A COMPUTATIONAL STUDY OF LEXICALIZED NOUN PHRASES IN ENGLISH

DISSERTATION

Presented in Partial Fulfillment of the Requirements for

the Degree Doctor of Philosphy in the Graduate

School of The Ohio State University

By

Carol Jean Godby, B.A., M.A.

* * * * *

The Ohio State University
2002

Dissertation Committee:

Professor Craige Roberts, Adviser

Professor Chris Brew

Professor David Dowty

Approved by

_____

Adviser
Department of Lingistics

ABSTRACT

Lexicalized noun phrases are noun phrases that function as words. In English, lexicalized noun phrases are often realized as noun-noun compounds such as *theater ticket* and *garbage man*, or as adjective-noun phrases such as *black market* and *high school*. In specialized or technical subjects, phrases such as *urban planning*, *air traffic control*, *highway engineering* and *combinatorial mathematics* are conventional names for concepts that are just as important as single-word terms such as *adsorbents*, *hydrology*, or *aerodynamics*. But despite the fact that lexicalized noun phrases represent useful vocabulary and are cited in dictionaries, thesauri and book indexes, the traditional linguistic literature has failed to identify consistent and categorical formal criteria for identifying them.

This study develops and evaluates a linguistically natural computational method for recognizing lexicalized noun phrases in a large corpus of English-language engineering text by synthesizing the insights of studies in traditional linguistics and computational linguists. From the scholarship in theoretical linguistics, the analysis adopts the perspective that lexicalized noun phrases represent the names of concepts that are important to a community of speakers and have survived a single context of use. Theoretical linguists have also proposed diagnostic tests for identifying lexicalized noun phrases, many of which can be formalized in a computational study. From the scholarship in computational linguistics, the analysis incorporates the view that a linguistic investigation can be extended and verified by processing relevant evidence from a corpus of text, which can be evaluated using mathematical models that do not require categorical input.

In a engineering text, a small set of linguistic contexts, including *professor of*, *department of* or *studies in*, yields long lists of lexicalized noun phrases, including *public safety*, *abstract state machines*, *complex systems*, *computer graphics*, and *mathematical morphology*. The study reported here identifies lexical and syntactic contexts that harbor lexicalized noun phrases and submits them to a machine-learning algorithm that classifies the lexical status of noun phrases extracted from the text.

Results from several evaluations show that the linguistic evidence extracted from the corpus is relevant to the classification of noun phrases in engineering text. Informal evidence from other subject domains suggests that the results can be generalized.

ACKNOWLEDGMENTS

I returned to graduate school nearly 15 years after I dropped out and am now in a position to apply the results of my research in my professional work. These unlikely events converged because I have been fortunate enough to work in an intellectually rich and supportive environment. I am grateful to my managers and colleagues at OCLC, who believe in me and understand the relevance of linguistics for solving the problems presented by large repositories of machine-readable text. My deepest thanks go to Martin Dillon, who invited me to join the Office of Research and gave me an interesting problem to work on. This dissertation is the result. Joan Mitchell and Diane Vizine-Goetz supported my interest in terminology identification and invited me to many professional venues, where I could discuss my work and develop my ideas. Traugott Koch gave me permission to use a valuable data set that he and his colleagues at Lund University in Sweden collected, without which the project reported here could not have been completed.

I am also grateful to the professors and students in the Linguistics Department at Ohio State, who welcomed me back as one of their own. Their collegiality and professionalism made the work go as smoothly as possible. My heartfelt thanks go to my dissertation committee for their help in shaping my research problem into a linguistically sophisticated inquiry. Craige Roberts, my adviser, intervened several times with much-needed encouragement when I was overwhelmed and ready to give up. David Dowty, always a linguist's linguist, pushed me to higher standards of scholarship than I could have achieved on my own. Chris Brew arrived at Ohio State just in time with the expertise in corpus linguistics that I needed to do this project justice.

Special thanks go to my friend Lee Jansen, who did a thorough job of editing the manuscript. Finally, I thank my other old friends who kept me sane as I worked on this project: Mark Bendig, Jeff McKibben, Rosanne Norman, Debbie Stollenwerk, Margaret Thomas and Patricia Weiland.

VITA

1954	Born – Newport News, Virginia
1977	Bachelor of Arts German The University of Delaware
1981	Master of Arts Linguistics The Ohio State University
1982	Graduate teaching associate English as a Second Language The Ohio State University
1984	Education consultant Software Productions Columbus, Ohio
1984	Courseware developer College of Medicine The Ohio State University
1988	Senior programmer/analyst Online Computer Library Center (OCLC) Dublin, Ohio
1990	Systems analyst, OCLC
1992	Associate research scientist, OCLC
1995	Research scientist, OCLC
1999-present	Senior research scientist, OCLC

PUBLICATIONS

Carol Jean Godby and Ray Reighart. 2001. Terminology identification in a collection of Web resources. In Karen Calhoun and John Riemer, (eds.), *CORC: New Tools and Possibilities for Cooperative Electronic Resource Description.* pp. 49-66. Binghamton, New York: The Hayworth Press.

Carol Jean Godby, Eric Miller and Ray Reighart. 2000. Automatically generated topic maps of World Wide Web resources. *The Annual Review of OCLC Research.* Accessible at: <http://www.oclc.org/research/publications/arr/1999/godby/topicmaps.htm>

Anders Ardö, Jean Godby, Andrew Houghton, Traugott Koch, Ray Reighart, Roger Thompson and Diane Vizine-Goetz, 2000. Browsing engineering resources on the Web. In Clare Beghtol, Lynne Howarth, and Nancy Williamson, (eds.), *Dynamism and Stability in Knowledge Organisation.* pp. 385-390. Würzburg, Germany: Ergon Verlag.

Carol Jean Godby and Ray Reighart. 2000. Using machine-readable text to update the Dewey Decimal Classification. *Advances in Classification Research,* pp. 21-34.

Carol Jean Godby and Ray Reighart. 1999. The WordSmith indexing system. *The Annual Review of OCLC Research*. Accessible at: <http://www.oclc.org/research/publications/arr/1998/godby_reighart/wordsmith.htm>

Diane Vizine-Goetz and Carol Jean Godby. 1998. Library classification schemes and access to electronic collections: enhancement of the Dewey Decimal Classification with supplemental vocabulary. *Advances in Classification Research*, pp. 14-25.

Diane Vizine-Goetz, Carol Jean Godby and Mark Bendig. 1995. Spectrum: A Web-based system for describing Internet resources. *Computer Networks and ISDN Systems*. 27:985-1002.

FIELDS OF STUDY

Major Field: Linguistics

TABLE OF CONTENTS

Page

Abstract iii

Acknowledgments v

Vita vi

List of Tables x

List of Figures xiii

Chapters:

1. Words that Masquerade as Phrases 1
   1.0. Introduction 1
   1.1. The automatic identification of noun phrases 2
      1.1.1. Noun phrases in the information-retrieval task 2
      1.1.2. Noun-phrase collocations 7
   1.2. Perspectives from theoretical linguistics 8
      1.2.1. Syntactic properties of lexicalized noun phrases 8
      1.2.2. Lexicalized noun phrases and compositional semantics 12
   1.3. Toward a refined computational procedure 15
   1.4. An empirical study of lexicalized noun phrases 19
   1.5. The organization of this dissertation 24

2. Algorithms for Extracting Noun Phrases from Text 25
   2.0. Introduction 25
   2.1. Four components in a system for recognizing lexicalized noun phrases 26
      2.1.1. Part-of-speech tagging 26
      2.1.2. Syntactic parsing 29
      2.1.3. Identifying lexicalized noun phrases using statistical filters 32
         2.1.3.1. Measures of association 33
         2.1.3.2. Using measures of association to identify lexicalized phrases 36
      2.1.4. The assignment of internal structure 39
         2.1.4.1. Algorithms for assigning internal structure 41
         2.1.4.2. An extension 42
   2.2. A system architecture for recognizing lexicalized noun phrases 45

3. Corpus Evidence for Lexicalization 47
   3.0. Introduction 47
   3.1. A first look at the engineering corpus 48

3.2. A look at the engineering thesaurus....52
3.3. Toward the identification of lexicalized noun phrases from corpus evidence....62
3.4. The corpus contexts of lexicalized noun phrases....70
3.4.1. Contexts for names of disciplines....71
3.4.2. The contexts of quotation....73
3.4.3. Syntactic contexts....75
3.5. Extending the analysis....79
3.6. Toward a computational analysis of local context....82

4. A Machine-Learning Study....89
4.0. Introduction....89
4.1. Computational lexicography as a machine-learning application....91
4.2. The identification of attributes....94
4.2.1. Lexical attributes....97
4.2.2. Syntactic attributes....106
4.2.2.1. Conjunctions....105
4.2.2.2. Phrase structure....110
4.2.3. The linguistic attributes: a summary....113
4.3. The training phase....115
4.4. The test phase....125
4.4. Summary and conclusions....130

5. Concluding Remarks....131
5.0. Syntactic and lexicalized noun phrases in coherent text....131
5.1. A theory of common usage....134
5.2. Some extensions....138
5.3. Future prospects....139

Bibliography....141

LIST OF TABLES

Table Page

2.1. A contingency table for two variables 33

2.2. A contingency table for *steel scrap* 34

2.3. Contingency table for *this file* 34

2.4. An extended contingency table 37

2.5. Bigrams with high and low log-likelihood values 39

2.6. Astronomy terms that require modification 44

3.1. Frequency distributions for five nouns and noun phrases in six partitions of the engineering corpus 49

3.2. Selected entries from the Engineering Information Thesaurus 53

3.3. Token sizes and syntactic forms of the entries in a random sample from in the Engineering Information Thesaurus 54

3.4. Citations of thesaurus entries in the engineering corpus 56

3.5. Token sizes and syntactic forms of the 1000-word sample from the Engineering Information Thesaurus that are cited in the engineering corpus 57

3.6. Log-likelihood summary statistics for two classes of noun phrases in the first partition of the engineering corpus 59

3.7. Citations of *artificial* in noun phrases other than *artificial intelligence* 60

3.8. The distribution of *integrate** in the first partition of the engineering corpus 61

3.9. The distribution of *small* in the first partition of the engineering corpus 62

3.10. Semantic classes of adjectives and nouns that rarely occur in lexicalized noun phrases 65

3.11. Noun-phrase bigrams created from two classes of adjectives 67

3.12. Log-likelihood scores of two classes of noun phrase bigrams in the first partition of the engineering corpus 68

3.13. Names of disciplines in the engineering corpus 72

3.14. Objects of …*known as*, *is referred to*, *also known as* and *so-called* 74

3.15. Conjunctions from the engineering corpus involving lexicalized noun phrases 77

3.16. Lexicalized noun-phrase modifiers of compound nouns 78

3.17. Local syntactic contexts for *hanging chad* 79

3.18. Some common noun-phrase heads in two collections of documents 81

3.19. Categorizations of noun phrases using two sources of corpus evidence 84

3.20. Noun phrases related by *such as* in engineering text 86

4.1. Local contexts for cote 90

4.2. The 50 most frequent context-dependent adjectives and noun-phrase bigrams in the first partition of the engineering corpus 100

4.3. Distribution of positive lexical cues across the training portion of the corpus 101

4.4. Distribution of negative lexical cues across the training portion of the corpus 102

4.5. Average log-likelihoods of noun phrases in lexical and non-lexical contexts 103

4.6. Cross-tabulations of noun phrases and contexts 104

4.7 Frequencies of unique conjoined noun phrases in six partitions of the engineering corpus 109

4.8. Log-likelihoods of conjoined noun phrases 110

4.9. Raw frequencies of unique noun-phrase heads and modifiers in six partitions of the engineering corpus 112

4.10. Log-likelihoods of noun-phrase heads and modifiers 112

4.11. Counts of training contexts in all six partitions of the engineering corpus 114

4.12. Hypothetical co-occurrences of attributes with training data 117

4.13. Cross-validation results for all linguistic variables.....120

4.14. Cross-validation results for linguistic variables and log-likelihood 122

4.15. The relative contributions of attributes to the cross-validation results 123

4.16. Performance on unseen data of known status in the training corpus 124

4.17. Some new classifications in the training corpus 125

4.18. Classifications in the test data.....127

4.19. Agreements between the classification algorithm and human experts 129

4.20. Kappa scores for all pairs of judges 130

5.1. Citations of *overall survival* in a corpus of medical text 137

LIST OF FIGURES

Figure Page

2.1. Two part-of-speech (POS) tag assignments for a structurally ambiguous sentence 27

2.2. POS tag assignments for the declarative reading of *Time flies like an arrow* 28

2.3. POS tag assignments for the imperative reading of *Time flies like an arrow* 28

2.4. Patterns of noun phrases recognized by a template matcher 32

2.5. A hypothetical Zipf curve 35

2.6. Correct and incorrect noun-phrase hierarchies 40

2.7. Process flow for extracting lexicalized noun-phrase candidates from text 46

3.1. A document in the engineering corpus to which the noun-phrase parser has been applied 50

3.2. A discourse context for *character issue* 82

4.1. A sample ARFF file containing classified noun phrases 95

4.2. Process flow for identifying noun phrases in selected lexical contexts 105

4.3. Process flow for identifying noun phrases in conjunctions 106

4.4. Conjunctions with lexicalized noun phrases 107

4.5. Process flow for identifying noun-phrase structure 111

4.6. Process flow for constructing ARFF files 113

CHAPTER 1

WORDS THAT MASQUERADE AS PHRASES

1.0. Introduction

This dissertation is about English expressions, such as *high school*, *poison ivy* and *police dog*, which I will call 'lexicalized noun phrases' because they are multi-word phrases that can function as the subject or object of a sentence and have been collected and defined in dictionaries. Though lexicalized noun phrases have been described under many labels, most readers probably recognize the examples I have cited as compound nouns like those they learned about in grammar-school language-arts classes. But in the ensuing discussion, I will argue that the meaning of the phrase *compound noun* is too narrow to cover all of the interesting cases, and that other labels introduced by linguists and lexicographers have similar shortcomings.

The above examples may seem unremarkable because they are ordinary enough to be part of everyday discourse, but lexicalized noun phrases that are closer to the edge of our linguistic knowledge are more mysterious, and perhaps more valuable because they hold the key that can unlock the rich stores of textual information now freely available on the Web. For example, if I take more than a casual interest in the bear-shaped cookie jar that has been sitting on the top of my parents' refrigerator for the past forty years, I might start with a query to the Web. But which query? *Cookie jars*? *Pottery*? *Teddy bears*? *McCoy*, the word stamped on the bear's bottom? These queries get me somewhere, but if I use the lexicalized noun phrase *art pottery*, I discover a set of documents that describe the social context in which ceramic teddy bear cookie jars are produced and appreciated. Within a few minutes, I learn that *art pottery* is usually understood to consist of the antique and

collectible products of defunct pottery factories. I also learn that the McCoy factory was located in Ohio, as were many other long-closed factories whose output is now prized by collectors, including Roseville, Rookwood and Hall.

The purpose of the research reported here is to develop improved automated methods for identifying lexicalized noun phrases in machine-readable text. Many concepts, especially in esoteric or technical subjects, have phrasal names and are constantly evolving in the living world documented by the Web. For example, ten years ago, the concepts behind *Internet service providers*, *digital signatures*, *semantic Web*, and *ontology interchange language* did not exist, but there are now hundreds of Web pages devoted to the exploration of these topics. To make the knowledge in these pages accessible to those of us who are not specialists in the development of Internet standards for data exchange, we can write computer software that identifies the most significant noun phrases and collect them in subject indexes, lexicons, dictionaries and knowledge bases, creating reference works that give the novice a place to begin a serious inquiry. But this narrow and practical goal masks a multi-disciplinary subject that goes far beyond lexicography and computational linguistics, reaching into theoretical linguistics and philosophy, as well as the psychology and sociology of linguistic behavior. We need to review the treatment of lexicalized noun phrases from these perspectives these to learn how best to frame the problem at hand. What is special and difficult about these expressions?

1.1. The automatic identification of noun phrases

1.1.1. Noun phrases in the information-retrieval task

As a starting point for discussion, it is worth taking a closer look at the act of formulating an English-language query to a search engine because it provides an anchor for describing the conceptual problem that this dissertation attempts to address. Though databases have been available to the academic research community since the 1960s, computers with access to the Internet and its large stores of textual information can now be found in most libraries and classrooms in the United States, as well as a growing number of homes and offices. Increasingly, a query to a search

engine is an everyday activity, and proficient exercise of this skill is destined to become a facet of basic literacy, just as the ability to find information in dictionaries and encyclopedias is.

Consider what happens when a third-grader or a non-native speaker of American English consults an Internet search engine by issuing the query *system* to obtain information about the sun and the heavenly bodies that are Earth's closest neighbors. As literate native speakers, we know that this query can retrieve the desired information because documents about solar systems are relevant to the user's information request, but it is too ambiguous or general to be effective. Dictionaries may list several senses of *system*, including *arrangement or organization*, *a group of things working together*, and *the behavior of matter obeying the laws of chemistry and physics*. Accordingly, my recent search for *system* on Yahoo[1] returned approximately 120 articles about astronomy, scattered among 33,000 documents about The International System of Units, the University of Alabama health care system, the UNIX operating system, movie rating systems, the wheeling system in the Florida lottery, California's electricity system, and hospital patient locator systems. In other words, the astronomy articles were lost among tens of thousands of documents about no subject in particular.

In linguistic terms, what has happened? Given the current state of development of search engines for textual databases, the user was put in the position of having to second-guess the literal wording of unseen documents to satisfy a need for information. Some researchers would argue that if the user makes the query *system* more specific by attaching *solar* to it, the search engine returns only the 120 articles of interest—a much more satisfactory experience. Alternatively, the hypothesis that I will defend in this dissertation is that the effective use of a search engine exercises the same linguistic skill that is required for extracting information from a dictionary or encyclopedia: it's a test of vocabulary, and *solar system* is a word that the user must know to perform an effective search.

[1] Accessible at <http://www.yahoo.com>

But this claim has hidden complexity. In English, a word is usually identified as a single sequence of letters bounded by whitespace on a printed page, but this description fails to consider the fact that words must also have definitions that refer to objects or concepts. When it does, multi-word units such as *solar system*, *operating system* and *movie rating system* are also identified as words. These may be obvious examples of lexicalized noun phrases to literate adults, but as laymen, we get to experience the third-grader's confusion whenever we try to navigate the unfamiliar linguistic landscapes of antiques and collectibles, engineering, finance, medicine, or hundreds of other specialized subjects that are now accessible from our desktops through the Internet. Are *additional activities, astronomical bubble*, *recurrent erosion* and *wireless alphabet soup* lexicalized noun phrases, too?

Once we admit that English words can consist of more than one token, we introduce a serious problem: we can't easily identify them. In the past twenty years, researchers who study information retrieval have done many experiments that simulate the user's interaction with a search engine by identifying phrases of various kinds and submitting them as queries to textual databases. In this research tradition, phrasal queries are identified in collections of machine-readable text by simple automated methods that search for sequences of words whose parts of speech qualify them as noun phrases, perhaps filtered by length and frequency. But in study after study, the most commonly reported result is that a small number of successes are buried in a large number of failures. Noun-phrase queries to a search engine return too many unrelated documents or none at all because they simulate the user's unfair task of having to second-guess the exact wording of an unseen document, and they do it poorly. The hit-and-miss quality of these queries can be illustrated by submitting the list of noun phrases at the end of the previous paragraph to Yahoo. *Additional activities* returns about 35,000 documents on no particular subject. *Wireless alphabet soup* is the name of an electronic journal originating from a personal-computer users' group in Australia. *Astronomical bubble* returns no documents at all. But *recurrent erosion* returns 150 documents, 149 of which are about a disease of the human cornea characterized by an abrasion that reappears after it has apparently healed.

Mitra, et al. (1997) presents an excellent review of the research on noun-phrase queries in information retrieval. Unfortunately, this scholarship fails to give satisfactory answers to the most basic questions raised by this data. For example, why is *recurrent erosion* a more successful query than *additional activities* or *astronomical bubble*? The information-retrieval task raises raises an interesting linguistic problem, but we need to appeal to linguistics for a sophisticated understanding.

1.1.2. Noun-phrase collocations

I believe that the treatment of noun phrases in information retrieval research is problematic because it offers no serious account of which ones are lexicalized. But a seminal paper written by a computational linguist and a lexicographer, Church and Hanks (1990), presents a starting point for solving this problem. This study argues that when machine-readable text is processed only by filtering for parts of speech or word frequencies, a significant characteristic is missed: persistence. Statistical measures of persistence account for the fact that *solar system* and *recurrent erosion* are 'frozen' expressions. In statistical terms, this means that, when *solar* and *system* occur in a collection of specialized text, they usually occur together, and the same can be said for *recurrent* and *erosion*. Only rarely is one seen without the other. But the other noun phrases I discussed previously don't have this property. For example, the phrases *additional activities* and *solar system* may be equally frequent in some texts, but *additional activities* is not persistent because *additional* combines with hundreds of other words, forming phrases such as *additional assessment*, *additional candidates* and *additional electricity*. And so does *activities*. Smadja (1993), whose research builds on the insights of Church and Hanks, refers to persistent expressions as *collocations*, or recurrent sequences of words.

Collocation is a valuable concept in our discussion, in part because it bridges statistics with lexicography. From the lexicographer's perspective, a collocation is lexical knowledge that arises from habitual use. Words that constantly appear together in experience are eventually associated in the minds of language users and may be listed together in dictionaries and thesauri. For example, we can consult a

thesaurus to discover that *start* and *finish* are antonyms. *Start* also elicits *finish* in the psychologist's word association test, which may be an unremarkable observation until we realize that there are many words with roughly similar meanings, such as *begin* and *initiate* or *terminate* and *halt*, which don't exhibit this behavior. Only *start* and *finish* are collocations—and not, for example, *initiate* and *halt*—perhaps because we frequently encounter this pair of words as names of important features on race tracks, games and computer displays, as well as in the many verbal descriptions of such things. As Halliday and Hasan (1976) argue, a lexical collocation is one of the linguistic elements that transform a collection of words into a coherent discourse.

Lexical collocations may also consist of frozen sequences of words, as are all of the examples I have discussed, except *start* and *finish*. If so, they are treated as immutable linguistic chunks that can be produced and understood at will by competent language users at all levels of education and linguistic awareness. Ordinary people jot down *peanut butter*, *cat food* and *spaghetti sauce* in their grocery lists next to *bananas* and *yogurt*. When lexicographers create a dictionary, they write definitions for *guinea pig*, *sea wall*, *human being* and *diesel engine*, as well as for *filth* and *hyperbole*. Ophthamologists who specialize in disorders of the cornea create Web sites[2] that describe *recurrent erosion*, *corneal dystrophy* (inherited bilateral, non-inflammatory disease) and *petrygium* (scar tissue on the surface of the cornea). These examples illustrate a property that is at the heart of the definition of 'word' taught to new students of linguistics: a word is internally stable but positionally mobile. Like words, lexicalized noun phrases are a minimal unit of language that can appear in lists and many other contexts of use, and it may simply be an accident of orthography that such words have a white space in the middle.

