3a discussed in Section 8.4.3. For each sentence, indicate which parse represents the correct structure.
4. Using the grammar in Figure 8.3, draw a parse tree for the last sentence of cardiac catheterization report shown in Figure 8.7.
5. Using the grammar in Figure 8.4, draw parse trees for the following sentences: `no increase in temperature; low grade fever; marked improvement in pain; not breathing`. (Hint: some lexemes have more than one word.)
6. Identify all the referential expressions in the text below and determine the correct referent for each. Assume that the computer attempts to identify referents by finding the most recent noun phrase. How well does this resolution rule work? Suggest a more effective rule.

```
The patient went to receive the AV fistula on December 4.
However, he refuses transfusion. In the operating room it was
determined upon initial incision that there was too much edema
to successfully complete the operation and the incision was
closed with staples. It was well tolerated by the patient.
```