The lexicographer's concept of collocation helps us realize that the task facing users of search engines may not be so unreasonable after all. To find information in a collection of text, users must formulate their requests in terms of vocabulary that is appropriate for their domain of interest. The experiments conducted by researchers interested in information retrieval that simulate queries to a search engine can help in this effort because the same computational techniques can be used to construct

[2] For example, <http://www.cornealdocs.com/patient_education.html#q4>

indexes that enable readers to discover vocabulary in an unfamiliar subject, much as back-of-the book indexes do. But the failure of these experiments implies that the computational methods for identifying noun phrases need to be made more sophisticated, perhaps by incorporating statistical measures of collocation.

Unfortunately, statistical measures are a noisy estimate of lexical collocation. One of the major goals of the research reported in this dissertation is to supplement measures of lexical collocation with evidence that reduces the error. The details are deferred to later chapters, but I can state the problem here because it is conceptually simple. For starters, not every linguistic pattern that scores high on measures of statistical collocation is lexical knowledge. For example, all Web pages in a collection from a large news organization may carry the same copyright statement, and the words in the statement may co-occur so frequently that the entire sentence is a statistical collocation. But this collocation probably does not belong in a dictionary. Conversely, in collections on highly specialized subjects, many lexicalized noun phrases are formed from the same small set of words. For example, in a corpus of texts about molecular biology, *cell* combines with words commonly found in coherent texts about biology, forming noun phrases such as *cell metabolism*, *cell membranes*, *cell morphology*, *cell formation*, and *cell linings*. These phrases may be listed in indexes or dictionaries of molecular biology, but are unlikely to achieve high scores on measures of statistical collocation because *cell* combines too freely with other words that are frequent in the text. Because of problems like these, the refinement of statistical models of lexical collocation is an active subject of research. Schone and Jurafsky (2001) provide a review and evaluation of recent proposals.

1.2. Perspectives from theoretical linguistics

One of the hypotheses I explore in this dissertation is that lexicalized noun phrases can be identified with greater precision if statistical collocations are supplemented with linguistic knowledge that can be obtained from coherent text. How can lexicalized noun phrases be identified, using principles of theoretical

linguistic analysis? Answers to this question form a rich tradition in theoretical studies of syntax and semantics, and highlight the significance of lexicalization as a linguistic phenomenon that deserves a sophisticated account.

1.2.1. Syntactic properties of lexicalized noun phrases

Perhaps the least controversial examples of lexicalized noun phrases are compound nouns such as *lighthouse*, *shellfish*, *doghouse*, *blockhead*, *hairpin* and *eggplant*, which have been cited by many scholars of English word formation, including Ball (1941). These phrases are compounds because they are made up of two words, and they are unquestionally lexicalized because they are written as a single token. Because noun-noun sequences are uncommon in English, except for those rare sentences in which unmodified noun-phrase constituents appear fortuitously next to one another, as in *They gave her dog biscuits*, a first definition of lexicalized noun phrases might say that all unbroken surface sequences of nouns are lexicalized. This has the advantage of subsuming the much larger class of compound nouns, including *state police*, *parish priest*, and *theater ticket*, which are not written as single words and which scholars of compounding have almost always regarded as functionally similar.

There is a compelling reason for stopping with this simple definition, as Downing (1977) did in her seminal work on English nominal compounding. Because sequences of unmodified nouns in well-formed sentences are almost always nominal compounds, their presence might carry a social message. Why would a speaker or writer choose this mode of expression instead of a sentence or a complex phrase? According to Downing, speakers use a compound noun instead of an expression that contains adjectives or other parts of speech when they wish to imply that a relationship is permanent, recurrent, or generic. For example, citing Gleitman and Gleitman (1970), Downing says that not every man who takes out the garbage is a *garbage man*, only those men who remove garbage for a living:

> Compounds, unlike full sentences, typically serve as naming devices, used to denote 'relevant categories' of the speaker's experience….In such instances, the speaker is presumably faced with a situation where he wishes to denote an entity or a member of a category which has no pre-existing name, but which merits easy access in communication by means of a lexical item instead of a description. (Downing 1977:823)

Marchand (1969) expresses a similar idea in his standard reference on English word formation:

> Many…compounds denote an intimate, permanent relationship between the two significates to the extent that the compound is no longer to be understood as the sum of the constituent elements. A summer-house, for instance, is not merely a house inhabited in summer but a house of a particular style and construction which make it suitable for the warm season only….(Marchand 1969:18)

Unfortunately, two problems arise when the study of lexicalized noun phrases is restricted to noun-noun compounds. First, Downing observed that not all noun-noun compounds express habitual or permanent relationships because many arise spontaneously in conversation and are quickly forgotten. In a celebrated example, she describes a breakfast party with a table that has been set with a different beverage at each plate. As the guests arrive, the hostess directs someone to sit at at what she dubs the *apple juice seat*, thereby coining a noun-noun compound that may never be uttered again because we rarely encounter such oddly configured breakfast tables. In other words, if *apple juice seat* is a word, it's a useless one.

The second problem with the equivalence between lexicalized noun phrases and as noun-noun compounds is that many other noun phrases have the same functional properties. The most often studied are adjective-noun combinations such as *redcoat* and *black market*, but Marchand (1969:80) cites phrasal compounds—or 'lexical phrases', in his terminology—that may be listed in a dictionary. For example, *mother-of-pearl*, a noun phrase with an embedded prepositional phrase, is a rainbow-colored material that forms the lining of some seashells; and *kiss me under the garden gate* and *love lies a-bleeding*, which are complete sentences, are also the names of two flowers in an old-fashioned garden. Even without the complexities introduced by Marchand's exotic examples, the analytical task of identifying the adjective-noun phrases that should be included in a study of lexicalized noun phrases is elusive. Though some are obviously lexicalized because they appear in dictionaries and linguistic studies of word formation, most are not, and it is difficult to specify formal criteria that distinguish the two cases.

Levi (1978) offers the most comprehensive and insightful treatment of this issue, and I adopt many details of her analysis in the computational account of lexicalized noun phrases that I describe in Chapters 3 and 4. First, she performs the valuable service of distinguishing lexicalized noun phrases from linguistically interesting noise. Linguistics textbooks describe a hallmark behavior of native speakers of human language that now seems commonplace but was revolutionary when Noam Chomsky wrote about it in the 1960s (eg., Chomsky 1963). As he argued, when children master the syntax of their native language, they can create sentences that have never been said before, and the same competence enables other native speakers to understand them. This remarkable skill is exercised unconsciously in the everyday acts of having conversations, attending work and school, writing email, and reading the newspaper. Indeed, most of the thousands of sentences and smaller expressions that we encounter in a single day are novel, and as a matter of course, we process this language to extract its content and quickly forget its literal form. At this point in my dissertation, what reader remembers that I have used the phrases *unremarkable observation*, *important features*, *excellent review*, or *collections of machine-readable text*? Levi calls these reflexes of linguistic creativity 'syntactic phrases' to distinguish them from the small subset of noun phrases that persist in memory because they name an object or concept of lasting significance.

Levi refers to the noun phrases that have been lexicalized as 'complex nominals,' a term that I do not adopt here because I believe it is opaque and confusing. Though her analysis is consistent with Downing's analysis of noun-noun compounds, Levi extends the scope of the study by identifying criteria for classifying some adjective-noun phrases as lexicalized. On the one hand, she argues that some adjective-noun phrases must be classed as lexicalized because they occupy the same syntactic position as nouns in phrases that are synonymous or parallel in meaning. For example, *corporate lawyer* and *tax lawyer*, or *parental prerogativ*e and *student prerogative*. It's possible to supplement her examples with pairs of technical terms such as *city planning* and *urban planning*, or *civil engineering* and *highway engineering*. In some cases, a noun-noun compound is essentially synonymous with

an adjective-noun phrase whose modifier has undergone a a word-formation rule that transforms a noun to an adjective, as in *atom bomb* and *atomic bomb*, or *linguistic difficulties* and *language difficulties*.

On the other hand, Levi argues that some adjectives are more likely to form syntactic, not lexicalized noun phrases. For example, adjectives with an adverbial meaning, especially time-denoting adjectives such as *future*, *occasional*, *eventual* and *potential*, are more likely to characterize syntactic phrases, as in *potential enemy, former roommate*, *future dependents*, and *occasional visitors*. Levi also observed that degree adverbials such as *very* can be used as a reliable test to distinguish syntactic from lexicalized noun phrases. Syntactic phrases can be modified by *very*, as in *very destructive riots, very extensive injury, very efficient conductor*—Levi's examples; or phrases *very unremarkable observation*, *very important features*, or *very excellent review*—my examples. But lexicalized noun phrases that have been modified by *very* are, in her terms, ungrammatical, as the starred examples illustrate: **very urban riots*, **very bodily injury, *very civil engineering*, **a very solar system* and **a very atomic bomb*. Similar observations can be made about adverbs that end in *–ly*. Speakers and writers may discuss *extremely important features* or *quietly efficient conductors*, but not **exceptionally civil engineering* or **strangely atomic bombs*.

Levy describes another useful linguistic test for distinguishing adjectives that can form lexicalized noun phrases from those that usually do not, which is more subtle than the previous tests and hinges on the difference between attributive and predicative adjectives. In English, both classes of adjectives may appear before the noun, as in *important features* or *solar system*, but predicative adjectives may also surface in a predicate, usually introduced by a form of the verb *to be*. Levy argues that only attributive adjectives form lexicalized noun phrases. Writers write about *solar systems*, but not **systems that are solar*; *civil engineering*, but not **engineering that is civil*; and *urban riots*, but not **riots that are urban*. On the other hand, the syntactic locations of *beautiful*, *logical*, *efficient* and *important* are left to the discretion of the speaker. On different occasions, we may read or hear about

beautiful princesses, or *princesses who are beautiful*; *efficient conductors*, or *conductors who are efficient*; *conclusions that are logical*, or *logical conclusions*; and *important features*, or *features that are important*.

Unfortunately, as Levi herself observes with lament, all of the linguistic tests that distinguish syntactic from lexicalized noun phrases have exceptions. For example, the phrase *very important person* is so frozen that is is often represented as the acronym *VIP*. The same can be said about *frequently asked questions*, or *FAQs*, the lists of elementary questions that typically introduce well-organized Web sites about specialized or technical topics. Lexicalized noun phrases can also be formed with time-denoting adjectives, as in the physicist's concept of *potential energy*; or with predicative adjectives, as in *high school* or *artificial intelligence*. Because syntactic tests may be a little slippery, we need to go to a deeper level of linguistic analysis to look for the criterion that distinguishes the two classes of noun phrases.

1.2.2. Lexicalized noun phrases and compositional semantics

Near the beginning of the previous section, I cited a quote by Marchand, who observed that lexicalized phrases denote a 'permanent relationship between the two significates to the extent that the compound is no longer to be understood as the sum of the constituent elements.' I discussed permanence in the earlier context, but it is also important to understand the final part of Marchand's quote because it describes an essential difference between syntactic and lexicalized phrases. To comprehend a simple sentence such as *John runs* we add the meaning of *John* to the meaning of *runs* to obtain a composite meaning. The same process applies to a smaller phrase such as *logical conclusions*. The class of concepts denoted by the word *conclusions* intersects with the class of concepts deemed *logical*, and the meaning of *logical conclusions* resides at that juncture. In both cases, explicit rules can be defined for deriving the meaning of the larger expressions from the words that constitute them, which are formalized by linguists who specialize in compositional semantics. By contrast, the import of *summer house*, *garbage man*, *recurrent erosion*, *solar system* and the other lexicalized noun phrases I have mentioned so far cannot be identified by compositional semantic rules. We can guess that a garbage man has something to do

with garbage, but the expression doesn't yield any clue that this is the established name of a profession. Instead, lexicalized phrases are simply defined somewhere in our experience, or we don't truly understand what these expressions mean.

If we conclude that syntactic noun phrases are interpreted by compositional semantic rules, while lexicalized noun phrases always have idiosyncratic meaning, we can draw a distinction that accounts for the fundamental difference in their use. It explains why we can understand sentences that we've never heard before. Though it is logically impossible to store an infinite number of sentences, as Chomsky argued, we can store a finite number of rules for unpacking their meaning. It also explains why lexicalized phrases are part of our vocabulary. At any given moment, there are a relatively small, finite number of lexicalized phrases, many of which are defined or listed in dictionaries, thesauri and encyclopedias or indexes. We consult these resources in some literal or metaphorical fashion whenever we produce or comprehend them.

The semantic distinction between syntactic and lexicalized noun phrases also permits a deeper insight about the syntactic observations made by Levy and Downing. Thus it is possible to argue that Downing, whose fundamental interest is the psychology and sociology of naming, is correct in restricting her focus to noun-noun compounds because these constructions never have a compositional meaning. If the compound is not destined to become an entry in a dictionary because it has a temporary referent, it is necessary to appeal to the phrase's immediate context of use for interpretation. Downing's *apple juice seat* is a vivid, if concocted, illustration of this problem, but such compounds are also commonplace in news headlines.

For example, a headline from a Reuters news story that appeared on the Web during the week of November 26, 2001 proclaims that 'Artificial heart operation man dies.' As linguists interested in compositional semantics, we have to ask: is the 'man' the doctor or the patient? The four-noun compound offers no help, so the first sentence in the story has to clarify the meaning: 'A man suffering from chronic heart failure died from severe bleeding during surgery to implant a self-contained mechanical heart, doctors said.' Similar comments can be made about noun-noun compounds that have been lexicalized. They are equally opaque, but because such

expressions name useful, recurrent concepts, we understand their meanings because they have been committed to memory. For example, *grocery bags* are flimsy plastic bags that are dispensed in supermarkets to hold a customer's purchases of food and drugs; *shopping bags* are large, sturdy, reusable bags, typically with a logo from a high-status department store; and *paper bags* are either made of paper, or are intended to hold paper. In none of the examples I have just cited is the meaning of the noun-noun compound derived from the application of a regular compositional rule. As Dowty (1979:316) argued in a classic study, the relationship between the words in such examples can be anything that is 'appropriately classificatory'—in other words, the words in these compounds are in no regular, semantically definable relationship at all.

A similar argument has been made about attributive adjectives, which have a special status in Levy's analysis of lexicalized noun phrases. Semanticists observe that predicative adjectives such as *red* are well-behaved because they form noun phrases whose properties intersect: a *red ball* refers to the set of red things that are also balls. But noun phrases formed with attributive adjectives—which, in Levy's analysis, appear exclusively in lexicalized noun phrases—are idiosyncratic in the same way that noun-noun compounds are. As Murphy (1988:536) notes, the interesting thing about the adjective modifiers in compounds such as *lunar rock*, *musical clock* and *corporate lawyer* is that none of them can be defined in terms of a principle, such as set intersection, that could be used to specify an explicit compositional semantic rule. Perhaps such expressions have a common core of meaning, he argues, but it is vague. Noun phrases of the form *lunar X*, *musical X* and *corporate X* have something to do with the moon, music, and corporations but the salient features change depending on which nouns are modified. *Lunar rock* is rock from the moon, while a *lunar landing* is a visit by a spacecraft; a *musical clock* is a clock that plays a tune, while a *musical education* is the formal study of music; and a *corporate lawyer* attends to the legal affairs of a corporation, while *corporate stationery* is writing material embossed with a company's logo.

Though it is possible to generate many more examples like those in the previous paragraphs, some scholars object that the picture is not so chaotic. For example, Pustejovsky (1995), Johnston and Busa (1996) and Copestake and Lascarides (1997) argue that many noun-noun and adjective-noun compounds are generated by rules that assign predictable interpretations. For example, if *race car* is a car used for racing and *steak knife* is a knife used for cutting, perhaps there is a rule of the form 'N_2 is-used-for N_1' that can be invoked to interpret novel compounds such as *shop car*, a car used for shopping. Similarly, *glass door*, *leather shoe* and *gingerbread house* imply the existence of a rule 'N_2 is-made-of N_1.' And *molasses cookie*, *peanut-butter fudge* and *eggplant lasagna* suggest that there is a productive rule 'N_2 has-primary-ingredient N_1.' Such regularities among English nominal compounds obviously exist, and have been extensively documented (Marchand 1969), but their linguistic status is controversial. Are they an essential element of semantic knowledge, as Pustejovsky claims, or are they rules of thumb that ease the burden of creating and interpreting names? As Carroll (1985) claims, we can assign an interpretation to *eggplant lasagna* or *pumpkin lasagna* even if we have never heard these phrases before because they are the output of naming conventions that are commonly used in recipes or restaurant menus. But semanticists would object that such phrases can be interpreted only if we invoke our previous experience in the relevant subject domain. And rules of thumb don't cover all the cases. According to a sign in a Columbus, Ohio bakery, *preacher no-bake cookies* were whipped up quickly one day from a batch of fudge when a homemaker saw the preacher approaching her front door.

1.3. Toward a refined computational procedure

With the survey of relevant issues in theoretical linguistics, I can now return to the central problem of my research. How can we use the insights of linguists who specialize in the study of the lexicon to improve the methods for algorithmically identifying lexicalized noun phrases in stores of machine-readable text? Of course, a pessimistic conclusion is that the problem is now more difficult than it first appeared. A reasonable start is to identify noun phrases and filter the results with a measure of

statistical collocation, as Daille (1996) proposed, but the discussion in the previous section reveals just how unclear this simple proposal turns out to be. The problem is that lexicalized phrases may take any syntactic form, ranging from noun-noun compounds to complete sentences—if we broaden our focus to the 'lexical phrases' that Marchand discussed—so any decision about how to configure a parser must be ad-hoc.

But even if the parser is restricted to recognize only short, simple patterns, such as sequences of adjectives, nouns and prepositions, the output from software programs that attempt to identify lexicalized noun phrases in unrestricted text is not directly usable. As a result, some researchers—for example, Bourigault (1992)—report that they hand over the lists of noun phrases generated by their software to professional terminologists for final checking. Others, including Xhai (1997) and Lin (1999), appeal to semantic compositionality to isolate the noun phrases that are lexicalized, but they propose algorithms that do not encode a sophisticated grasp of theoretical issues. For example, Xhai (1997:3) argues that *stock market* is not, in his terms a 'lexical atom,' because the meaning can be derived from compositional semantic rules. As he says, 'both *stock* and *market* carry their regular meanings in contributing to the meaning of the whole phrase *stock market*.' On the other hand, Xhai claims that *white house* is a lexical atom because, to the uniniatiated, the phrase conceals its reference to the mansion where the president of the United States lives. Yet semanticists would argue that *stock market* is lexicalized, too, because it is a noun-noun compound that refers to a persistent concept in American culture. Thus, an algorithm for recognizing lexicalized noun phrases that distinguishes between *white house* and *stock market* performs an uninteresting task.

Fortunately, another interpretation of the issues I have presented in this chapter is far more positive. Despite the fact that I have considered the distinction between syntactic and lexicalized noun phrases from the perspectives of several academic disciplines and potential real-world applications, a surprisingly coherent picture emerges.

For one thing, several lines of evidence suggest that we can productively focus our attention on noun-noun compounds and a subset of adjective-noun compounds. Such noun phrases can be extracted from machine-readable text with relatively simple computational tools, aided by diagnostic tests like the ones that Levy and Downing proposed. Though many of these tests work from observable linguistic evidence such as adverbial modification, they point to the defining semantic distinction between syntactic and lexicalized noun phrases that is not directly observable because it encodes a relationship between language and the world. The same short, syntactically simple noun phrases also usually score high on measures of statistical collocation and get noticed by linguists, lexicographers, highly educated experts in arcane subjects, and ordinary people with special interests. I believe that much more can be learned from this simple observation. The computational study reported in Chapters 3 and 4 of this dissertation describes a systematic way of collecting this knowledge from coherent text by starting with some of Levy's diagnostic tests and supplementing them with empirical evidence that can only be obtained from a large corpus.

Results from the information-retrieval task also fit into this coherent picture. The distinction between syntactic and lexicalized noun phrases confronts us every time we issue a query to a Web search engine. If we are successful, our queries consist of noun phrases that represent the names of significant, persistent concepts in a domain of interest. In other words, such queries represent the upper bound on the literal sequences of text that users of search engines have to second-guess when they try to satisfy their information needs from unseen documents. When we issue a query that exceeds this bound, the result is either nothing at all, or the idiosyncratic expression of a single writer. For example, if I consult Yahoo with the query 'In the long journey out of the self, there are many detours, washed-out interrupted raw places where the shale slides dangerously,' I retrieve one document: the text of Theodore Roethke's poem 'Journey into the Interior.'[3] Only one person has ever assembled these words in exactly this way, and I was able to issue this query because I had to memorize Roethke's poem as a college sophomore.

[3] Accessible at: <http://gawow.com/roethke/poems/187.html>

Though I am not primarily concerned about the performance of search engines, the results of the linguistic inquiry reported here are relevant to this problem in two respects. First, the results can be used to interpret the failures in previous attempts by information-retrieval researchers to identify phrasal queries. If their algorithms identified mostly syntactic noun phrases, it is not surprising that the queries generated from such algorithms were disappointing and even counterproductive. Second, it provides system designers with an improved procedure whose output is a list of noun phrases that act as words and help readers discover the knowledge encoded in a collection of text about an unfamiliar subject.

Because I have several sources of observable evidence—including syntactic tests, the recorded linguistic behavior of language users, and at least one real-world method for double-checking the output—I can make progress on the problem of identifying lexicalized noun phrases without having to tangle with the thornier semantic problems that I discussed in the previous pages. Indeed, the semanticist's inquiry is motivated by a simple question that also produces an observable result: what do we put in the dictionary? Lyons (1976) echoes the view commonly held among linguists and computer scientists that dictionaries are repositories of exceptions, where we list words that must be defined, or store oddball data that can't be processed by the normal rules:

> Now, one way of looking at the dictionary, or lexicon, in relation to the grammatical description of a language is to regard it as a kind of appendix to the grammar—an appendix in which we find, appropriately indexed, all the information that we need to know about particular lexemes or their associated forms and cannot derive from anything else that the grammatical…analysis tells us about them (Lyons 1976: 514).

When phrases are listed here, it's because they have meanings that can't be computed by the standard machinery of compositional semantics.

But professional lexicographers don't always refer to compositional semantics when they make their decisions about what goes in a dictionary. According to the Oxford Advanced Learners' Dictionary (Hornby 1974: 239), a dictionary also has a social function; it is a 'book listing and explaining the words of a language, or the words or topics of a special subject.' The National Information Standards Organization (NISO) specification for the design of machine-readable thesauri gives this advice for the treatment of compound nouns: 'Retain as a single word when the

compound noun has become so familiar in common use that it is considered for practical purposes to represent a single concept,…[as in] *data processing* and *gross domestic product* (NISO 1993).' Similarly, Chan (1995:43), an expert on the subject headings used in libraries for organizing collections of published material, says that 'Single-concept headings appear in the form of single- or multiple-word terms…When a single object or concept cannot be properly expressed by a single noun, a phrase is used. Multiword terms appear in the form of adjectival or prepositional phrases…[and include examples] such as *chemical engineering*, *mathematical statistics*, [and] *earthquake engineering*.'

Nevertheless, what is common in these citations is a point about which semanticists would not disagree: lexicalized noun phrases have unique referents. Whether the referent is characterized in such a way that phrases such as *art pottery*, *data processing* or *earthquake engineering* are theoretically comprehensible when we encounter them for the first time, or whether the unique referents guarantee that all lexicalized phrases are semantically opaque remains a controversial issue that we can sidestep without having to commit ourselves to a semantically unsophisticated analysis. The very fact that such phrases get noticed implies that they are used repeatedly, perhaps because they are the conventional names for important concepts. If so, they are operationally useful in a way that syntactic phrases are not. In the information-retrieval task, the lexicalized noun phrase *art pottery* returns a manageable number of documents about collectible antique ceramics, which implies that the phrase is the established name of a single, persistent referent in a specialized subject, while the syntactic noun phrase *additional activities* returns tens of thousands of documents about nothing in particular because the referent presumably changes from context to context. Lexicographers may have a man-on-the street answer to what goes in the dictionary that is theoretically less imprecise, but it is not inconsistent with the outcome of a semantic analysis.

1.3. An empirical study of lexicalized noun phrases

I have said many times in this discussion that competent speakers and writers display their awareness that some noun phrases may have lexical status. Linguistic

tests encode some of this knowledge because lexical status determines whether a concept is conveyed through a noun-noun compound, or through a noun phrase modified by *very*. But much more evidence can be found in a corpus of coherent text. For example, consider the two paragraphs cited below. The first is from an article that recently appeared in a newspaper in Columbus, Ohio. The second is from a trendy book series published in the 1990s that describes one feature of a popular computer programming language.

> They're called "popcorn fires" – those little nuisance incidents that cause smoke, set off fire alarms and drive college students grumbling from their dormitory rooms. College safety officials know that if they can cut down on popcorn fires, students are more likely to take real fire alarms more seriously.[4]

> You can sum up the big difference between beans on the one hand and Java applets and applications on the other in one word (okay, two words) : *component model*. Chapter 2 contains a nice, thorough discussion of component models (which is a pretty important concept, so I devoted an entire chapter to the subject).[5]

Even without a dictionary, collocation statistics, a sophisticated parser, an understanding of the academic arguments of compositional semantics, or a specialized knowledge of the taxonomy of fires or software development using the Java programming language, we can read these passages and infer that *popcorn fires* and *component models* are lexicalized noun phrases. How? Writers drop hints about the lexical status of the expressions they use, and the more deeply we analyze the text, the more clues we can discover. Only superficial scanning is required to discover that lexicalized noun phrases are often enclosed in quotes or are the objects of fixed expressions such as *is called*, which imply that the expression is not the writer's invention. They may be are printed in nonstandard typefaces, as when *component model* is italicized in the sentence 'You can sum up the big difference between beans on the one hand and Java applets and applications on the other in one word…: *component model*.' A deeper level of analysis requiring some knowledge of syntax reveals that lexicalized phrases are often conjoined with other words or lexicalized phrases. The previously cited example also illustrates this point: the single word *applications* is conjoined with the lexicalized noun phrase *Java applets*,

---

[4] The Columbus Dispatch, Columbus Ohio, January 20, 2000, p. 3A.

[5] *Java Beans for Dummies*. Emily Vander Veer Chicago, IL: IDG Books Worldwide. 1997, p. 14.

both of which are names for important concepts in the domain of Java programming. Additional evidence for the hypothesis that a phrase may be the name of a concept can be obtained with an even deeper level of understanding that requires discourse analysis. In the second sentence of the same quote, *component models* is co-referent with *concept* and *subject*: 'Chapter 2 contains a nice, thorough discussion of component models (which is a pretty important concept, so I devoted an entire chapter to the subject).'

The major goal of the research reported in this dissertation is to analyze local evidence in coherent discourse for the light it can shed on the distinction between syntactic and lexicalized phrases, using computational methods. This study complements both traditions of scholarship discussed in this chapter. Starting from the studies of Downing and Levy, I construct a series of tests for classifying a noun phrase as syntactic or lexicalized, supplementing it with tests that can be constructed when lexicalized noun phrases are observed in the context of coherent discourse. Of course, an important conclusion from the linguistic scholarship is that observable sources of evidence pertaining to the lexical status of a noun phrase are rarely, if ever, categorical or definitional. Nor is the new evidence I consider. Not every phrase printed in italics is lexicalized; nor is every noun phrase in a conjoined list. As Levy (1978:46) asked, 'May we conclude…that there are *no* clear and consistent criteria according to which an entity called the *nominal compound* may be identified?' Despite Levy's negative conclusion, I believe that this evidence can be productively evaulated using an appropriate mathematical model.

The results from this study also complement the evidence supplied by statistical collocations. Statistical collocations, of the sort that Church and Hanks studied, are currently identified using global evidence from a corpus, and the results exhibit variable degrees of overlap with lexical collocations. When local evidence from a corpus of coherent text is also considered, it has the effect of increasing the linguistic input to the decision about which noun phrases are lexicalized. For example, if the entire text of *Java Beans for Dummies* is submitted to measures of lexical collocation, it is likely that *component model* would achieve a relatively high score because it is mentioned so frequently. If so, the local evidence in the quoted

passage would simply increase our confidence that *component model* is lexicalized. This application of local evidence is a strategy that has been successfully applied to other classic problems of lexical knowledge acquisition, such as word sense disambiguation (Ide and Veronis 1998) and the lexical semantics of verbs (Lapata 1999).

Chapters 3 and 4 identify many sources of local evidence and describe the computational study that evaluates them, but the procedure can be outlined here. A list of lexicalized noun phrases obtained from a published thesaurus provides the starting point for a linguistic and a computational analysis. A corpus that has many citations of the entries in the thesaurus is analyzed for the presence of local syntactic and lexical cues like those discussed in this chapter, using efficient computational techniques. This evidence is used to train a classification algorithm to make a binary decision regarding the lexical status of phrases whose status is unknown. For example, if a lexicalized noun phrase candidate is modified by *very*, evidence begins to accumulate that it is a syntactic, not a lexicalized phrase. Conversely, the phrase is eventually categorized as lexicalized if it is the object of *is known as*, or appears in a list of conjuncts that contain words or other lexicalized noun phrases. The result is a confidence measure for each phrase that combines the outcome of the classification algorithm with a collocation score, which is evaluated using human judges.

The corpus for this study is a collection of approximately 150,000 English-language documents in the domain of engineering that were harvested automatically from the Web in 1997-1998 and classified using the Engineering Information Thesaurus (Milstead, et al. 1995), under the auspices of the Engineering Electronic Library Project at Lund University in Sweden, whose goal is to increase the accessibility of specialized collections of Web documents. The results are available in a searchable and browsable Web interface accessible at <http://eels.lub.lu.se/ae/>. The length of the documents, discourse style and content is highly variable, but much of it consists of academic papers and home pages for university departments, commercial engineering services, and scientific institutes. The harvesting process preserves as much of the text as possible while eliminating some common problems with Web documents, making it appropriate for linguistic analysis, but it is

insufficient for reproducing the 'Web experience' of the original documents because embedded computer software code and references to graphics are eliminated. Since the corpus was automatically obtained, the quality of the text varies, as does the subject matter.

Though a corpus like this presents analytical difficulties because it has not been subjected to editorial processes that ensure quality and stylistic consistency, it has several compelling advantages for studies like mine. First, it is rich in lexicalized noun phrases. Levi's theory of complex nominals already gives prominence to the subtleties in the deceptively ordinary-looking phrase *electrical engineering* and *solar generator*, but when I extend it in Chapters 3 and 4, it must account for many more phrases from this domain, such as *air traffic control*, *applied mathematics*, *chemical agents*, *heat transfer*, and *strength of building materials*. Second, the Engineering Information Thesaurus is rich with terminology that engineers actually use and is cited in the corpus, which provides a starting point and a set of correct answers for seeding my analysis. Finally, this corpus, with approximately 500 megabytes of usable text, is very large by current standards. A large corpus is necessary for studying lexical issues because words, especially lexicalized noun phrases, are sparsely distributed.

The immediate outcome of this research is a falsifiable method to identify lexicalized noun phrases that can be viewed as a testbed for evaluating linguistic tests like those found in classic studies such as Levi's, and possibly supplementing them. Unlike previous attempts to solve this problem, it doesn't require judgments about the compositionality of the words in the phrase or human operators who filter the output. An important side effect of this work is that the sources of evidence can be ranked by the classification algorithm, which promises to lead to computationally cheaper methods for the linguistic objects of interest.

This research also has theoretical implications. The most fundamental is a theory of common usage, which, as Abney (1996) argues, is one of the major contributions that statistics can make to linguistic analysis. I believe that a theory of common usage is already implicit in the linguistic scholarship on nominal compounding but it isn't fully grounded. Without such a theory, too much

importance may be attributed to rare or unusual data. Perhaps *eggplant* has been erroneously identified as the textbook example of the lexicalized nominal compound, with its semantically opaque roots, and orthographic representation as a single token. But lexicalized phrases that fail these classic linguistic tests may be far more common.

1.5. The organization of this dissertation

The rest of this dissertation is organized as follows. Chapter 2 is a technical review of the computational tools required to recognize noun phrases in coherent text, to identify those that might have terminological status, and to assign an internal structure to them. Chapter 3 describes the distribution and behavior of lexicalized and syntactic phrases in the corpus of engineering documents and develops a theory of the linguistic limits of such phrases. Chapter 4 presents the results of a machine-learning study that uses the local evidence obtained from a corpus to classify a noun phrase as lexical or syntactic. Chapter 5 considers how the analysis can be extended to new sources of linguistic evidence.

CHAPTER 2

ALGORITHMS FOR EXTRACTING NOUN PHRASES FROM TEXT

2.0. Introduction

This chapter reviews the technical infrastructure required to recognize noun phrases in coherent text, to identify those that might have terminological status, and to assign an internal structure to them. The practical goal of a system developed from these tools is to discover data that would shed light on the philosophical and psychological issues regarding lexicalized noun phrases discussed in Chapter 1 by automating some of the methods for collecting them. The material in this chapter is mostly review because it describes the kernel of a system that can be constructed from the results of previous research. I will use this system to generate noun phrases that will be evaluated more rigorously in the study described in Chapters 3 and 4. To achieve further grounding in the central problem of this dissertation, I also take a first look at the engineering corpus that I described in Chapter 1 and supplies most of the data for my investigation.

Together, Chapters 1 and 2 constitute a tutorial on the rich problems presented by lexicalized noun phrases as they are addressed by several communities of scholars: theoretical linguistics, philosophy, lexicography, information retrieval, and several sub-disciplines of computational linguistics. Given that lexicalized phrases reside at the boundary between syntax and the lexicon, encode clues about the significant concepts defined by a language community, are important for the proper functioning of natural language processing systems, and show promise of being discovered without incurring great computational expense, it should not surprising that they have

been studied from so many perspectives. Computationally cheap methods for the automatic identification of noun phrases make it feasible to work on large stores of text, such as the 500-megabyte corpus in my study. Such methods have been investigated since the early 1980s, first in the information retrieval community and later in the computational linguistics community at large. Interest in these methods has been driven at least partially by the hunch that important lexical information can be extracted without incurring the computational overhead of a parser that attempts to assign structure to complete sentences. Nearly twenty years of work on the problem has produced general agreement that the task has four parts, which can be executed in a linear sequence: part-of-speech tagging, noun-phrase identification, internal structure assignment, and an optional filter that is used to distinguish a small number persistent and therefore potentially lexicalized phrases from the far more common syntactic phrases.

2.1. Four components in a system for recognizing lexicalized noun phrases

2.1.1. Part-of-speech tagging

One of the first big successes in the statistical processing of machine-readable text corpora was Church's (1988) stochastic part-of-speech tagger. Patterned after successes in speech recognition, the Church tagger solved the problem of identifying the parts of speech of tokens in text by consulting a hidden Markov model. A hidden Markov model is essentially a finite state machine augmented with two sets of probabilities: transition probabilities assigned to the arcs and emission probabilities assigned to the nodes. A Markov model is said to be hidden if we don't know which path was taken through the finite-state machine to reach the final state. Markov models solve problems by predicting the most likely current state based on the contents of the adjacent prior context. The size of the context can vary but there is a tradeoff between the accuracy in performing the task and the expense of maintaining and traversing potentially large tables of transitional probabilities. In practical terms, Markov models that are created to solve problems in corpus linguistics are usually

limited to one or two words of prior context—the so-called bigram or trigram models. Charniak (1996) contains a useful tutorial on the linguistic applications of hidden Markov models.

Using a bigram model for simplicity, we may want to know the part-of-speech assignments for the words in the sentence *Time flies like an arrow*. Since this sentence is structurally ambiguous, it has the two plausible tag assignments shown in Figure 2.1. The first structure assignment identifies the usual declarative interpretation of this sentence, a cliché about the quick passage of time; while the second assignment identifies the grammatical structure of a bizarre imperative to measure the speed of flies using an arrow as a guide. The tags are the same ones that were used to annotate the Brown Corpus (Kucera and Francis 1967) and have been widely adopted in the computational linguistics community.

```
# NN   VBZ  IN   DT   NN
  Time flies like an   arrow

# VB    NNS IN    DT   NN
  Time flies like an   arrow
```

Legend:

DT	Determiner
IN	unspecified part of speech
NN	Uninflected noun
NNS	Plural noun
VB	Uninflected verb
VBZ	3rd-person singular verb
#	Sentence boundary

Figure 2.1 Two part-of-speech (POS) tag assignments for a structurally ambiguous sentence

In Church's algorithm, part-of-speech tags can be assigned using two sources of information: the tag assigned to the prior word and the probability that the current word is assigned a given tag. Probabilities for both of these values are estimated from raw frequencies obtained from a corpus that has been tagged by hand. Sample probabilities for the sentences in Figure 2.1 are shown in Figures 2.2 and 2.3. The equations cited in these figures simply compute, for each sentence structure, a product of probabilies using Church's two sources of information: the probability that *time* is a noun, that *flies* is an inflected verb, the probability that the word immediately to the

right of a sentence boundary is a noun, and so on. The algorithm eventually selects the tag assignments in Figure 2.2 because, in a typical corpus, *time* is almost always a noun. And the token *flies* is tagged as an inflected verb because *flies* is usually an inflected verb, especially when it is immediately preceded by a noun. Of course, this discussion is a simplification because the part-of-speech assignments in Figures 2.2 and 2.3 are only two of the many logically possibly ones that would be considered in an execution of a hidden Markov model.

```
# NN   VBZ  IN   DT   NN
Time   flies  like an  arrow

P(NN|#)P(V|N)P(IN|VBZ)P(D|IN)P(NN|DT) x
P(N|time)P(VBZ|flies)P(IN|like)P(an|DT)P(n|arrow)
```

Figure 2.2 POS tag assignments for the declarative reading of *Time flies like an arrow*

```
# VB   NNS  IN   DT   NN
Time   flies  like an  arrow

P(VB|#)P(NNS|VB)P(IN|NNS)P(D|IN)P(NN|DT) x
P(VB|time)P(NNS|flies)P(IN|like)P(an|DT)P(n|arrow)
```

Figure 2.3 POS tag assignments for the imperative reading of *Time flies like an arrow*

If, counter to expectation, the structure assignment in Figure 2.3 turns out to be the correct answer, as it might be in a literary work or in a corpus of linguistics textbooks about structural ambiguity, a stochastic tagger could arrive at this result in one of two ways. First, the tagger could learn from hand-tagged input containing a large number of sentences like Figure 2.3. Second, it could make a second pass through the data and repair mistakes by learning from rules that are peculiar to the current data. This is the strategy of the tagger developed by Brill (1995), which is freely available and widely used in the research community.

Stochastic taggers that encode the algorithms described in this section achieve accuracy rates approximating 95%-98%, leading many researchers to conclude that part-of-speech tagging is a mature technology. I adopt Brill's part-of-speech tagger in my system for identifying noun phrases.

2.1.2. Syntactic parsing

A parser assigns syntactic structure to a well-formed sentence that has usually been annotated with part-of-speech tags. The construction of sentence parsers has been a fruitful research enterprise among linguists and computer scientists since the late 1970s, but here I will discuss the arguably far easier problem of partial parsing. Partial parsing is a reasonable technical solution in a natural language processing system when the objects of study are short phrases in large collections of text because such expressions can be quickly located and analyzed with a reasonable degree of accuracy. As a result, partial parsers are commonly used in applications whose goal is to extract isolated words, proper names, noun phrases or verb phrases from large stores of machine-readable text—as in information-retrieval systems, terminology extractors, and in many machine- translation systems. In an excellent survey of the technical approaches to partial parsing, Abney (1996) states that, regardless of how a partial parser is implemented, the goal is to scan a sentence for an 'island of reliability,' which presumably contains the linguistic unit of interest and has the necessary context from which the rudiments of an accurate structure can be assigned.

Partial parsers vary considerably in the linguistic naturalness with which they carry out their tasks, a point that can be illustrated by considering two processes for identifying the noun phrases in a structurally ambiguous sentence such as *She gave her dog biscuits*. On the one hand, the Fiddich parser (Hindle 1994), which encodes a high degree of linguistic naturalness, would attempt to assign a complete structure to the sentence. It would have no serious trouble until it encounters the verb-argument ambiguity of *dog biscuits*, which forces human readers to wonder whether the woman gave biscuits to her dog, or whether she give dog biscuits to an unspecified female. To solve the problem, the parser isolates the offending constituent and assigns a structure to the rest of the sentence. The unparsed subtree is assigned a structure with

a finite state automaton that is dedicated to the analysis of such problems and the results are attached to the sentence. In this example, the Fiddich parser would correctly report that the sentence is structurally ambiguous.

On the other hand, a partial parser such as FASIT (Dillon and McDonald 1983), which has been widely employed in the information-retrieval community, uses templates to identify the linguistic structures of interest. To assign a structure to the sentence *She gave her dog biscuits*, the FASIT parser would scan the part-of-speech-tagged text, looking for nouns. When a noun is located, FASIT would examine a small window of left and right context in an attempt to match the part-of-speech tags of the tokens to a set of templates that specify the structure of a simplex English noun phrase—i.e., a noun phrase with a simple linear structure and no embedded clauses, such as *the man who came in from the cold*. Since *her dog biscuits* is a legal noun phrase in English and FASIT is optimized to identify the longest sequence of tagged tokens that match a template, the structural ambiguity of the sentence is overlooked. Essentially the same result can be obtained with an even simpler parser that employs regular expressions such as (NN|NNS|NNP|NNPS|JJ|DET|)*(NN|NNS|NNP|NNPS|VBG), a strategy explored by Turney (1997) in his evaluation of algorithms for the identification of noun phrases in unrestricted text. This pattern identifies arbitrary sequences of adjectives, nouns, prepositions, and determiners and relies on input that is syntactically correct, for the most part, to compensate for its obviously deficient phrase structure specification.

Sophisticated symbolic partial parsers such as Fiddich and the simple pattern-matching parsers have tradeoffs if the goal is to identify lexicalized noun phrases. As I suggested in Chapter 1, long or complex noun phrases are rarely, if ever, lexicalized, so the advantage that the Fiddich parser would have over the simpler solutions in identifying these structures is not compelling because it is also computationally far more expensive. Though the Fiddich parser is better equipped to handle structural ambiguities, corpus evidence can be used to resolve problems that result from parsing failures. If *She gave her dog biscuits* is embedded in a large corpus that contains other citations of *dog biscuits*, the template-matching parser extracts useful information from this sentence, despite a possibly inappropriate structure assignment.

Because of these issues, I have adopted a template-matching parser whose design is similar to FASIT. Like FASIT, the parser does its essential work by scanning text that has been processed by the Brill part-of-speech tagger, looking for simple linear patterns of tags that represent simplex noun phrases. To eliminate some computational overhead and potential sources of error, the input text is first chunked by heuristics that grossly identify noun phrase boundaries. For example, punctuation marks are reliable end boundaries; and determiners and quantifiers are reliable start boundaries. Since the boundaries are eliminated from the parsed output, the effect of the chunker is to normalize the noun phrases to something that looks like a dictionary citation form, such as *clean energy technologies*, which facilitates the tabulation of the citations *clean energy technologies*, *these clean energy technologies* and *some clean energy technologies* as instances of the same phrase.

If the goal is to examine the full scope of lexical phrases, which Marchand (1969) defined and I discussed in Section 1.2.1 of Chapter 1, the list of patterns would include 'N-and-N,' which would permit the recognition of *bread and butter*; or 'N-of-N,' which would admit *mother-of-pearl*. Otherwise, this list is restricted to patterns of adjectives and nouns. The patterns are stored in a text file that can be easily modified; a fragment is shown in Figure 2.4. The last pattern in illustrates a common problem. Though the Brill part-of-speech tagger can distinguish between nominal (NNG) and verbal (VBG) gerunds if it is appropriately trained and given sufficient context, the tagging of gerunds remains a difficult and error-prone process. If patterns with VBG are excluded from further analysis, many valid noun-phrase candidates would be dropped, but some verb phrases would be admitted if VBG is listed as a valid tag. As with all decisions about which patterns to include in the template file, the VBG tag introduces a nuisance that may be effectively handled in a later processing step.

Pattern	Example
NN NN	resource management
NNS IN NN IN NNS	rules for certification of airmen
JJ NNS NNS	human factors resources
JJ CC JJ NNS	molecular and mesoscopic structures
JJ JJ NN NN	electronic computational chemistry conference
VBG NN	diving community

Figure 2.4 Patterns of noun phrases recognized by a template matcher

My goal in constructing the list of patterns is to maximize the number of simplex noun phrases that can be recognized by the parser, using a part-o-speech-tagged sample from the engineering corpus as a guide. Because a template-matching parser is not sophisticated enough to encode recursion without introducing problems of overgeneration, the list of patterns is a simple enumeration.

2.1.3. Identifying lexicalized noun phrases using statistical filters

In Chapter 1, I suggested that syntactic parsing cannot guarantee the discovery of lexical knowledge and must be supplemented with a concept of persistence that models the lexicographer's concept of a collocation. Church and Hanks (1990) argued that linguistically interesting collocations consist of pairs of words that are not merely frequent in a given corpus but also highly associated with one another. Lexicalized noun phrases—such as *civil engineering*, *stock market*, and *white house*, which I cited in Chapter 1—often achieve high scores on statistical measures of collocation when they are embedded in a large corpus of coherent English text. But sequences of words that are merely frequent, such as the recurring syntactic patterns such as *of the* or *in a*, do not usually correspond to names of concepts and are, by most statistical measures, not highly associated, as I will show below.

2.1.3.1. Measures of association

Statistical measures of association commonly used in the computational linguistics community are based on values obtained from contingency tables. The simplest contingency tables summarize the relationship between two variables *a* and *b* and can be used to test the hypothesis that they are linked, dependent, or associated. Table 2.1 shows a simple contingency table for two variables, L_i and L_j. The tilde (~) signifies the absence of the variable.

	L_j	$\sim L_j$
L_i	*a*	*c*
$\sim L_i$	b	d

Legend:
a. the frequency of pairs involving both L_i and L_i
b. the frequency of pairs involving L_i, but not L_j
c. the frequency of pairs involving L_j, but not L_i
d. the frequency of pairs involving neither L_i nor L_j

Table 2.1 A contingency table for two variables

The relationships in in Table 2.1 can be made concrete by the data in Tables 2.2 and 2.3, which show contingency tables with frequency counts for two bigrams: one with a high association value but a relatively low frequency, *steel scrap*; and one with a low association value and a high frequency, *this file*. *Steel scrap* is a potential lexicalized noun phrase that could be listed in an index; *this file* is an ordinary syntactic phrase.

	L_j	$\sim L_j$
L_i	7	19
$\sim L_i$	1	842571

Legend:
L_i = steel
L_j = scrap

Table 2.2 A contingency table for *steel scrap*

	L_j	$\sim L_j$
L_i	47	2915
$\sim L_i$	637	842524

Legend:
L_i = this
L_j = file

Table 2.3 A contingency table for *this file*

The values in Tables 2.2 and 2.3 are obtained directly from the corpus and are used to estimate the probability that the two word sequences are associated. For *steel scrap*, L_iL_j, or 7, is the joint frequency of *steel* and *scrap*; $L_i\sim L_j$, or 19, is the count of *steel* in bigrams other than *steel scrap*; $\sim L_iL_j$, or 1, is the count of *scrap* in bigrams other than *steel scrap*; and $\sim L_i\sim L_j$, or 842571, is the frequency of bigrams containing neither *steel* nor *scrap*. Even without an association measure, it is obvious from this data that there is a strong dependency between *steel* and *scrap* and a weaker one between *this* and *file*. *Scrap* appears apart from *steel* only once, while *file* appears in contexts other than *this* 637 times and the bigram frequency of *this* apart from *file* is even larger.

Statistics such as chi-square, mutual information, and log-likelihood use the values from the contingency table in different ways to compute an association measure. However, mutual information, the measure introduced to the computational linguistics community by Church and Hanks (1990), is perhaps intuitively the most straightforward. Without normalization, it is simply the ratio of the value (a) in Table

2.1 to the product of (a+b) and (a+c), or the ratio of the joint frequency of bigrams such as *steel scrap* to the product of the total frequencies of *steel* and *scrap* in the corpus. Standard elementary statistics textbooks (for example, Hays 1981) explain how to compute the chi-square measure from contingency tables. But an important paper by Dunning (1993) warns that chi-square and mutual information assume normal distributions, and words in a corpus of coherent text are not normally distributed. When words are tabulated and plotted against their rank order, their distribution forms a hyperbolic curve like that in Figure 2.5, as Zipf (1949) first observed.

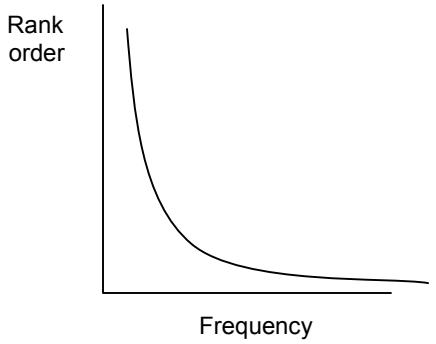

Figure 2.5 A hypothetical Zipf curve

The Zipf curve illustrates an important generalization about the distribution of words in coherent text that is true regardless of language or subject matter: only a few words are used frequently, while most words are used once or infrequently. Thus if statistics assuming a normal distribution are used to compute associations, the importance of the rare words in the long tail of the Zipf curve is overestimated. In other words, they are given more importance than they deserve. According to Dunning's critique, we have to be careful in our interpretation of the lexical status of *steel scrap*, given the data I have presented. It might be an artifact because Table 2.1 shows that the absolute frequency in the corpus is suspiciously low. To avoid this problem, recommends using the log-likelihood statistic, which is robust even on data that is not normally distributed, assigning a score that can be more confidently interpreted as a measure of lexical collocation. If *steel scrap* should achieve a high log-likelihood

score, we can be more confident of its status as a lexicalized noun phrase. Daille (1996) gives useful formulas for computing mutual information and log-likelihood from values in a contingency table.[6]

2.3.1.2. Using measures of association to identify noun phrases

How can measures of statistical association be used in a software program that automatically identifies lexicalized noun phrases? Justeson and Katz (1995) propose a system that first applies the statistical filter to the raw text, identifying a list of highly associated bigrams such as *best practice*, *he said*, and *linear regression*. The parser then filters all but the last item because it is the only phrase in the list that exhibits the noun-noun part-of-speech pattern that reliably identifies multi-word terms in technical texts. However, Daille (1996) argued that fewer legitimate terms are lost if the sequence of the two processes is reversed. In her model, the parser extracts noun phrases from the text and the collocation filter identifies from this output the much smaller subset of those that have been lexicalized.

Unfortunately, this account glosses over a technical detail that must be addressed before the filter can be used in a realistic application. As the examples I have cited so far imply, statistical measures of association are defined only for pairs of observations. But noun phrases that occur naturally in coherent text can be arbitrarily long, and the procedures I have described so far cannot be used to identify lexicalized noun phrases such as *biomedical engineering department*, *air safety information resource* and *standard generalized markup language*.

One solution is to complicate the mathematical model by extending the contingency table. For example, to identify the associative strength of the words in trigrams such as *biomedical engineering department*, the contingency table would

[6] Daille's computing formulas for the two statistics discussed here are as follows. Using the values a,b,c,d in the contingency table given in Table 1a:

Mutual information (Church and Hanks 1990) = $\log_2 (a/(a+b)(a+c))$

Log-likelihood (Dunning 1993) = $a\log_a+b\log_b+c\log_c+d\log_d$
$-(a+b)\log(a+b)–(a+c)\log(a+c)$
$-(b+d)\log(b+d)–(c+d)\log(c+d)$
$+(a+b+c+d)\log(a+b+c+d)$

have 2^3 entries. This is difficult to depict graphically because the table is a cube, but for expedience, it can be represented as a traditional 2-by-2 table with the third variable completely nested in the other two. Table 2.4 shows an example.

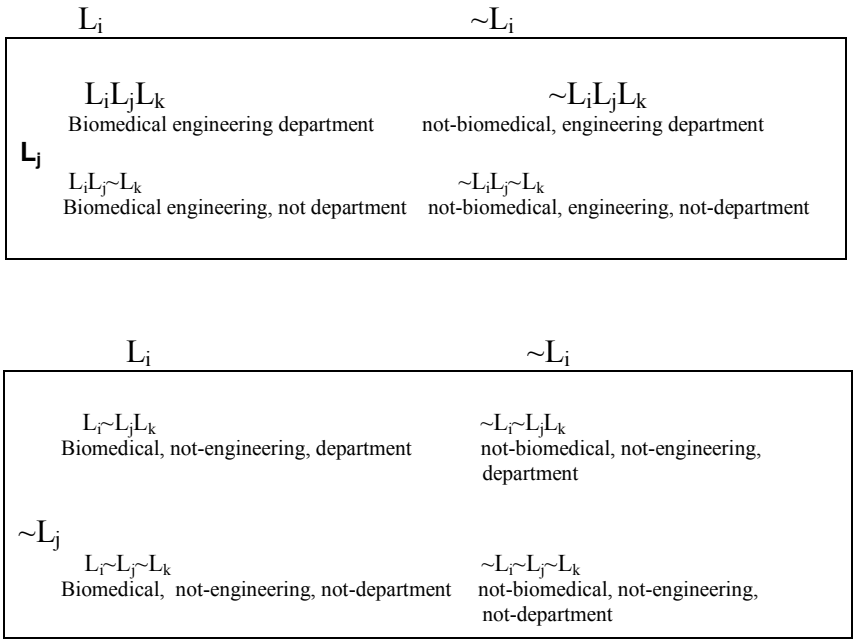

Table 2.4 An extended contingency table

As in the simpler contingency table with two variables, the values for the eight cells in Table 2.4 are probabilities that are estimated from the frequencies observed in the corpus. Since many of these cells will have very small probabilities, it is tempting to remove them from the model, which is perhaps one of the motivations for a commonly used shortcut that makes multiple passes through the data, creating bigrams at each step. On the first pass, highly associated bigrams such as *biomedical engineering* are identified. On the next pass, this bigram is treated as a single term and the bigram *biomedical-engineering department* is identified—a process that

continues until an expedient upper limit is reached. Zhou and Dapkus (1997) use this heuristic to identify potentially lexicalized noun phrases in a large corpus of news text.

For both solutions, a pass through the data is required for each increment in the size of the word sequence. Though this is computationally expensive in a large corpus, a worse problem is that the natural cutoff in the data is difficult to identify without introducing even more complexity. For example, the iterations may stop when certain lower-bound association thresholds are no longer reached, as should happen when the lexicalized phrase *biomedical engineering* is paired with a syntactic boundary such as *is*. But since reasonable thresholds may be difficult to specify, a common alternative solution is to iterate a given number of times—say, 6 or 7—using the common-sense assumption that simplex English noun phrases rarely exceed this length. Nevertheless, the cutoff is arbitrary with respect to the corpus, and it introduces problems for filter-first models such as the one proposed by Justeson and Katz because it creates false truncations. For example, a filter-first system might produce *international journal of plant* from a corpus of biology text. This is a syntactically valid noun phrase, but if the real collocation is *international journal of plant pathology*, the phrase is truncated too soon if the corpus accidentally produces a low association value for *plant pathology*.

Parse-first models such as Daille's avoid this problem, but they must still select the lexicalized noun phrases from a list ranked by the value assigned from the association measure. In this model, some genuine lexicalized phrases are missed, while spurious phrases that achieve a high score are erroneously selected. Table 2.5 illustrates the problem in bigram data from the engineering corpus. Though most of the phrases that receive high scores are probably lexicalized, *economically feasible* is probably not. Nevertheless, it is ranked higher than *health service*, which probably is lexicalized.

Bigrams with high associations		Bigrams with low associations	
sri lanka	11.07	which contains	2.87
saudi arabia	10.76	these letters	2.85
zonal wind	10.75	procedures for	2.84
avant garde	10.64	this file	2.83
steel scrap	10.25	search contents	2.82
hazardous waste	10.02	health service	2.81
fossil fuels	9.91	arranged by	2.78
economically feasible	9.73	publications data	2.77

Table 2.5 Bigrams with high and low log-likelihood values

Because of the issues considered in this section, I adopt a parse-first model. A filter that identifies lexicalized phrases based on the log-likelihood statistic is eventually applied to the data, but only after considering the cues in the linguistic context found in the corpus that bear on the decision to identify a noun phrase as lexical or syntactic. My hypothesis is that these cues go a long way toward compensating for the errors that result when an association statistic is the sole input to the decision. Additional linguistic analysis is required to identify these cues, and computer software must be developed to identify them automatically. These topics consume most of Chapters 3 and 4.

I also follow the practice commonly adopted by computational linguists and apply the log-likelihood statistic as a utility function. If two contiguous words are highly associated, the result can often be interpreted as linguistically interesting lexical knowledge. Other pairwise associations between linguistic objects may not have this narrow interpretation, but they are nevertheless important for system-building. An example is developed in the next section.

2.1.4. The assignment of internal structure

When applied to a large corpus, the processes I have described so far produce many noun phrases that consist of three or more tokens. Should *information retrieval experiment* be analyzed as [[information retrieval] experiment] or [information [retrieval experiment]]? Or should *aerospace engineering department* be analyzed as [[aerospace engineering] department], or as [aerospace [engineering department]]? Noun phrases that have adjective modification may also have complex internal

structure and are common in collections of technical text, such *electrical engineering department*, *amyotrophic lateral sclerosis, thermal properties measurement*, and *tactical communications protocol*.

An algorithm that assigns internal structure to such noun phrases satisfies the need for completeness in a research program devoted to the extraction of terminology from text, but it also has real uses. For example, an accurate internal structure is required for the construction of correctly formed hierarchical listings of noun phrases that are typically found in indexes or taxonomies. An example is shown in Figure 2.6. The index in the left column is constructed from correctly parsed phrases such as [microwave [background radiation]], while the one on the right propagates the mistake of identifying *ban treaty* as a phrase that can exist independently. The correct parses of the noun phrases on the right, such as [[[antiballistic missile] ban] treaty], reveal that *ban treaty* is an odd phrase because it requires a left modifier and is split by a constituent break. Neither problem is observed in the internal structures of noun phrases containing *background radiation*.

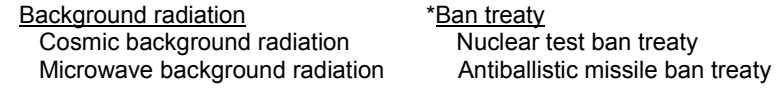

Figure 2.6 Correct and incorrect noun-phrase hierarchies

A more subtle point about examples like those shown in Figure 2.6 was made by Marchand (1969: Ch. 4) in a discussion of the process by which compound nouns become lexicalized. As he said, once a compound is established, it is free to appear in the same syntactic positions as single words. In expository text, and especially in technical text, the modifier position of a compound noun is a common location for lexicalized compound nouns, perhaps because noun phrases can be formed so freely from heads such as *department* from single-word modifiers such as *math department*. As a result, by many measures described in this dissertation, the phrases *aerospace engineering*, *sanitation engineering* and *electrical engineering* show evidence of lexicalization. Other examples extracted from the engineering corpus include *information technology*, *wastewater engineering*, and *health care*, which appear in

the longer noun phrases *information technology division*, *wastewater engineering virtual library*, and *health care providers*. In Section 3.4.3 of Chapter 3, I use the results of an algorithm for identifying the internal structure of noun phrases to construct a test that provides one source of evidence for the existence of lexicalized noun phrases in unrestricted text.

2.1.4.1. Algorithms for assigning internal structure

When confronted with the need to decide between [[information retrieval] experiment] and [information [retrieval experiment]] as the correct internal structure for *information retrieval experiment*, a human parser with access to deep semantic information might select the first assignment because English has a productive process for creating nominal compounds from a nominalized verb and its direct object. Other examples are *snow removal* and *error recovery*. The second structure is problematic because the nominal head *retrieval experiment* is missing an essential argument and is thus not a viable free-standing compound. The first structure assignment is preferred because *retrieval* is a deverbalized noun that needs a direct object such as *information*.

The correct answer can also be derived algorithmically from distributional evidence in the corpus without incorporating computationally expensive knowledge of deverbalized nouns and other lexical properties that may influence the internal structure assignment of compound and complex noun phrases. Lauer (1995) hypothesizes that the structure could be assigned by examining the mutual information of *information* and *retrieval,* as well as *retrieval* and *experiment*. If the first association score is higher than the second, *information retrieval* is a likely collocation, and the first structure [[information retrieval] experiment] is preferred. Unfortunately, this appealingly simple idea runs aground because a given corpus may not contain enough instances of the words needed to obtain reliable association statistics for all of the compounds in need of a structure assignment. To increase the number of observations, Lauer counts equivalence classes of words instead of raw tokens. Thus, compounds such as *information retrieval test* or *text retrieval*

experiment would be equivalent because *information* and *test* are in the same semantic field, as are *test* and *experiment*. He uses Roget's thesaurus to obtain equivalence classes, with mixed results.

Lapata (1999) developed an alternative corpus-based algorithm for assigning internal structure to noun-noun compounds and showed that its performance was superior to the predecessors reported by Lauer and his predecessors. Working with 100 million words of news text in the British National Corpus, Lapata studied this problem as a side issue in a larger research program devoted to the evaluation of corpus evidence for verb arguments. She argued that some cases can be resolved by referring to simple heuristics. For example, the verb phrase *killed henry phipps* probably has a single argument because a sequence of proper nouns is usually a single noun phrase. But the verb phrase *offer an express bus service* can't be so easily disambiguated. Is *express bus service* parsed as the single noun phrase [an [express bus] service]], as the alternative structure [an [express [bus service]]], or as two noun phrases [an [express bus]] [service]? To make a decision solely based on corpus evidence and a dictionary, Lapata devised an algorithm that works on linear sequences of three or more nouns. If the sequence n_1n_2 is in the dictionary, the structure [[n_1 n_2] n_3] is assigned; or if the sequence n_2n_3 is in the dictionary, the structure [n_1 [n_2 n_3]] is assigned. If neither sequence is in the dictionary, the log-likelihoods for both n_1n_2 and n_2n_3 are computed and the sequence with the highest value is chosen, assuming that they cross a threshold. For sequences of nouns whose log-likelihoods are below the threshold, the corpus does not support the claim that they are noun-noun compounds, and they are best interpreted as multiple noun phrases. The algorithm can be applied iteratively to longer phrases.

2.1.4.2. An extension

The Lapata algorithm produces impressive results, and I have incorporated it in my system for identifying lexicalized noun phrases, but I have made a few modifications. First, I extend her analysis to include adjective-modified phrases and explore other sources of linguistic evidence relevant to the assignment of internal structure in noun phrases. Moreover, unlike Lapata, I must process Web text of

unreliable quality. Web text that has been lightly parsed has fragments such as *windowwithborder windowwithtitle* or *news updates job opportunities*, which a parser recognizes as legal noun phrases but are probably remnants of programming language code or the text of buttons that are adjacent to one another on a displayable page. My modifications to Lapata's structure-assignment algorithm can eliminate these phrases from further analysis by assigning structures such as [windowwithborder] [windowwithtitle], indicating two adjacent simple nouns; or news] [updates][job opportunities], indicating adjacent nouns, one of which is missing a left modifier.

The extensions to the Lapata algorithm are derived from the argument presented in Godby and Reighart (1999) that corpus evidence can be used to identify single-word tokens of topical interest in a subject-restricted collection of text. In earlier proposals, such words were identified by consulting a dictionary, or by comparing word frequencies against the words in a background corpus that covers a broad range of subjects. For example, Zhou and Dapkus (1995) argue that topical terms such as *moon*, *stars*, and *galaxy* would be relatively more frequent in a corpus about astronomy than in a corpus of mixed subjects.

The same result can be obtained by constructing, for each word, a ratio of local syntactic contexts, like those used by Hindle and Rooth (1993) for solving the conceptually similar problem of attachment ambiguity in sentences with prepositional-phrase complements. Consider, for simplicity, the proposal that *moon* is commonly found in compound nouns such as *full moon*, *harvest moon* and *waxing gibbous moon*, though it also frequently appears without any modification. This amounts to the claim that when all of the left contexts of *moon* are tabulated, markers of clause and phrase boundaries such as conjunctions, determiners, prepositions and punctuation marks are relatively more common than adjectives or nouns. Along with *moon*, the nouns that appear most frequently without modification in a corpus of texts about astronomy are *sun*, *earth*, *atmosphere*, *evening*, *nature*, *morning* and *equator*. Conversely, the nouns shown in Table 2.6 require modification because the left contexts contain modifiers more frequently than boundary markers. The same analysis can be extended to identify the logical boundaries of longer sequences of nouns.

Term	Sample collocation
einstein	albert einstein
herschel	william herschel
armstrong	neil armstrong
belt	asteroid belt
borealis	corona borealis
centauri	alpha centauri
cluster	star clusters
way	milky way

Table 2.6 Astronomy terms that require modification

This method for identifying noun-phrase boundaries is consistent with Lapata's algorithm for assigning internal structure to multi-word phrases and can be used in two contexts. First, it provides input that may settle the indeterminate cases, where the log-likelihoods are too low to support the assignment of any internal structure. In a phrase such as *software engineering major*, the log-likelihoods for *software engineering* and *engineering major* are low in the engineering corpus because all three words are common and appear in many other combinations. But the low log-likelihood score may be an accidental failure due to an unfortunate combination of words. Corpus evidence can be used to determine whether *engineering* tends to form larger phrases, and whether those phrases branch to the left or the right.

The algorithm is executed when part-of-speech tags are partitioned into modifiers and boundaries and the text is partitioned into sequential three-word windows. Then, for each word at position 2 in the window, two pairs of log-likelihoods are calculated, one pair each for the left and right flanks. The calculations are based on two sets of counts: those of the word paired with modifiers, and those of the word paired with boundaries. If the log-likelihood of the word-modifier pair is higher than that for the word-boundary pair, then the centered word expects a modifier at the flank; otherwise it expects a boundary. To illustrate, consider how the information obtained from this calculation might resolve an indeterminate case such as *software engineering major*. If evidence from the corpus suggests that *engineering* usually appears with a modifier, the structure [[software engineering] major] can be

tentatively assigned. If *engineering* itself is usually a modifier, the structure [software [engineering major]] is favored. And if *engineering* appears most commonly in boundary environments, the sequence is assigned a structure that represents the three separate nouns [software] [engineering] [major].

The same algorithm can also be used to check the integrity of the boundaries of the sequence of words identified by the noun-phrase parser, an issue not directly addressed by Lapata or others who devised noun-phrase structure algorithms based on corpus evidence. For example, the words *image processing* may have a high log-likelihood score, which can be used to assign the appropriate structure to a larger phrase such as [[image processing] software]. But corpus evidence might reject the the phrase as ill-formed because the log-likelihood for *image* paired with a modifier is high, which suggests that the noun-phrase extractor erroneously truncated the left flank of a longer noun phrase such as *document image processing software.* A check like this is especially important for Web text of uncertain quality. The corpus I have used for this study contains many noun phrase sequences such as *organization agreements*, *rgb column vector*, *directive background document followup*, *control information* and *electronic form postscript*. These phrases are extracted from Web pages whose layout may be as complex as a magazine page and which a partial parser is not powerful enough to decode. They are best excluded from further study.

2.2. A system architecture for recognizing lexicalized noun phrases

I am now in a position to propose a system architecture for extracting lexicalized noun-phrase candidates. As shown in Figure 2.7, the most straightforward system is simply a linear sequence of the processes described in the previous section. Two components require further comment. The component that assigns internal structure is not literally necessary for the analysis that I describe in Chapter 3, whose input is the raw list of noun phrases extracted from the corpus. The internal-structure component is applied at a much later stage to obtain a source of evidence for the lexical status of noun phrases, which I describe in Section 4.2.2.2 of Chapter 4. Similar comments apply to the component that calculates log-likelihoods. Though

the calculation is done on the raw text, when the raw frequencies of words in the corpus and pairwise frequencies of adjacent tokens can be tabulated, the information is used much later.

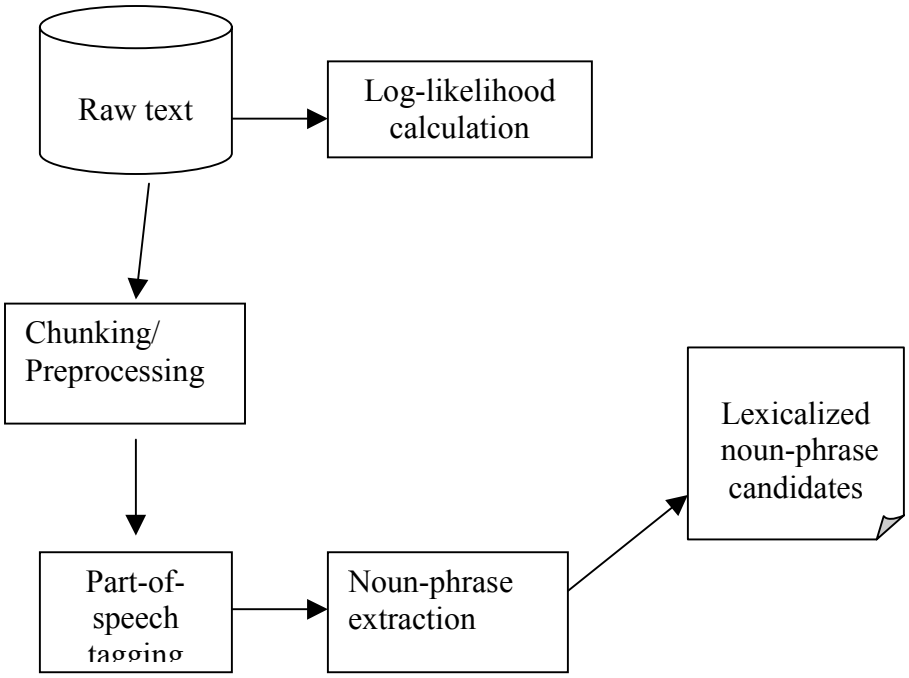

Figure 2.7 Process flow for extracting lexicalized noun-phrase candidates from text

Together with Chapter 1, this chapter builds the foundation for the study at the core of this dissertation. Starting with a literature review that emphasizes philosophical issues and data obtained from observation as well as intuition, we have developed the necessary foundation for an empirical study that supports and extends the theoretical linguistic arguments regarding the significance of lexicalized noun phrases. When the computational techniques reviewed in this chapter are applied to a large corpus, the result is a list of noun phrases whose lexical status must be evaluated. This is a large subject, which I explore in the next two chapters.

CHAPTER 3

CORPUS EVIDENCE FOR LEXICALIZATION

3.0. Introduction

With computational and theoretical linguistic grounding, I am now ready to address the central problem of this dissertation. How can lexicalized noun phrases be identified in a corpus of coherent text about engineering? This is not simply a technical problem that can be solved by writing software that encodes the formalizable insights of previous scholarship; Chapters 1 and 2 identified serious gaps in our understanding that would guarantee failure. In this chapter, I attempt the sometimes tricky analytical task of filling in some of these gaps by appealing to insights and methodologies from several research traditions. From corpus linguistics, I adopt a respect for data that can be observed and counted, as well as the philosophical view that language is best understood in context. From theoretical linguistics, I adopt the goal of seeking a solution that is linguistically natural and reflects insights about language that generalize beyond the current narrowly defined problem. And from psycholinguistics, I adopt the view that observed language is a sample of behavior, which enlightens our understanding of differences between lexicalized and syntactic noun phrases in ways that the other perspectives cannot. Of course, this program is ambitious and I can only sketch the highlights here; a more coherent account must await the concluding remarks in Chapter 5.

The immediate deliverable from this analysis is a list of linguistic contexts that will be sumbitted to the computational model described in Chapter 4, which attempts to classify noun phrases as syntactic or lexicalized, using the contexts as sources of evidence. Though the input includes tests identified by linguists including Marchand, Downing, and Levy, the primary purpose of this chapter is to argue that a large corpus is rich with evidence that supplements the traditional accounts, as long as we respect an

essential quality of the data that previous scholarship has already pointed out. As Levy (1978) said, there is no categorical evidence that distinguishes lexicalized from syntactic noun phrases. But this conclusion does not have to be an admission of failure or an impediment to progress. Relevant evidence that is robust but not categorical is still valuable, and can be evaluated with the appropriate computational tool.

3.1. A first look at the engineering corpus

The corpus used in this study was obtained with an algorithm that searches the Web for sites about engineering, starting from a list of suggestions supplied by a human operator.When an appropriate site was found, all local documents were downloaded and the text was stripped out, which eliminated such extraneous material as HTML markup and pointers to images or other pages. Though the resulting pages vary considerably in size and rhetorical style, the harvesting algorithm successfully partitions the data into sixteen 30-megabyte blocks, each containing approximately 9,000 documents. Thus the total corpus consists of approximately 150,000 documents and occupies nearly 500 megabytes; the average document size is approximately 3,300 kilobytes. Koch (1998) describes the document selection process and the harvesting algorithm in more detail.

Of course, it is impossible to inspect a corpus of this size. Even the automated analysis that is the primary subject of this chapter and Chapter 4 uses only a subset of the data as input. The rest will be used to validate the analysis by making predictions about the unseen portions. Thus it is important to determine whether the partitions are relatively similar, at least in the ways that matter most to my research problem. Though I address this issue more extensively in Chapter 4, Table 3.1 gives an encouraging first impression.

The top half shows raw tabulations for the five most frequent words in the first six partitions, excluding closed-class words such as prepositions and determiners. The bottom half shows the five most frequent noun phrases in each partition that were recognized by the noun phrase parser, which I described in Section 2.12 of Chapter 2 and is here configured to recognize simplex noun phrases with no embedded prepositional phrases or conjunctions. The tabulations show remarkably high counts for words and

phrases that are suggestive of engineering topics. They also show such balanced distributions of the terms and phrases across the partitions that the rank order in Partition 1, which was used to organize the data in Table 3.1, closely matches those in the remaining five partitions. Neither outcome can be guaranteed from a semi-automated process that collects text of uncertain quality. For example, it is conceivable that the Web page harvester could have started from a human-supplied cue to search for documents about neural networks and filled only the first partition with them. Or it could have filled several partitions with documents that are not primarily about engineering because the automated process that identifies relevant documents is also fallible.

	Count in partition					
Word/phrase	1	2	3	4	5	6
Information	13,843	13,223	12,658	11,299	13,579	13,787
Page	10,875	10,316	11,876	9,304	10,636	10,192
Engineering	9,218	9,253	8,042	7,718	9,780	9,328
University	8,849	9,748	8,476	7,085	9,601	10,052
Research	8,191	7,738	7,641	6,604	8,312	8,136
Computer science	846	842	781	484	868	754
Electrical engineering	385	384	349	240	407	353
Mechanical engineering	368	430	392	327	502	366
Neural networks	273	171	209	216	198	241
Virtual library	226	252	261	234	295	240

Table 3.1 Frequency distributions for five nouns and noun phrases in six partitions of the engineering corpus

Figure 3.1 shows a sample document in the corpus. The underlined words and phrases represent the output of the noun phrase parser on this document, while the italicized expressions identify noun phrases that the parser failed to recognize. At 1,020 bytes, this document is only one-third the size of the average document in the corpus, but it illustrates some typical data-processing issues that my study must confront. First, the sample consists primarily of coherent text about an engineering topic, though it has not been carefully edited. It has spelling mistakes, such as *acquistion*, and errors of diction: *divested* should be followed by *itself*. Second, the automatic process that eliminates page markup introduces noise because headers such as *Acquisitions and Dispositions* are

combined with the text. In this example, the noun phrase parser correctly identified the nouns near the problem because partial parsers look for islands of text that contain relevant patterns without attempting a full sentence parse. But to such a parser, the header and the sentence following it, *acquistions and dispositions since its inception*…appear as a single unit of text and parsing mistakes can occur if the sentence following a header begins with a noun. On this document, the performance of the noun phrase parser is reasonable. It recognized 43/45 noun phrases, or 95%, and did not falsely identify any noun phrases. Moreover, the failures are due to part-of-speech tagging problems. *Proven*, *reserves*, and *existing* are ambiguously adjectives or verbs and the part-of-speech tagger made the wrong choices in the context of this document.

Acquisitions and dispositions Since its inception, the company has had an ongoing acquisition program which has resulted in periodic acquistions of natural gas and oil properties. Our most recent significant acquistion is that of the Manilla Village, Jefferson Parish, Louisiana. Acquired from Wolf Productions, Loiuisiana this producing area with *proven reserves* surmounting 300,000 BBL, proves to be yet another successful acquisition with promising future returns. During 1994 and early 1995, natural gas and oil reserves generally available for acquistion were at usually high costs. As a result and in reaction to the market conditions, the company focused most of its activities during that period on its exploration program as opposed to the acquisition of proved developed oil and gas properties. The company took advantage of this situation and divested of selected proved producing natural gas and oil properties. The company recently sold one of its interest in the Frio/Miocene Trend, Chambers County, Texas resulting in net gains of $800,000. The company will continue to evaluate *existing reserves* for potential sale if economically advantageous. Back to Cedyco Corporation Home Page

Figure 3.1 A document in the engineering corpus to which the noun phrase parser has been applied

With the foundations developed in Chapters 1 and 2 and the text of Figure 3.1 in view, I can now state the goals of the research that is reported in the rest of this dissertation. Philosophically, it is to identify the noun phrases that represent persistent names for significant concepts, using evidence from a corpus to guide the decision. This goal excludes three classes of nouns identified by the noun phrase parser in text samples like that shown above. First, single-word nouns are only of minor interest. They can be used to check the performance of the noun-phrase parser, as I have done here; and, more subtly, to provide evidence regarding boundary assignments in multi-word phrases, as I argued in Section 2.1.4 of Chapter 2. Second, proper names, dates and quantities, such as *Cedyco Corporation*, *Louisiana*, *1995* and *300,000 BBL*, are eliminated from further consideration. By linking my analysis to the tradition of scholarship that studies

compound nouns and lexical phrases, I am restricting my attention to common nouns which may happen to consist of more than one word. Finally, since I rely on an external source for judgments about lexicalized noun phrases in engineering texts, my analysis is necessarily skewed toward the identification of others that resemble those in the authoritative list. Accordingly, the lexicalized phrases are restricted to the domain of engineering, and noun phrases such as *net gains* and *market conditions* are excluded.

By the measures identified in this dissertation, the only candidates for further analysis in Figure 3.1 are *natural gas*, *oil reserves*, and *natural gas reserves*. Since only 3/45, or 6%, of the noun phrases in a small sample of text are potentially lexicalized noun phrses, the analytical task is to develop a filter that eliminates the far more common artifacts of syntactic productivity in a way that is linguistically informed and conceptually simple by making use of the information in the corpus to guide the decision.

When evaluating the filter, I make reference to a simple metric that is widely adopted by computational linguists and is a variant of the precision/recall measure used by information retrieval researchers when they assess the performance of text retrieval engines. Whenever a user issues a request to a search engine for information, the documents returned are either relevant or irrelevant to the information need. *Precision* is defined as the ratio of relevant documents to the total number of documents that are returned from the search; *recall* is the ratio of relevant documents returned from the search to the total number of relevant documents. More abstractly, precision is a measure of noise and recall is a measure of completeness. Though I discuss the issue of evaluation in more detail in Section 4.3 of Chapter 4, it is conceptually important here because it helps frame the problem I am trying to solve.

In the context of my problem, a high precision score indicates that the output consists mostly of lexicalized common noun phrases from the domain of engineering and is relatively free of extraneous problems such as tagging and parsing mistakes; a high recall score indicates that the output contains most of the lexicalized noun phrases in the corpus. I am more interested in maximizing precision precision than recall. Though both goals are desirable, they are mutually contradictory, and I am interested in simulating the editor's task of identifying candidates for a domain-specific lexical resource such as an

engineering thesaurus. A list of noun phrases that maximizes recall at the expense of precision creates too much work for the editors because they would have to sift through noisy output. A smaller, cleaner list might mean that some lexicalized noun phrases are missed, but it promises to be far more comprehensive than a purely manual effort.

3.2. A look at an engineering thesaurus

To begin the core task of my analysis, I take a glimpse at the contents of The Engineering Information Thesaurus (Milstead 1995). Since my professional expertise is linguistics, not engineering, I don't have consistently clear intuitions about the terminology of engineers, so I rely on a resource maintained by professional lexicographers who specialize in engineering as a trustworthy substitute. As the name implies, The Engineering Information Thesaurus is more than a simple list of terms. It is valuable to researchers primarily because it organizes terms into relationships that can be construed as a high-level map, or ontology, of the subject of engineering. Such relationships are familiar to users of general-interest thesauri such as Roget's Thesaurus. They include broader-than—*abutments/bridge components*; narrower-than—*accident prevention/blowout prevention*; and related-to—*acoustic microscopes/imaging techniques*. Nevertheless, I am interested only in the terms, not the relationships, and The Engineering Information Thesaurus is the most comprehensive resource available on the subject, with nearly 17,000 entries in the current edition. Though I refer to the Engineering Information Thesaurus for illustration in this chapter, many of the common terms listed there appear in other lexical resources that contain substantial lists of engineering vocabulary, such as the relevant schedules in the Dewey Decimal Classification (Mitchell 1996) and the Library of Congress Subject Headings[7], which is freely available to researchers.

Since my goal is to identify terms in the engineering corpus that are similar to those in an engineering thesaurus, it can't be accomplished unless some of these terms are also found in the corpus. This requirement suggests that the analysis at this stage has two dimensions. The first is a description of the salient linguistic characteristics of

engineering terms that are observable from an isolated list. For the subset of those that are cited in the corpus, the description can be rounded out with observations that can only be made from a context of coherent discourse. The result is a glimpse at how lexicalized and syntactic noun phrases differ, from the perspective of a data-intensive analysis.

To anchor the initial discussion, I have listed some hand-selected noun-phrase entries from the Engineering Information Thesaurus in Table 3.2.

Acoustic properties of materials	Highway engineering
Acoustical technology	Information retrieval
Aerospace engineering	Integrated circuits
Air traffic control	Mechanical engineering
Applied mathematics	Natural gas
Artificial intelligence	Nickel and alloys
Biomedical engineering	Nuclear engineering
Chemical agents	Nuclear fuels for fission reactors
Chemical apparatus	Nuclear power plant construction
Chemical engineering	Oil reserves
Chemical operations	Sanitation engineering
Chemical plants	Sanitary engineering
City planning	Strength of building materials
Combinatorial mathematics	Structural design
Communication engineering	Synthetic rubber
Data processing	Textile mills
Diesel engines	Urban planning
Highway engineering	Water analysis

Table 3.2 Selected entries from the Engineering Information Thesaurus

The entries in this table suggest that most, if not all, of the vocabulary is comprehensible to the educated non-specialist. Fortunately, it means that my study does not reduce to a purely formal manipulation of tokens that would effectively have to be treated like a language that I don't know. In the initial stages, at least, I can use traditional tools of linguistic analysis—as have others who have considered some of this data. For example, Levy (1978) discussed syntactic and semantic properties of the phrases *city planning*, *urban planning* and *electrical engineering*, all of which appear on this list.

[7] Accessible at ,http://www.loc.gov/catdir/cpso/lcco/lcco.html>

The most straightforward linguistic observation to be made about the entries in Table 3.2 concerns their syntactic form. Most of the terms are short, simplex noun-noun and adjective-noun compounds, an observation that is supported more rigorously by the tabulation shown in Table 3.3 obtained from a random sample of 1000 terms listed in the Engineering Information Thesaurus.

Token size	Count	Example
1	226	adsorbents
2	517	copper mines
3	191	waste heat utilization
4	43	underground air conditioning systems
5	19	ground vehicle parts and equipment
6	3	complementary metal oxide semiconductor integrated circuits
7	1	combined gas and steam cycle power plants
Total:	1,000	

Syntactic form (other than nouns)		
Adjectives	242	wooden bridges
conjunctions	34	tin and alloys
prepositions	19	settling of structures

Table 3.3 Token sizes and syntactic forms of entries in a random sample from the Engineering Information Thesaurus

Only rarely do the phrases contain prepositions and conjunctions, as in *nuclear fuels for fission reactors* or *nickel and alloys*, about 5% of the items in the sample. Missing altogether are complex noun phrases containing *that* clauses, such as *interpreted data models that describe concepts that are common to more than one Application Protocol*, a citation from the corpus. For the most part, the entries in the thesaurus conform to Levi's (1978) description of complex nominals. Though most are noun-noun compounds, about 24% are adjective-noun compounds such as *nonferrous metals*, *sanitary engineering*, *structural design* and *urban planning*, which cannot, in principle, be excluded from the analysis. Thus the adjective-noun compounds pass two of Levi's critical tests for inclusion as complex, or lexicalized, noun phrases, rather than syntactic

noun phrases. First, they perform the same semantic work as a noun in an essentially synonomous phrase (*urban planning* vs. *city planning* and *sanitation engineering* vs. *sanitary engineering)*. As the previous example suggests, many adjectives alternate with nouns to form parallel but non-synonymous expressions, such as the names of the various subdisciplines of engineering: *nuclear engineering*, *mechanical engineering*, and *industrial engineering* vs. *ocean engineering*, *communication engineering*, and *highway engineering*. Second, Levi observed that adjectives in syntactic phrases may be modified by *very*, as in the following citations from the engineering corpus: *very graphic presentation*, *very crude capitals, very simple interpreters* and *very short routines*. Not surprisingly, *little useful information*, *crude capitals*, *simple interpreters* and *short routines* are not listed in the thesaurus. Conversely, linguists as well as engineers judge the lexicalized thesaurus entries to be ungrammatical when they are modified by *very*, as in **very mechanical engineering* and **very urban planning*, and **very integrated circuits*.

My analysis would have to stop here if none of the thesaurus entries appeared in the engineering corpus. But *natural gas* and *oil reserves*, lexicalized-phrase candidates that I identified in the sample document shown in Figure 3.1, also appear in Table 3.3, a hint that some of the entries in the Engineering Information Thesaurus are also cited in the corpus. Table 3.4 shows the tabulation of the top ten single-word and noun-phrase thesaurus entries that are most commonly cited in the first six partitions of the corpus. These distributions show that the corpus has many citations of the thesaurus entries, which are more-or-less evenly balanced across the six partitions. The tabulations for the single terms are cleaner than those of the noun phrases because the most common citations also have nearly the same rank order in each partition. Among the noun phrases, the same ten noun phrases usually rank among the most commonly cited in each partition, but their rank order is scrambled from one partition to the next.

	Count in partition					
Term	1	2	3	4	5	6
research	1,915	1,860	2,062	1,784	1,937	1,875
technology	1,870	1,634	1,607	1,538	1,798	1,684
design	1,631	1,384	1,479	1,543	1,534	1,553
engineering	1,512	1,316	1,277	1,391	1,556	1,564
industry	1,051	970	1,014	1,001	1,031	1,066
control	1,032	1,085	981	905	1,033	1,230
water	1,002	1,215	852	984	923	854
paper	913	1,004	617	800	861	979
analysis	829	760	608	772	884	720
education	827	790	892	693	774	699

	Count-Rank in partition					
Noun phrase	1	2	3	4	5	6
computer science	444-1	470-1	417-1	243-1	426-1	390-1
neural networks	259-2	178-7	204-5	208-4	193-6	226-4
electrical engineering	226-3	242-3	207-4	168-6	259-4	249-2
artificial intelligence	214-4	114-13	98-12	180-5	147-9	176-6
civil engineering	205-5	208-5	220-3	166-7	284-3	161-7
mechanical engineering	186-6	253-2	221-2	208-3	305-2	207-5
chemical engineering	178-7	212-4	159-7	215-2	186-7	242-3
materials science	166-8	174-8	169-6	122-8	184-8	139-9
environmental engineering	159-9	151-10	124-8	119-9	136-11	99-14
information technology	120-10	145-11	111-9	107-11	153-10	150-8

Table 3.4 Citations of thesaurus entries in the engineering corpus

The data in Table 3.4 is a fortunate result for the purposes of my study, but not one that I could assume without verification. Thesauri, dictionaries and other lexical reference works often adopt stylistic conventions that may reduce the hit rate when their entries are used literally as search terms in a corpus of naturally occurring text. For example, they may list only the plural forms; they may split compounds into their component parts, creating entries such as *engineering—electrical*; or they may retain archaic language such as *aeroplanes*. But much of the terminology in the Engineering Information Thesaurus is vocabulary that is actually used by engineers. Some of the thesaurus entries are even highly frequent in the corpus. For example, all of the noun

phrases listed in Table 3.1 (except for *virtual libraries*, which is not listed in the thesaurus), as well as *data processing*, *geothermal energy*, *heat transfer*, *high energy physics*, *information retrieval* and *laser applications*, rank in the top half when they are tabulated as words along with more conventional tokens, such as *transportation*, *bridges* and *surveying*.

Table 3.4 hints at another conclusion that I can use to focus the rest of my analysis. Despite the evidence from Table 3.3 showing that the entries in the thesaurus exhibit a small number of the syntactic forms—including prepositional phrases and conjunctions—that, in Marchand's analysis, justified a superordinate category *lexical phrase* to which compound nouns and Levi's complex nouns belong—none of these entries are counted among the most commonly cited in the corpus. This conclusion is supported more explicitly by the data in Table 3.5, which shows a tabulation of the gross forms of the sample from the thesaurus that are cited in the corpus. In other words, no credible corpus evidence exists for deciding the status of terms in the thesaurus that formally resemble familiar collocations such as *bread and butter* and *beast of burden*, which I discussed in Section 1.2.1 of Chapter 1, in a review of Marchand's research on this topic.

Token size	Count	Examples
1	204	turbomachinery, visualization, ultrasound, mechanization
2	343	white noise, voice recognition, transfer functions, statistical thermodynamics
3	69	time series analysis, solid state relays, water supply systems, air traffic control
4	6	high performance liquid chromatography, ocean thermal energy conversion
5	2	reflection high energy electron diffraction
Total: 624		

Syntactic forms other than nouns: counts and examples		
Adjective	249	thermodynamic stability, seismic waves, rare earth elements
Conjunction	0	
Preposition	0	

Table 3.5 Token sizes and syntactic forms of the thesaurus sample with corpus citations

As a result of this gap in the data, nothing interesting can be inferred about noun phrases that appear in Table 3.2 such as *nickels and alloys*, *acoustic properties of materials,* or *nuclear fuels for fission reactors*. Their absence in the corpus may be due either to sampling errors, or to the fact that they reflect the editorial style of the thesaurus, not the natural vocabulary of engineers. Nevertheless, the scarcity of these forms suggests that little of interest is lost if the noun-phrase parser is configured to exclude conjunctions and prepositions from further analysis.

So far, the analysis of the entries in the Engineering Information Thesaurus that are cited in the corpus provides empirical support for claims made in previous studies of lexicalized noun phrases. As Marchand, Downing, Levi, and Justeson and Katz, and other linguists that I discussed in Chapter 1 observed, noun phrases that function as words are almost always short, simple, and restricted in their syntactic form. But lexicalized noun phrases are restricted in another important respect that is difficult to document without the benefit of a corpus: they are frozen. As a result, they exhibit far less syntactic variability than the lexical semantics of their components would predict. This observation is perhaps the basis of one of Levi's tests that determines whether noun phrases containing adjectives are syntactic or lexicalized phrases. As she argues, lexicalized noun phrases—'complex nominals,' in her terminology—are formed with attributive adjectives, such as *electrical* in *electrical engineer* or *urban in urban planning*, which can appear only in the prenominal position. Syntactic phrases, on the other hand, are formed with predicative adjectives, such as *short* and *simple*, which can appear either as prenominal or predicate modifiers. Thus, it follows from the attributive/predicative distinction that the phrases *simple interpreters* and *interpreters that are simple* are grammatical, while **engineer who is electrical* is not. However, the lexicalized noun phrases in the engineering corpus present a problem for Levy's distinction: predicative adjectives are commonplace. Even the sparse sample represented in Tables 3.2-3.5 includes almost as many examples as Levy discussed in her entire treatment of the subject: *integrated circuits*, *natural gas*, *sanitary engineering*, *synthetic rubber*, *structural design*, *wooden bridges*, *artificial intelligence*, *geothermal energy*, and *white noise*.

Nevertheless, evidence from the corpus suggests that these expressions are as fixed as Levy's examples *electrical engineer* and *urban planning*. Lexical semantics do not restrict the formation of sentences such as *circuits that are integrated* or *intelligence that is artificial,* or *gas that is natural*, yet alternative expressions like these are not observed. Though this may be another accidental gap in the data from which nothing interesting can be inferred, lexicalized noun phrases achieve a high score when they are submitted to a statistical model of lexical collocation, the log-likelihood statistic that I discussed in Section 2.1.3 of Chapter 2—and this score is a hint that the components of the phrase participate in very few alternations. Similar comments apply to Levy's examples. *Electrical engineer* is a highly frozen expression, but not because of a limitation imposed by lexical semantics. As a handyman once said when he diagnosed a problem in my house, 'I think your problem is electrical.'

The frozenness of the expressions in the thesaurus is reflected in the high mean log-likelihood scores of those that appear in the corpus, relative to all noun phrases that were extracted using the parser described in Section 2.1.2 of Chapter 2, which was configured to recognize simplex noun phrases with no embedded conjunctions or prepositions. Mean log-likelihood scores for all multi-word noun phrases in the first partition of the engineering corpus are shown below.

	Mean log-likelihood
All noun phrases in the sample	58.85
thesaurus entries	74.23

Table 3.6 Log-likelihood summary statistics for two classes of noun phrases in first partition of the engineering corpus

Among the thesaurus entries are some noun phrases that are exceptionally frozen. For example, *artificial intelligence* has the highest log-likelihood score, 4,170, which is a consequence of the fact that *artificial* has a frequency of 235 and the phrase has a

relatively high frequency of 214. In other words, 91% of the occurrences of *artificial* are followed by *intelligence* in the first partition of the corpus. It is instructive to examine the other occurrences of the root word *artificial*; they are listed in Table 3.7.

```
4 Artificial lakes
1 Artificial life
2 Artificial organs
6 Artificial neural networks
7 Artificial radioactive elements
1 Artificially accelerated atomic particles
```

Table 3.7. Citations of *artificial* in noun phrases other than *artificial intelligence*

The phrase *artificial organs* is listed in the Engineering Information Thesaurus and *artificial life* is cited in many other subject indexes. But the partition of the engineering corpus that I processed for this example contains no instances of predicative adjectives and only one phrase containing the adverbial form. The adjective *artificial*, it seems, has a much more restricted distribution than its morphology or semantics would predict in a corpus of engineering documents.

The same point can be made about *integrated*, which appears in the engineering corpus and in the Engineering Information Thesaurus in the phrase *integrated circuits*. The higher than average log-likelihood score of 340 is evidence that the phrase is relatively frozen. But it is much lower than the score for *artificial intelligence*, which reflects something about the subject matter of engineering text. A log-likelihood score that is only somewhat elevated suggests that integration is an important principle and that circuits of all kinds are widely discussed, including *arithmetic, asynchronous, calculator, digital, logic, low-voltage, non-linear, solid-state,* and *vlsi circuits*. However, despite the high frequency of *integrated* in the corpus—and the fact that *integrate** is a root that can be grammatically realized as a prenominal or an attributive adjective, as well as a verb or a noun—the distribution has noticeable gaps. The corpus contains no examples of *integrated* as a predicative adjective; and 91% of the occurences of the root *integrate* are attributive adjectives, appearing in phrases such as *integrated gas*, *computer-integrated construction*, *computer-integrated manufacturing*, *integrated optics*, *integrated transport*

systems, *integrated manufacturing* and *integrated application resources*. The rest of the occurrences of *integrate** in the first partition of the corpus are various verb forms, as summarized in Table 3.8.

Syntactic Form	Count	Example
Prenominal adjective	1592	*integrated* optics
Nominalization	13	synchronized multimedia *integration* language
Verb		
Present	49	This package *integrates* connection, file transfer and terminal emulation modules.
Infinitive	62	…in order to *integrate* their options
Passive	25	The temporal subschema that will allow geographic information to be *integrated* with other aspects of information technology.
Total :	1,741	

Table 3.8 The distribution of *integrate** in the first partition of the engineering corpus

By contrast, consider the adjective *small*, which can be modified by *very*, can be attributive or predicative, and is similar in frequency in the corpus sample to *integrated*. Table 3.9 shows that 73% of the occurrences of *small* are attributive—still a skewed distribution, given the variety of syntactic contexts and morphological forms that possible for *small*, but it shows that a word that primarily forms syntactic phrases exhibits greater syntactic variability than *integrated*, a word that primarily forms lexicalized phrases in this corpus. Accordingly, 85% of the word pairs starting with *small* have log likelihoods that are smaller than the standard deviation for the collection, compared with 47% of those starting with *integrated*. This reflects the fact that *small* primarily combines with a large number of highly frequent words to form infrequently occurring phrases such as *small contribution*, *small processors*, *small data*, *small noise*, *small quantities*, *small as*, *small scale* and *small group*.

Syntactic Form	Count	Example
Prenominal positive	1,265	…a very *small* number of cases are available electronically
Predicative positive	99	…in a sense this is dithering, but with device dots so *small* that acceptable pictures can be produced at reasonable viewing distances
Prenominal comparative	122	The *smaller* design allows a faster clock rate to be achieved.
Predicative comparative	222	Binary format files can be loaded up to 5 times faster and are some 25% *smaller*.
Prenominal superlative	24	…with two exceptions, the *smallest* size fraction
Total:	1,732	

Table 3.9 The distribution of *small* in the first partition of the engineering corpus

3.3. Toward the identification of lexicalized noun phrases from corpus evidence

Starting with noun phrases identified by lexicographers who specialize in engineering, the analysis in the previous section suggested that the entries in the Engineering Information Thesaurus can be said to represent the living vocabulary of engineers because they are frequently cited in a large corpus of academic engineering text. And, as vocabulary, these noun phrases differ from syntactic noun phrases, which are the usual reflex of linguistic creativity, in many ways that can be observed from corpus evidence. The predictive power of this analysis can be tested by applying it to the central problem of this dissertation. Is it useful for discovering additional noun phrases in the corpus that resemble those listed in the thesaurus but are, as of now, not listed? This is a difficult test because the analysis doesn't have access to a key piece of information that the human expert can take for granted. As I argued in Sections 1.1 and 1.2 of Chapter 1, lexicalized noun phrases involve a relationship between language and a persistent object or concept, which can be known but not directly observed.

One encouraging result of the analysis at this stage is the suggestion that the hunt for lexicalized noun phrases can be narrowed down dramatically because they appear to be restricted in their syntactic form. In effect, the noun-phrase parser needs only to recognize fairly short, grammatical sequences of nouns and adjectives. This conclusion is nothing new. The evidence I have considered merely supports the conclusions of other

linguistic studies of multi-word terminology, technical and otherwise. Carroll (1985), a psychologist who studied the genesis of names in experimental settings, suggested that the formal simplicity of noun phrases that function as names is grounded in cognitive psychology. In one experiment, human subjects were supplied with lists of ingredients for recipes and expected to assign names to them. Most followed the simple strategy of creating compound nouns whose modifiers were the ingredients and the head was the product: for example, *molasses peanut cookie* was the name assigned to a cookie made from molasses and peanuts. Carroll speculated that the act of naming is difficult enough without incurring the additional burden of syntactic complexity.

Nevertheless, as I argued in Section 2.1.3 of Chapter 2, syntactic form alone is an insufficient criterion for distinguishing lexicalized from syntactic noun phrases. Consider the least problematic case, the noun-noun compounds that Downing (1977) studied. A corpus of engineering text has many noun-noun compounds such as *aircraft maintenance*, *aluminum chloride*, *antenna design*, *wind conditions* and *wildlife toxicology*. Given the arguments I made in Section of 1.2 of Chapter 1, can't we assume that noun phrases of this form are always names? Unfortunately, the answer is 'no.' A large corpus has many noun-noun sequences that cannot reasonably be interpreted as lexicalized because they are the artifacts of noisy automated processing. For example, some are obtained from common Web page layouts, such as *news classifieds archives*, which are probably names that appeared on navigational buttons and were accidentally represented in a text file as an uninterrupted string when layout markup was carelessly stripped out. Some spurious noun-noun sequences result from parsing failures that can be traced to failures by the part-of-speech tagger, such as the phrase *bet/NN cause/NN problems/NN*. Some cannot be removed from their discourse contexts without a severe loss of meaning, a point that I consider in more detail in Section 3.5 of this chaper.

A more serious problem is that the analytical tools I have used so far are insufficiently powerful to distinguish lexicalized noun-noun compounds from the nonce compounds that were the focus of Downing's study. As I discussed in Section 1.1 of Chapter 1—and, more technically, in Section 2.1.3 of Chapter 2—statistics such as log-likelihood, which measure the strength of association underlying the claim that a

sequence of words is a lexical collocation, make incorrect predictions on rare, sparse, or non-occurring data. And nonce compounds are, by definition, rare or unique. A heuristic for filtering noun-noun compounds based only on frequency might correctly eliminate *control specification problem* and *bullet sprawl* in the engineering corpus, but only at the expense of also eliminating low-frequency noun-noun compounds in the corpus, such as *pattern recognition*, *application programming*, *user interface design* and *video electronics*, all of which are listed in the Engineering Information Thesaurus. Because of this problem, it is necessary to proceed with caution. Interpretations of rare occurrences are problematic in the engineering corpus because it represents a sample of engineering discourse that may continue to grow and change. Thus, *control specification problem* may be a nonce compound in the part I have analyzed, but it may be frequent enough in a later sample to be identified as lexicalized. Claims about relative frequency are on firmer methodological ground when they are made with respect to the Engineering Information Thesaurus, which is a work of scholarship that purports to represent a complete map of the important concepts of engineering.

Of course, this account also fails to consider nouns modified by adjectives, which constitute over 20% of the entries in the thesaurus and are arguably the most difficult part of the analysis because no clear criteria have been established for determining which adjectives can be included or excluded in a lexicalized noun phrase. And the real answer may be that no such criteria can be identified. This result would have been anticipated by the studies of Lyons and Marchand that I discussed in Chapter 1, which claimed that speakers and writers can create persistent expressions using any linguistic means at their disposal. But I believe it is possible to advance the analysis with a simple observation. When a noun phrase is identified for inclusion in a lexical resource, it is, according to expert judgment, no longer the property of an individual speaker because it is used by a community of speakers as the conventional name of an object or concept. Thus, a lexicalized phrase must retain a constant meaning when it is removed from a particular text or context of use. Accordingly, adjectives or nouns that imply dependencies on a

text, or on common but changeable elements in a situation such as time, space, other objects or concepts under discussion, or the properties of a speaker, are rarely found in lexicalized phrases. Some of these classes are listed in Table 3.10.

Anaphoric and deictic elements: *subsequent*, *next*, *previous*, *aforementioned*
Cardinal numbers: *first*, *second*, *third*
References to time and space: *past*, *future*, *early*, *late*, *distant*, *nearby, eventual*
References to a speaker's attitude, state of mind, or state of knowledge: *interesting*, *daunting*, *difficult*, *obvious*, *terrible*, *particular*, *significant*, *special, favorite, excellent*
Degree adjectives: *large*, *new*, *small*

Table 3.10 Semantic classes of adjectives and nouns that rarely occur in lexicalized noun phrases

The sample from the Engineering Information Thesaurus has only seven citations containing the adjectives listed in Table 3.10. All involve the degree adjectives *large* and *small*: *small automobile engines*, *small nuclear reactors*, *small power plants*, *small turbomachinery*; *large scale integration*, *large scale systems*, and *large screen projection television*. In Carroll's terms, the rarity of context-dependent words in lexicalized noun phrases may be a reflex of the psychological difficulty of naming. If they are admitted as part of a name, a detailed context would have to be carried along as part of the name's referent. Of course, names may sometimes contain these elements if a community can agree upon a stable referent or a conventional interpretation that can be divorced from a particular context of use. But they are confusing to outsiders, perhaps because the preferred interpretation is the context-dependent one: *small nuclear reactors* compared to what, or according to whom? To consider an example from a different subject domain, I once attended a meeting with entrepreneurs from a Web startup company. When they described their financial circumstances, they used the phrases *first-round* and *second-round funding*. The true meaning of these expressions was lost on me because I simply assumed that they had tried a couple of times to sell their ideas, according to the needs of their project. They had to explain patiently that cycles of funding are part of an established ritual for obtaining venture capital.

Of course, I do not wish to claim here that lexicalized noun phrases are never formed from the classes of adjectives listed in Table 3.10. Some are, and they may be elevated to ordinary discourse. For example, to American English speakers, *terrible twos* are rambunctious two-year-old children; *significant others* are adult partners of unspecified gender, sexual orientation or marital status; and *second-stringers* are athletes who are skilled enough to be chosen for a sports team, but not skilled enough to play very often. But in terms of corpus evidence, lexicalized noun phrases formed with context-dependent adjectives are rare relative to the frequency of the adjectives. Consider for example, the two lists shown in Table 3.11, which represent the most common noun-phrase bigrams in the first partition of the engineering corpus formed from two adjectives—only one of which, in my analysis, is a context-dependent adjective. In this sample, *neural* and *significant* are closely matched in frequency: *neural* occurs 470 times, while *significant* occurs 511 times. Only *neural* is used to name concepts. As a result, readers can reasonably guess the subject domain from which the phrases shown in the left half of Table 3.11 are obtained: coherent text that contains these phrases routinely discusses topics either in artificial intelligence or neurobiology. Moreover, several of these phrases, including *neural network(s)*, *neural net* and *neural computers*, appear in the Engineering Information Thesaurus or the Dewey Decimal Classification. The phrases that are not listed in lexical resources, such as *neural controllers*, can be issued as a queries to a Web search engine, where related concepts such as *neural controllers*, *evolutionary algorithms*, *robots*, *fuzzy-neural methods* and *mathematical neuron model*s may be discussed, and where a patient search may eventually turn up a definition. However, none of these points can be made about the noun phrases formed from *significant*. Only one phrase in this list—*significant results*—has a specialized meaning that might have to be defined for the benefit of readers who are not familiar with elementary concepts in statistics.

neural networks	significant amount
neural network	significant increase
neural nets	significant contributions
neural computers	significant impact
neural systems	significant progress
neural tube	significant number
neural prosthetic	significant earthquakes
neural models	significant improvement
neural controllers	significant results

Table 3.11 Noun-phrase bigrams created from two classes of adjectives

The arguments I have just made suggest that the corpus can provide an additional source of evidence for distinguishing lexicalized from syntactic noun phrases: context-dependent adjectives are more likely to modify syntactic noun phrases. This principle must be stated carefully because it is obviously possible to modify lexicalized noun phrases with context-dependent adjectives, producing results such as *my favorite electrical engineer* or *previous seismic waves*. But when the noun phrase is a bigram, it is more likely to be syntactic than lexicalized. Table 3.12, which compares the mean log-likelihood scores of bigram noun phrases containing context-dependent adjectives with all noun-phrase bigrams in the first partition of the engineering corpus, supports this observation. The low log-likelihood scores for the bigrams with context-dependent adjectives reflect the fact that these adjectives are highly frequent and are attested in many combinations. For example, *available* has a frequency of 4,480, *talent* has a frequency is of 37, *available talent* has a frequency of 1, and the log-likelihood score is 2.19 in this corpus.

Phrase type	Score	Examples
All noun-phrase bigrams	10.13	zinc dust, yellow pages, academic collaboration, air duct, alternative lifestyles, basic interface, buckyball bibliography, carbon paper, characteristic symptoms, civil engineering, interstellar clouds, japanese submarines, labor costs
Noun-phrase bigrams with context-dependent modifiers	6.59	acceptable alternatives, additional parameters, available talent, certain browsers, considerable work, enough memory, exciting thrust, favorable kinetics, first biomass,further optimizations, second prototype, unacceptable vibration

Table 3.12 Log-likelihood scores of two classes of noun-phrase bigrams in the first partition of the engineering corpus

I believe that this account of context-dependent adjectives provides a generalized and corpus-aware interpretation of many of the linguistic tests that Levi proposed for distinguishing lexical from syntactic noun phrases. For example, as Levi suggested, *very* is a reliable test for distinguishing the two classes of expressions. But in many cases, *very* is used either as a deictic element, as in *You're the very man I want to see*, or as an intensifier, as in a *very small component*, which usually indicates a single speaker's attitude toward a subject. As Levi also suggested, adjectives with a putative adverbial source—especially time-denoting adjectives such as *future*, *occasional*, *eventual* and *potential*—are also more likely to characterize syntactic than lexicalized phrases, as in her examples *potential enemy, former roommate*, *future dependents*, and *occasional visitors*. But in my view, these are just the context-dependent adjectives that denote a changing referent, an anchor in a single context, or a single speaker's state of mind or state of knowledge, none of which may be shared by a community.

This analysis is important for my goals because it suggests that the distribution of lexicalized and syntactic noun phrases in a corpus is not random, but is limited in ways that can be specified by linguistic principles. If so, it provides one reason why the log-likelihood statistic and other measures of statistical association come up short when they are applied to the problem of identifying lexical collocations in sequential tokens of coherent text: in their usual mode of application, they fail to take this linguistic context

into account. For example, an appeal to the log-likelihood score would make the wrong choices in this ranked list of adjective-noun bigrams from the first partition of the engineering corpus: *further information* 991.88; *first generation* 93.53; *second stage* 28.7; *synthetic lubricants* 24.90; *structural geology* 21.31; and *nonlinear equations* 11.33. The noun phrases in the bottom half of the list are citations from the Engineering Information Thesaurus and achieve low scores because they have low frequencies or are formed with words that are observed in many other combinations. But if we eliminate the top half because they are outliers in a distribution that has a lower-than-average log-likelihood score and appear in linguistic contexts that favor syntactic, rather than lexicalized noun phrases, we have overridden the inaccurate information provided by the log-likelihood measure in this narrow application.

Unfortunately, the linguistic evidence presented so far is relevant to the identification of syntactic phrases, but is less helpful for finding the real objects of interest, partly because of the limitations of the evidence and partly because of the methodology I have adopted in this study. Scholars such as Levy consulted their intuitions and concluded that lexicalized noun phrases are ungrammatical in the environments that favor syntactic noun phrases. But without the luxury of trustworthy intuitions, I can't make the same inference because the non-appearance of a lexicalized phrase in a particular environment could be a random gap in the portion of the corpus I am currently looking at. For example, as I examine the first partition of the engineering corpus, I do not find any noun phrases listed in the sample from the Engineering Information Thesaurus that are modified by *very*, but I may discover some cases in a future study of other partitions. The claim that lexicalized and syntactic noun phrases have different distributions in coherent text would be stronger if we could identify linguistic contexts that favor lexicalized noun phrases—the complementary part of the analysis in Levy's seminal work. And for that, citations in the corpus of entries from the Engineering Information Thesaurus can be used as a starting point.

3.4. The corpus contexts of lexicalized noun phrases

Though it is a truism of contemporary linguistics that every sentence is potentially original, linguistic creativity often takes the back seat to social convention when the goal is effective communication. Deference to social convention is observed in part by referring to objects and concepts of common interest with usual and expected vocabulary instead of idiosyncratic expression. When someone does this successfully, we say that he talks the talk, or speaks our language. In Chapter 1, I suggested that writers often drop hints when they use an expression that is not their own. Here I wish to lay the foundation for the argument that, if we collect and analyze these hints, we can discover many lexicalized noun phrases using methods that are linguistically motivated and conceptually simple. The full development of the argument will consume the rest of this chapter and the next.

The goal of this section is to identify a short list of linguistic contexts in the engineering corpus where entries from the Engineering Information Thesaurus are commonly found—and, by extension, other lexicalized noun phrases that are suggestive of engineering topics. The list is by no means complete because, at this stage, I am interested only in proof of concept. To this end, I present a set of linguistic contexts that are semantically or syntactically restricted and can be viewed as thumbnail sketches that could be expanded into separate studies. The list is biased toward noun-phrase contexts because I first discovered linguistic contexts for lexicalized noun phrases in the output of the noun-phrase parser when it was configured to recognize noun phrases containing conjunctions and prepositional phrases. I will generalize my account in Section 3.5. As might be expected from the quality of the other data that is relevant to the distinction between syntactic and lexicalized noun phrases, the linguistic tests that emerge from this data may be robust but not exceptionless. But they can be evaluated with the computational tools that are introduced in Chapter 4.

3.4.1. Contexts for names of disciplines

I once heard a professor of comparative literature say, "Disciplines rest on chairs, professors, departments, journals and congresses." By this remark, he meant that areas of academic study have firm boundaries that are patrolled by high-status members of the profession. One probable linguistic consequence is that social conventions for the names of disciplines are firmly established and their referents are clear and stable. If so, discipline names are textbook cases of lexicalized noun phrases. When I wrote the first sentence of this paragraph, I knew that *professor of* creates a slot for a conventional name of an academic subject, so I filled it with *comparative literature*, a phrase that appears in dictionaries and college catalogs, instead of a creative description. In the engineering corpus, the noun-phrase objects of *professor of*, *chairman of*, *department of*, *journal of*, *school of*, *degree in*, *society for*, *career in*, *center for* and *workshop on* contain so many names of disciplines that I am conducting a separate study of this vocabulary (Godby 2001). Though some of the data is obtained from proper names, such as the names of journals or workshops, it does not violate my restriction against including proper names in the analysis because these are functional names that are made up of common nouns, much like the ones that Carroll observes in many of his studies. Table 3.13 shows some examples. The underlined forms are *not* listed in the Engineering Information Thesaurus.

Career in: electrical engineering, food science, automotive diagnosis, transportation

Degree in: physics, computer science, electrical engineering, nuclear engineering, geology, petroleum engineering, engineering, fire protection administration, environmental technology, forestry

Department of: optoelectronics, physics, public safety, pure mathematics, statistics, transportation, mathematical sciences, environmental protection, atmospheric sciences, radiation physics, biomedical engineering

Professor of: earth sciences, photogrammetry, computer science, engineering design, toxicology and environmental health, civil engineering

Journal of: astroparticle physics, adaptive behavior, control systems, computational physics, urban planning and development, offshore mechanics and arctic engineering, numerical heat transfer, neuropsychological rehabilitation, biochemistry, dynamic systems, irreproducible results, urban economics, leisure sciences, biological chemistry, food technology, environmental engineering, thermal spray technology

Workshop on: adaptive and learning systems, agrobiodiversity assessment, computer-assisted orthopedic surgery, digital image processing, computer architecture, earthquake-resistant design, historic mining resources, intelligent information agents, software metrics, uncertainty and probability in artificial intelligence, thin dieletric film metrology

Bibliography on: abstract state machines, active databases, algebraic specification, combinatorial optimization, complex systems, computational geometry, computer arithmetic, database systems, digital watermarking, electronic publishing, empirical software engineering, file systems, font readability, form features, geometric modeling, graphics, hardware verification, implicit surfaces

Study of: air flow contaminants, brain biomechanics, bubble and liquid flow characteristics, civil engineering, coastal geology, estuary hydrodynamics, finite amplitude thermal convection, fundamental problems, heat transfer, metallurgy, neural networks, plate boundary deformation, russia and east european countries, scientific and engineering principles, undergraduate biology majors

Table 3.13 Names of disciplines in the engineering corpus

Table 3.13 shows that all of the discipline-name contexts have citations in the Engineering Information Thesaurus. Presumably, many of the underlined noun phrases name areas of study, too, especially the ones that appear near the top of the table. To fully understand the organization in this data, it would be necessary to make reference to a knowledge ontology that has hierarchical structure, such as WordNet (Fellbaum 1998) or the Dewey Decimal Classification (Mitchell, et al. 1996). With such a tool, we could explore the hypothesis that some contexts, including *degree in* and *career in*, have noun-phrase objects that respresent highly established names and are listed at relatively high nodes in a knowledge hierarchy; while others, including *workshop on*, *bibliography on* and *study of*, have objects that are less likely to be lexicalized and less commonly represented in published knowledge ontologies. But when they are listed, they appear at low nodes because they are highly specialized and closer to the leading edge of knowledge. If the goal is the creation or maintenance of a knowledge hierarchy, the engineering corpus could be consulted for additional lexical clues that reveal even lower nodes in the hierarchical structure of concepts. For example, from the expressions *chemistry of hydrocarbon resources*, *electrochemistry of zeolite-encapsulated iron*, and *physics and chemistry of mercury cadmium*, we can infer, among other things, that professional chemists have a concentrated interest in the chemical properties of hydrocarbons, that electrochemistry is a sub-discipline of chemistry, and that mercury cadmium is an object of study in physics as well as chemistry.

I mentioned at the beginning of this chapter that one of the objectives of the research in this dissertation is to develop computer software that assists in the maintainence of a thesaurus. But I said that only in the spirit of focusing the study on an

interesting philosophical problem. If it were a literal objective, the list of noun-phrase contexts like those in Table 3.13 might be sufficient. All of these these contexts imply the existence of something talked about, written about, or studied—the Oxford Advanced Learner's Dictionary definition of 'subject'—the same something that motivates the listings in a specialized thesaurus, which provides a high-level map of a body of knowledge. Since thesauri are maintained by human effort, it is not surprising that some of the noun-phrase objects even in high-level contexts, such as *professor of*, fail to appear in the Engineering Information Thesaurus. These might be an easy source to mine for new vocabulary, but there is greater payoff in the low-level contexts because they contain hints about where the frontiers of knowledge are changing. But the data is noisier, as the list of noun phrases appearing in the context of *study of* in Table 3.13 shows. To compensate for the noise, I need to consider other sources of evidence and a more sophisticated computational model. In doing so, I can generalize the analysis beyond a small number of lexical choices in a corpus of academic text about engineering.

3.4.2. The contexts of quotation

The citation from a newspaper article cited in Section 1.3 of Chapter 1 starts out, "They're called popcorn fires…" The writer proceeds to define popcorn fires as fires that start in dorm rooms when students try to make popcorn with cheap, dangerous electrical appliances. This sentence is significant here because it illustrates a linguistic context of quotation or attribution, where the writer signals through his choice of the verb *call* and the passive voice that the noun-phrase object is not his own expression. Since it has an established referent that is shared by a community of speakers and writers and is presumably a fixed expression, it also fits the definition of lexicalized noun phrase that I have developed in this dissertation. And since it appears in a newspaper article about a topic that is not covered in the engineering corpus, it extends my argument for the existence of positive contexts for lexicalized noun phrases in several ways. But not surprisingly, given the pedagogic intent of much academic writing, similar citations are abundant in the engineering corpus.

Table 3.14 shows some examples. The underlined expressions are entries in the Engineering Information Thesaurus, a confirmation of the hunch that at least some of the noun phrases that appear in these contexts are lexicalized. Other noun phrases in these contexts are obvious words because they consist of a single token; still others, such as *multidimensional scaling*, *digital library* and *descriptive markup*, are listed in lexical resources such as the Dewey Decimal Classification. As in the case of linguistic contexts for discipline names, these contexts are noisy, in part because they reveal other potentially valuable lexical information, including the introduction of proper names and the definition of acronyms.

Each entry in a database is called *a record*.
The technique we are going to try is called *multidimensional scaling*.
Duplication resulted during what is called *overlay processing*.
A member of I is called *an IP (Internet Protocol) address*.
In the last decade, there has been a growing interest within the human-computer interaction (HCI) community in what is called *user-centered design*.
Increasingly, the collection of information available on the Web is referred to as *a digital library*, *a virtual library* or *THE global digital library*.
When the image has cartographic or bibliographic information added, it is referred to as *a remote-sensing map*.
By now, you have probably heard more than you wanted to know about the so-called *Y2K bug*.
Some applications of such interfaces are database queries, information retrieval from texts and so-called *expert systems*.
This kind of tagging, known as *descriptive markup*, should improve searching precision.
The type of structure adopted here...is known as a *frame*.
Includes links to other sites about the fire, which is known as *the 'Great Fire'* or '*Great Chicago Fire'*.
In three dimensions there are only five such solids known as the *Platonic polyhedra*, which were discovered by the Ancient Greeks.
…form what is generally known as a *hyper-cube* or *4-dimensional cube*.
They are also known as *floppy disks*, *stiffy disks*, *computer diskettes* or *floppy diskettes*.
..asynchronous communications. Also known as *serial communications*.
Java enables document page animation through special-use software applications known as *applets*.
..established an initiative, known as the *Digital Object Identifier*.
The IETF is working on a replacement to be known as the *Portable Network Graphics*.
The Telecommunications Association also maintains a political action committee, known as *TELEPAC*….
This process of trapping the longwave radiation is known as the *greenhouse effect.*

Table 3.14 Objects of ...known as, *is referred to*, *also known as* and *so-called*

3.4.3. Syntactic contexts

The contexts for discipline names and quotation/attribution yield over 100 noun-preposition collocations whose objects are often lexicalized noun phrases, including *basics of*, *courses in*, *information about*, *branch of*, *concept of*, *elements of*, *(on) the subject of*, *research on*, *topics in*, and *theory of*; others that are more narrowly focused by subject include *algorithm for*, *biology of*, *chemistry of*, and *science of*. In academic writing, word choices like these are common, but, I believe, finite—and their noun-phrase objects yield large quantities of data that will be evaluated more rigorously in Chapter 4. Here I wish to make the psycholinguistic argument that this data constitutes evidence that, in the prosaic act of constructing a sentence to satisfy a perceived communication need, speakers or writers constantly make minute choices about when to be creative and when to retrieve entries from an enriched lexicon that may have many phrasal entries. When engineers write the phrases *topics in combinatorial chemistry*, *topics in hazardous materials*, and *topics in networking*; or *theory of neural networks*, *theory of evolution*, and *theory of quantum mechanics*, they are not uttering collocations as lexicographers define them because these aren't fixed expressions; as a linguist, I might say *topics in computational lexicography* or *theory of categorial grammar*. Instead, *topics in* creates a context that favors collocations because the permissible objects are variable and potentially infinite, restricted only by the semantics of the dominating noun phrase.

The psycholinguistic argument would be stronger if I could generalize it beyond a small set of lexical choices and the confounding issues of subject domain and discourse style. Fortunately, the data I have examined so far show two syntactic contexts that harbor lexicalized noun phrases, and they have already been surreptitiously introduced. Table 3.14 cites sentences from the engineering corpus that contain four lists, all conjoined with *or— hyper-cube* or *4-dimensional cube; floppy disks*, *stiffy disks*, *computer diskettes* or *floppy diskettes; 'Great Fire'* or '*Great Chicago Fire*'; *a digital library*, *a virtual library* or *THE global digital library*. Other conjoined lists from the corpus are shown in Table 3.15. In a celebrated dissertation that treats lists as a subject for literary criticism, Robert Belknap says, 'Lists are deliberate structures, built with care

and craft, and perfectly suited to rigorous analysis (Monaghan 2001).' Even in the writing of engineers, lists of noun phrases are collections of like things. The examples in Tables 3.13 and 3.14 show sets of synonyms and names of disciplines at similar levels of generality. More importantly for my analysis, items in the list have similar lexical status, much like those in an ordinary grocery list: *eggs*, *bananas*, *potato chips*, *milk*, *toilet paper*, *peanut butter, hot dogs, coffee*. The underlined examples in Table 3.15 are citations from the Engineering Information Thesaurus, and the noun phrases that are listed with them are usually also lexicalized, a hypothesis that I will test in Chapter 4.

....the department is a centre for teaching and research, with divisions for mechanics, structures, materials, fluid mechanics and heat transfer, electrical engineering and information engineering

....engineering, artificial intelligence, expert systems, neural nets, and genetic algorithms

....our research is focused on advanced soil mechanics, cold region soil mechanics and environmental geotechnical engineering

....artificial intelligence and cognitive science

....product design and materials handling

...on manufacturing and health care

...aluminum, chemicals, forest products, glass, metalcasting, petroleum refining and steel

...the college offers degrees in engineering, computer science, construction engineering management, engineering physics and radiation health physics
...covering thermodynamics, heat transfer, fluid mechanics, materials science, control engineering, aerodynamics, dynamics of machines, etc.

International Computer Power is a world leader in the design, application and manufacture of high-performance AC power converters, rotary ups, the patented kinetic battery and power conditioners…

Electrical engineering encompasses a wide range of topics, including computers, communication systems, automatic control systems, digital systems, electronics, energy conversion, signal analysis, neural networks, fuzzy logic and integrated circuits.

The ISR's research projects encompass a diverse set of systems problems; they include intelligent control of processes, electromechanical motion control, wireless communication networks, high-speed satellite and terrestrial communication networks, telemedicine systems, and virtual factories for the manufacture of electromechanical devices.

…has active research-oriented graduate programs in the areas of power, communications, signal processing, control and systems theory, microelectronics, computer engineering, VLSI, applied electromagnetics, and nondestructive evaluation.

…Alpha Tec Ltd specializes in graphics, computer vision, signal and image processing….

Table 3.15 Conjunctions from the engineering corpus involving lexicalized noun phrases

The second syntactic context was introduced in Table 3.14 in the phrase *remote sensing maps*, where the modifier of the compound noun is an entry in the Engineering Information Thesaurus. Modifiers of multi-word compound nouns are commonly lexicalized, an observation that is supported by the large number of underlined expressions in Table 3.16, which indicate that these phrases are listed in the thesaurus. This pattern is general and not especially subtle. In speech, the lexicalized expression may be preceded or followed—but not split up—by a long and sometimes filled pause, as in this overheard sentence: *Surely, they will do a rights management …component*. The underlined phrase preceding the pause is the established name of important concept in the digital library world that I inhabit. In writing, the lexicalized modifier may be set apart with distinctive fonts or scripts. For example, I once saw a box of salt behind the counter at Wendy's that was labelled *French Fry Salt*. *French fry* was printed in red cursive writing, but *salt* was in yellow block letters. Since I don't have access to extralinguistic cues like these when analyzing the data in the engineering corpus, I must resort to corpus evidence to assign internal structure to long compound nouns so I can discover the lexicalized modifiers. The algorithm is described in Section 2.1.4 of Chapter 2.

When I view this data from a psycholinguistic perspective, I am compelled to ask two questions. First, what is special about the modifier position? Of course, lexicalized noun phrases can appear in compound-noun heads, too, and Table 3.16 lists two examples: *virtual library* and *research center*. But even two-word compounds refer to highly specific concepts, so there may be fewer occasions to distinguish them even further. Second, it is reasonable to wonder why the modifiers in multi-word compounds are often lexicalized, a hypothesis that I will support with more evidence in Chapter 4. If

I am right that a syntactic noun phrase describes an object or concept in a context where temporary relationships are important, we may need the full expressive power of syntax to make these relations explicit. For example, as I write this sentence, I notice a news headline, which is incomprehensible because it is nearly devoid of syntactic markup: *Big client firm reps accused spy*. Conversely, a lexicalized noun phrase simply refers to something persistent, and is thus psychologically no more complex than a single word that may be used as a modifier. Perhaps this difference underlies the exhortations by prescriptive grammarians to avoid the creation of novel compounds. For example, Kilpatrick (1999) warns that a long compound noun such as the one in the sentence *He had other improper organized crime ties* 'rolls more trippingly on the tongue' if it is recast as *He had other improper ties to organized crime*.

[information technology] division	[ferroelectric [random access]] memory
[wastewater engineering] virtual library	[water conservation] program
[reservoir engineering] professionals	[natural gas] association
[civil engineering] world	[algebraic mapping] networks
[concurrent engineering] research center	[latex particles] journal
[[high energy] physics] theory	[emergency preparedness] plan
[international [water resources]] association	[entrepreneurial management] program
[health care] providers	[long term] stability
[air pollution] effects	[radiological washdown] requirements
[computer programming] practice	[defense system] life cycle
[information retrieval] systems	[special purpose] machines
[electronic materials] conference	[variable complexity] modeling
[fuzzy logic] hardware	[compound interest] calculator

Table 3.16 Lexicalized noun-phrase modifiers of compound nouns

Taken together, the evidence presented in this section and the previous one imply that a corpus can be consulted to obtain positive as well as negative contexts for lexicalized noun phrases, all of which are linguistically motivated. In Chapter 4, I argue that the evidence can be viewed as a list that can expand and contract according to the goals of the research. But even with the small number of contexts I have described here, I can collect enough data for a sophisticated test.

3.5. Extending the analysis

If I am correct in claiming that coherent text contains many linguistic and metalinguistic cues indicating the lexical status of noun phrases, these cues should be present in texts covering a broad range of subjects. The examples so far have been drawn primarily from scholarly articles on engineering topics. But many of the same cues can be found in coherent text on other subjects, expressed at lower levels of formality than is usually found in academic writing. For example, the American voting public was introduced to *hanging chad* in the aftermath of the 2000 presidential election, which was contested in Florida because of faulty voting procedures. This phrase is embedded in remarkably similar contexts, as a small sample from Internet discussion groups obtained in November 2000 shows:

Is called

If this chad is loosened but not removed, it is called a "*hanging chad*."

The trick with the Votomatic is something called "*hanging chad*."

Conjunctions

Without warning, viewers were treated to plot twists! Conflicts! Mystery! False leads! Highly paid anchors with egg on their face! Even odd new terms like "butterfly ballot" and "*hanging chad*"!

That's a far different process than the one that we've seen on TV, where they're sitting here looking at a pregnant chad or a dimple or a *hanging chad* or a swinging chad. I mean, that's a heck of a lot different.

It gave rise to a new lexicon: swinging chad, *hanging chad*, tri-chad, pregnant chad and dimpled chad—a ”chad” being the bit of paper that did not fully detach when the voter punched a selection.

Modifier position in a compound noun

Punched cards give results that are more or less repeatable (ignoring the *hanging chad* problems).

"If I hear one more *hanging chad* joke I’m going to hit somebody,” says standup comedian Barry Weintraub, who staged his own mock campaign.

The race for the White House has been reduced to hand-to-hand, "*hanging chad*" combat between lawyers before courts and elections boards across the state.

Table 3.17 Local syntactic contexts for *hanging chad*

The discussion can be also be generalized by looking beyond syntactic and lexical environments for clues that a given noun phrase is lexicalized. Though I have focused on clues that can be identified in the phrase's local context, I believe that a larger window of discourse context is perhaps a richer source of evidence. The analysis of discourse is beyond the scope of this dissertation, but I cited two small examples at the beginning of Chapter 1 to illustrate some of the consequences that can be observed when a speaker or writer acknowledges that an expression is not original. Here is another, a snippet of overheard conversation from a librarian who had just returned from an international convention: 'I didn't even know the vocabulary the people were using. For example, I didn't know what *legal deposit* was. I found out that, in other countries, they're already talking about *legal deposit* loans!'

The meaning of *legal deposit*, like *hanging chad* in the previous set of examples, can be computed from the transparent composition of its parts. Both expressions also pass one of Levy's tests for syntactic phrases because the adjectives can be predicative. So why do the discourse contexts show evidence that these are words that must be defined before they are fully understood? In the above example, *legal deposit* is treated as a fixed expression that has a metalinguistic referent to *vocabulary* and an extra-linguistic referent that is unknown to the speaker. Even so, its status as a fixed expression permits its appearance as a modifier in the compound noun *legal deposit loans*, whose referent also eludes the speaker. The interaction of lexical choice and discourse is a topic that is only beginning to yield to computational analysis.

For example, Wacholder (1998) observed that, when all noun phrases in a coherent text are tabulated, topical concepts can be identified by creating clusters of noun phrases whose heads appear most frequently. As she observed, in a computer user manual, *file* and *disk* are frequent noun-phrase heads. A tabulation of all noun phrases containing these heads would, among other things, include a list of the kinds of files and disks that are discussed in the manual, giving some superficial clues about the topic of the document. This observation can be translated into a reliable heuristic that also works on collections of text in a restricted domain as well as single documents. Table 3.18 shows the most frequent noun-phrase heads and the noun phrases that were constructed from

them in two document collections: a corpus of political news and one of popular articles about astronomy. More examples from these collections are cited in Godby and Reighart (1999):

Program(s): Affirmative action programs, corporate welfare programs, domestic spending program, housing program, jobs program
System: Air defense system, ballistic missile system, child care system, criminal justice system, democratic system
Issue(s): Abortion issue, campaign issues, character issue, foreign policy issues.
Material: Organic material, circumstellar material
Galaxy: Parent galaxy, andromeda galaxy, elliptical galaxy, dwarf galaxies, cartwheel galaxy
System: Surveillance system, astrophotography system, solar system, ring system, planetary system

Table 3.18 Some common noun-phrase heads in two collections of documents

I dwell on Wacholder's heuristic at some length because it addresses a problem that is similar to the one that motivates my research: to develop a conceptually simple automatic method for identifying noun phrases in coherent text that are suggestive of a given subject domain. However, the noun phrases produced by the application of her heuristic may or may not be lexicalized. Of course, Table 3.18 lists one noun phrase—*solar system*—that is so strongly lexicalized that native speakers of English can verify its status through introspection. Most of the galaxy types are also lexicalized because they appear in indexes of astronomy terms, and perhaps constitute a local ontology, as Johnston, et al. (1995) would define it. Many other entries in Table 3.18 are three-word compound nouns whose modifier is a noun phrase that would be identified as lexicalized by the measures discussed in this chapter. The lexicalized phrases found in this local syntactic context include *affirmative action*, *corporate welfare*, *child care*, *domestic spending* and *criminal justice*. Other noun phrases in Table 3.18 derive their meaning from the discourse that embeds them. For example, consider the citation of *character issue* in the text fragment shown in Figure 3.2, which is extracted from one of the articles in the political news corpus. Here, *character issue* is an anaphoric element that refers to North's ideologically inconsistent behavior, for which the episode described in the previous sentence is offered as evidence.

Republicans and Democrats alike are widely circulating a piece by Rachel Wildavsky of Reader's Digest titled "Does Oliver North Tell the Truth?" In the article, Wildavsky relates many instances in which North has claimed one thing (such as a close relationship with former President Reagan) and witnesses, most of them conservatives, have reported something else. North's problems on the *character issue* limit his effectiveness when he lobs charges of immorality against his probable Democratic rival…. [Italics mine]

Figure 3.2 A discourse context for *character issue*

It is perhaps not surprising that the output from the application of Wacholder's heuristic, which follows from an insight about the development of topics in coherent text, contains several elements of cohesive discourse that were identified by Halliday and Hasan in their classic work (Halliday and Hasan 1976). Since lexical cohesion is one such element, it is also not surprising that the output from Wacholder's heuristic includes some lexicalized noun phrases, though it admits too much noise to serve my current research goals.

3.6. Toward a computational analysis of local context

To summarize the argument so far in this chapter, I have used methods of traditional linguistic analysis, supplemented with some descriptive statistics, to distinguish lexicalized from syntactic noun phrases in a corpus of texts about engineering. The lexicalized noun phrases of greatest interest are relevant to topics in engineering and may thus be said to constitute the shared vocabulary of a community, not the idiosyncratic expression of isolated speakers or writers. I used the lexicalized noun phrases that are listed in an engineering corpus and cited in the corpus to bootstrap the analysis. Tabulations reveal that the relevant citations consist of adjective-noun and noun-noun sequences, which is consistent with the remarks of Marchand, Levi, Johnston and Pustejovsky, and Justeson and Katz, who observed that lexicalized noun phrases are almost always simplex and short. But despite this apparent simplicity of syntactic form, considerable analytical effort is required to characterize those that can be inserted into another text while preserving their core meaning. The main conclusion is that a lexicalized expression can have no obvious dependencies on a particular context of usage, such as a given speaker's state of mind, state of knowledge, or the relationship of the current topic to other topics under discussion.

But we can learn more about the differences between lexical and syntactic phrases by using the corpus to generalize across contexts. The log-likelihood measure, a simple statistical model of the lexicographer's concept of collocation, can be used to document the extreme frozenness of the lexicalized phrases in an engineering thesaurus, reflecting the fact that lexicalized phrases are word-like in their internal stability and positional mobility. There is no ready explanation for the frozenness of expressions such as *artificial intelligence* or *integrated circuits* because they are formed with words that have the linguistic potential to combine much more freely than has been observed. On the other hand, syntactic phrases—as identified by tests reported in traditional linguistic studies, as well as my own analysis reported in this chapter—are far less fixed. So much can be inferred from the low log-likelihood scores that are computed from syntactic phrases in a large corpus because such phrases often consist of words that are highly frequent and appear in many combinations. Speculating beyond the evidence supplied by the log-likelihood scores, I cited anecdotal evidence suggesting that heavily context-dependent words such as *new* or *small* appear not only in many more noun phrases than *artificial* or *integrated*, but also in a greater variety of morphological and syntactic contexts, which is a symptom of their status as building blocks for productive syntactic phrases, not frozen lexicalized phrases.

Observations like these may highlight an important difference between syntactic and lexicalized noun phrases, but an analysis derived solely from measures of frequency obtained from a corpus is deceptive because it is easily corrupted by sampling errors. The remaining task in this dissertation is to explore the hypothesis that the distinction can be made more reliably by supplementing this distributional evidence with linguistic knowledge that is also found in the corpus. After all, the context surrounding the citation of a noun phrase has much more than the anaphoric or deictic elements that can't survive transplantation. As I argued in this chapter, writers and speakers drop linguistic hints regarding the lexical status of the expressions they employ. Many are found in the local syntactic context, while others may eventually be recovered from a larger sample of

discourse. To achieve analytical rigor, my focus is on the local context of lexicalized-phrase candidates in a corpus of engineering texts, but the evidence that I cited at the end of the previous section suggests that the analysis might extend to other subject domains.

Philosophically, this analysis suggests that writers of many different genres know the conventional names for concepts in their domains of interest and compose their sentences accordingly, constantly fine-tuning their choices about when to retrieve items from their mental lexicons and when to exercise their linguistic creativity. Computationally, the result is a model of the evidence used by the reader to discern the writer's lexical knowledge. The model calculates a score obtained from the relevant linguistic context, which compensates for some of the errors produced by the application of the log-likelihood statistic or other measures of statistical association. One possible interaction of the log-likelihood score and the proposed context score is shown in Table 3.19. For simplicity, this table assumes that the two scores have equal weight in the computational analysis, but I will revisit this issue in Chapter 4.

High log-likelihood score + High context score
The entries in this category are the uncontested lexicalized noun phrases, such as *electrical engineering*, *information retrieval*, and *artificial intelligence*.
Low log-likelihood score + High context score
These are the lexicalized noun phrases that may have a low frequency in the corpus, or consist of words that are frequent in the subject domain and appear in many other domain-specific noun phrases. But these noun phrases appear in lexical contexts. Examples from the engineering corpus include *pattern recognition*, *computational geometry,* and *engineering mechanics*.
High log-likelihood score + Low context score
These are statistical collocations that may represent lexical knowledge of some sort, but are not the lexicalized noun phrases that are of primary interest in this dissertation--i.e., those that may be idioms such as *worst case scenario*, proper names such as *Massachusetts Institute of Technology*, or remnants of boilerplate Web text, such as *News Links Search Contact*.
Low log-likelihood score + Low context score
These are the syntactic phrases, such as *first problem* or *unconvincing argument*, which do not appear in lexical contexts and occur too infrequently or in too many other phrasal combinations to accumulate high association measures.

Table 3.19 Categorizations of noun phrases using two sources of corpus evidence

The design and evaluation of the software that yields the context score is the primary subject of Chapter 4, but the discussion in this chapter can be used to sketch the outline. The problem of distinguishing syntactic from lexicalized noun phrases using linguistic cues from a corpus has two essential properties that can be exploited in a computational study.

First, I have argued that the local syntactic context yields a rich source of clues regarding the status of a lexicalized-phrase candidate. If so, syntactic and lexicalized noun phrases may be distinguished primarily by using knowledge-poor computational methods, which can execute without human intervention or reference to hand-built resources such as databases, knowledge-bases or inference engines. These methods have the considerable advantage of being computable on large stores of data, but perhaps at the risk of sometimes forcing simplistic analyses of linguistic data. Strategies such as partial parsing to extract noun phrases and the extraction of evidence from local context relevant to their classification as syntactic or lexical are knowledge-poor because they use minimal linguistic input. Thus, a partial parser can operate without the syntactic knowledge of complete sentences, and a context extractor can mine the local context of a noun phrase of interest without significant knowledge of the discourse that surrounds it. Most of the software tools that were introduced in Chapter 2 and applied to the engineering corpus to produce the baseline study described in this chapter encode knowledge-poor methods. Of course, when I refer to an engineering thesaurus for a clear set of judgments regarding the vocabulary of engineers, I am introducing an external source of knowledge, but I use it primarily for analysis. Chapter 4 discusses the possibility that the thesaurus may be dispensed with entirely once the software is mature.

A useful framework for understanding my approach to the problem of identifying lexicalized noun phrases is provided by the computer scientist's concept of data mining. In a typical application, algorithms skim large stores of data, looking for a small number of probes where the object of interest is most likely to be found. In my case, the data is a corpus of coherent text, and the probes are the linguistic contexts that often harbor lexicalized noun phrases. If my analysis succeeds at identifying these probes, the result is new linguistic knowledge that heretofore resided only implicitly in the text.

Data mining is the organizing theme of Chapter 4, but we can get an advance peek at the flavor of this work by briefly considering Hearst (1998), a classic and widely cited study of knowledge-poor methods for the automatic acquistion of lexical information. She was interested in mining lexical relations such as hyponomy from stores of coherent English-language text, with the goal of automatically enriching WordNet—a thesaurus-like lexical resource that is widely used in the computational linguistics research community. She found that many hypernym/hyponym relations could be discovered in the context of a small number of fixed phrases. A sentence in this paragraph has one of the productive patterns that she identified, which relates *lexical relations* and *hyponomy* through the fixed phrase *such as* when it appears in the pattern *NP such as NP**. In the engineering corpus, the *such as* pattern connects hypernyms to hyponyms in sentence fragments such as those in Table 3.20.

…useful for *applications* such as *geosciences*, *fluid analysis*, and *medical mapping*.
…the components have different *media types*, such as *audio*, *video*, *image* or *text*.
…the method works for *applications* such as *business graphics*
…in *graphic file formats* such as *GIF* or *TIFF*
…its parameters will be used by *devices* such as *flat-panel displays*
…in *languages* such as *C, C++* and *FoxPro*

Table 3.20 Noun phrases related by *such as* in engineering text

With sample data from an application of Hearst's algorithm in view, I can now consider a second property of the problem of identifying lexicalized noun phrases, as I have defined it, that looms large in a computational analysis. This chapter has argued that coherent text has many observable sources of evidence that are relevant to the classification of a noun phrase as syntactic or lexicalized. But more work must be done to ensure that this evidence is useful for solving the problem.

For example, the citations in Table 3.20 suggest that Hearst's patterns are relevant to my analysis. Lexicographers may argue whether *applications* and *geosciences* encode lexical knowledge that is as stable as the relationship between *languages* and the list of well-known computer programming languages in the last example in Table 3.20. But it is perhaps less controversial to argue that most of the noun phrases following *such as* in the above examples are, in fact, words. This is the logically prior conclusion that Hearst

needed to establish before attempting to mine lexical relations from a corpus of text. Fortunately, a cursory look at the data from the engineering corpus suggests that many noun phrases that participate in her pattern are obviously so, because they consist of single tokens—*video*, *image*—or are the the product of word-formation processes that compress noun phrases into single tokens, as in *TIFF* (*tagged image file format*) and *GIF* (*graphic image format*). Others are listed in conjuncts that feature at least one word or lexicalized noun phrase.

As an example of how my analysis will proceed, the engineering corpus can be consulted to provide several sources of evidence that *medical mapping*, a noun phrase listed in the first line of Table 3.20*,* is lexicalized. First, it has a higher than average log-likelihood score. It is also conjoined with the word *geosciences*, as well as *fluid analysis*, which is listed in the Engineering Information Thesaurus. Finally, if we choose to build on Hearst's analysis, we can observe that *medical mapping* appears in a syntactic pattern that writers commonly enlist to make their knowledge of lexical relations explicit. Put in operational terms, as an ever-larger corpus is investigated, evidence accumulates for the classification of *medical mapping* as lexicalized. In this case, a single citation contains more than one source, but other citations may not be as rich. The result of the computation is a score that reflects a measure of confidence in the classification.

Before this score can be computed, however, the sources of evidence must be evaluated because none of them are foolproof. All of the tests identified by linguistic analysis have exceptions and may not occur frequently enough in the corpus to be useful. The log-likelihood score may be artificially low because of sampling errors. Conjunctions are not always lists of lexicalized noun phrases. Lexical contexts such as *study of* do not always contain stable names of academic disciplines but may instead constitute part of an anaphoric element that does not belong in a dictionary of engineering terms, as in the phrase *study of greatest relevance to the current investigation*. And not all citations containing *such as* encode lexicalized noun phrases, let alone lexical relations. For example, two paragraphs ago, I used in it in the anaphoric expression *sentence fragments such as those in Table 3.20*. Perhaps it is not surprising that the central problem of this dissertation would be cast in these terms, given Levy's concerns

from a quarter of a century ago regarding the lack of categorical evidence for the distinction between syntactic and what she termed ‘complex’ noun phrases, which I discussed in Section 1.2 of Chapter 1.

As I argue in the next chapter, computational techniques developed by computer scientists who study machine learning are well-suited to the task of evaluating and ranking sources of evidence whose reliability is uncertain. If the analysis of the problem is sound, the result may be a conceptually simple, automated method to identify lexicalized noun phrases in a corpus of text that is too large to process by hand, as well as an empirical test of our linguistic intuitions.

CHAPTER 4

A MACHINE-LEARNING STUDY

4.0. Introduction

Chapter 3 introduces the essential elements required to execute a pilot study that algorithmically distinguishes syntactic from lexicalized noun phrases using a large corpus of coherent text, knowledge-poor computational techniques, and exemplars from a dictionary of engineering terms. A human expert knows that *artificial intelligence*, *magnetic resonance*, and *adaptive behavior* are the names of persistent concepts in engineering, while *enough memory*, *considerable work* and *additional parameters* are almost certainly not. The study reported in this chapter proceeds from the hypothesis that the same distinction can be made by a well-informed software program. But before this program can be implemented, I need to identify an appropriate framework for formalizing and testing the intuitions that underlie the arguments I made in the previous chapters.

To establish a starting point for discussion, it is instructive to review a study by Yarowsky (1994), who used a machine-learning algorithm to solve a conceptually similar problem. Yarowsky's goal was to restore accents and diacritics in Romance-language texts that had been stripped of all diacritics. For example, he observed that the corrupted French token *cote* is ambiguous between *côte (coast)* and *coté (side)*. But evidence for the two meanings of *cote* can be recovered from easily computable clues in the local syntactic context of coherent text that is correctly represented. From a list of contexts in a 45-million-word corpus of French newswire text, Yarowsky computed log-likelihood

measures for each distinguishing feature that associated the presence of a diacritic and the feature, which were ranked and represented in a table. Table 4.1 shows some of his results.

Collocation	côte	coté
du cote	0	536
la cote	766	1
un cote	0	716
notre cote	10	70

Table 4.1 Local contexts for *cote* (from Yarowsky 1994: 90)

Once created, Yarowsky's' table was used to make decisions about citations of *cote* observed in the corrupt corpus. For example, when an instance of *cote* is preceded by *notre*, the data in the table provides strong evidence for a representation as *coté,* though this evidence is not strictly categorical. *Coté* is simply the best guess in this context because the other possibility was observed in only 10 of the 80 instances. The data in Table 4.1 suggests that, for the problem of accent restoration in French, some sources of evidence are better than others. Yarowsky tested various algorithms for weighting the evidence to classify the instances of *cote* and concluded that highly accurate classifications could be achieved by considering only the single best source—in this case, the presence of *un* in the immediate left context.

As I design my study, I can take advantage of a resource that was not available seven years ago when Yarowsky conducted his study. Classification algorithms are one valuable result of research in machine learning, which is a mature sub-discipline of computer science. Until recently, machine learning was studied primarily by scholars interested in devising new algorithms, comparing the performance of existing algorithms, or using machine learning to solve problems in computer science. But Witten and Frank (2000) have written a brilliant and useful book that makes this material accessible to scholars in other fields of study. Their book describes the intuitive meaning behind the half-dozen or so machine-learning algorithms commonly used to mine information from

collections of numeric and textual data. The authors also implement the algorithms, standardize the design of the input and output files, and make their software available free of charge from the Web.[8]

The impact of the Witten and Frank book on studies like mine is twofold. First, it removes most of the burden of programming and allows me to focus on issues of linguistics that can be addressed by the application of machine-learning algorithms. Second, it imposes a structure on the task, which consists of four steps:

1. Define the characteristics of the problem that make it suitable as a machine-learning application.
2. Identify a set of attributes that can be subjected to automated study.
3. Run the training phase.
4. Run the test phase and evaluate the results with human judges.

These steps dictate the organization that I follow in the rest of this chapter.

4.1. Computational lexicography as a machine-learning application

Machine-learning algorithms are guided by characteristics of data analyzed with human effort to make decisions about unknown data. This strategy can be productively employed to study many linguistic problems. Indeed, Yarowsky argued that Romance-language accent restoration is one instance in a general class of problems involving lexical ambiguity, all of which require the analysis of local syntactic context to identify cues that reveal a word's meaning. For example, if the word *bank* is used in the sense of *travel with one side higher than another*, the local context is more likely to have words that mention plausible modes of transportation, such as *car* or *plane*, as well as syntactic evidence showing that *bank* is a verb. However, if the sense of *bank* is *financial institution*, the local context should contain words from the semantic field containing the words *teller*, *money* or *finance*. A machine-learning algorithm can be 'trained' to recognize the difference between the two senses of *bank* if it is first provided with text in which all instances of *bank* are tagged by human experts with the correct sense. The

[8] Accessible at: <http://www.cs.waikato.ac.nz/ml/weka/>

algorithm tallies the features found in the contexts of $[\text{bank}]_{\text{finance}}$ and $[\text{bank}]_{\text{travel}}$, which enables it to identify the senses of *bank* in test data that has not been annotated. For example, a given instance of *bank* in a test corpus is tagged with the *finance* sense if its local context has many features associated with discussions of financial institutions that were identified in the training data. Word-sense disambiguation is a well-studied problem that raises many issues, including the identification of the classifying features, the analysis of the ambiguous word's linguistic environment, the evaluation of suitable classification algorithms, as well as the definition of *word sense* itself, which is problematic to some philosophers and lexicographers (Kilgariff 1997). A good summary of recent progress can be found in a special issue of *Computational Linguistics* devoted to the problem (Ide and Veronis 1998).

The problem of distinguishing lexicalized from syntactic phrases is perhaps near the outer edge of Yarowsky's class of lexical ambiguity problems because it is primarily an ambiguity of lexical status, not meaning: the task is to determine whether a phrase is a persistent name, or an incidental description. But when a syntactic phrase becomes lexicalized, it is capable of acquiring an ambiguity of meaning. Not only does it have the meaning that can always be computed from the composition of the meanings of the component words, but it may also have a non-predictable meaning that arises from its repeated use as a name for a persistent concept. And the local context of the phrase in coherent text may harbor some clues regarding the difference. Thus it is reasonable that Xhai's study of 'lexical atoms' (Xhai 1997) that I discussed in Section 1.3 of Chapter 1 defines the problem as one of lexical ambiguity. If *white house* appears in a text that is primarily about houses, the local context should contain many other words from the same semantic field, or realm of experience, such as *paint*, *neighborhood*, *houses* and *fence*. But if *white house* refers to the mansion at 1600 Pennsylvania Avenue in Washington, D.C., where the president of the United States lives, the local context probably contains more words from the domain of politics.

As I argued in Chapter 1, however, Xhai's analysis is theoretically unsatisfying. It under-specifies the set of phrases that have been lexicalized because only a relative few have diverged so far from their compositional meanings that sense distinctions can be

recovered from the surrounding text. In the study that is the focus of this chapter, I investigate the hypothesis that what is essentially a set of metalinguistic cues can be exploited to make the same distinction because speakers and writers know the conventional names for important concepts in their language community and construct their sentences accordingly. After all, in Xhai's primary example, a metalinguistic cue—capitalization—is sufficient to distinguish the senses of *white house* in a given discourse, so a further investigation of the phrase's linguistic context is superfluous.

How does the problem of distinguishing lexicalized from syntactic phrases compare to word-sense disambiguation cast as a machine-learning problem? The computational effort in a word-sense disambiguation project is preceded by intellectual analysis that identifies two—or, more typically, multiple—senses, usually with reference to a lexical knowledge base such as a dictionary or WordNet, which serves as an objective source of judgments. My problem is arguably simpler because only a binary decision is required. In the training phase, the metalinguistic cues that feed the classification algorithm are identified by examining the places in the corpus where entries in the reference lexicon can be found. For example, the noun phrase that serves as the object of *department of* is identified as a cue because entries from engineering thesauri frequently appear in this context, perhaps because names of disciplines are listed in a specialized thesaurus and count among the most frequent and stable multi-word names in a corpus of academic text. Unlike the word-sense disambiguation task, the training phase is not technically distinct from the intellectual analysis phase because the cues are selected by hand, though automation of this step is possible, in principle. The cues are then fine-tuned and the optimal classification algorithm is selected, using the guiding principles of simplicity and parsimony. In the test phase, new lexicalized noun-phrase candidates are classified, in a procedure that is analogous to the one used in the word-sense disambiguation task.

Intuitively, this test encodes the hypothesis that other noun phrases appearing in the same contexts as attested lexical phrases may also be lexicalized. For example, many engineering thesauri list the phrase *electrical engineering*, which frequently appears in the context of *department of* in the engineering corpus. So does *radiation physics*, which

is not listed in an engineering thesaurus but may be accidentally missing from a lexical resource that is maintained by human effort. If the classification algorithm selects *radiation physics* as a lexicalized noun phrase and human judges agree with the classification, then we can say that the algorithm 'learned' how to make the correct categorization from the input cues and the corpus. As a result, *radiation physics* is identified as the name of a concept that may be added to a future edition of the engineering thesaurus.

There is one important difference between this study and the typical word-sense disambiguation experiment. In the test phase of a word-sense disambiguation task, each corpus citation of the ambiguous word is considered separately. For example, the cues in the local context of the first citation of *bank* in a corpus of, say, newswire text, might suggest the financial reading because they appear in a story about the stock market, while the third citation is tagged with the sense *ground near a river* because its local context has lexical cues that are typically found in stories about floods. In the test phase of my experiment, however, all citations of a given phrase are assumed to have the same classification because the corpus is restricted to a single subject domain. If so, evidence for the categorization of a given lexicalized noun-phrase candidate can accumulate throughout the corpus. Thus if an unclassified phrase such as *radiation physics* appears in the context of *professor of* and other attributes discussed in the next section, in addition to *department of*, the classification algorithm can be even more confident that it is a lexicalized, rather than a syntactic phrase

4.2. The identification of attributes

The machine-learning algorithms in the Witten and Frank implementation require a set of attributes, features, or cues, that can be represented in a standard format, which they call an ARFF file (Witten and Frank 2000: 49-50). Once the data is in this form, it is possible to perform many experiments with the Witten-Frank software library. A sample ARFF file with realistic, but hypothetical, linguistic data is shown in Figure 4.1.

A file like this is used twice: once to train the classification algorithm with exemplars whose status is known, and again to classify unknown instances. Figure 4.1 represents a file that might be used in a training run.

```
 1  @relation LexicalStatus
 2   % 1
 3  @attribute department-of real
 4  % 2
 5  @attribute definition-of real
 6  % 3
 7  @attribute topics-in real
 8  % 4
 9  @attribute very: real
10  % 5
11 @attribute compoundNounModifier: real
12 % 6
13 @attribute lexicalized {yes, no}
14 @data
15  % Phrase                          Attribute value
16  %-------------------------------------------------------------------------------------------------------
17  %                                 1       2       3       4       5       6
18  %-------------------------------------------------------------------------------------------------------
19  % Magnetic resonance
20                                    0,      0,      12,     0,      67,     yes
21  % Artificial intelligence
22                                    0,      2,      52,     0,      171,    yes
21  % Certain browsers
22                                    0,      0,      0,      0,      0,      no
23  % Abundant compounds
24                                    0,      0,      0,      40,     0,      no
23  % Civil engineering
24                                    59,     0,      0,      383,    2,      yes
25  % Remote sensing
26                                    0,      2,      97,     0,      207,    yes
25  % Molecular beam epitaxy
26                                    0,      0,      10,     0,      1,      yes
27  % High production
28                                    0,      0,      0,      7,      2,      no
```

Figure 4.1 A sample ARFF file containing classified noun phrases

Except for the line numbers, which I have added here to aid exposition, Figure 4.1 contains the literal text of an ARFF file that can be submitted to the Witten-Frank software package. Lines that start with the '*%*' symbol are comments. Otherwise, the file has two logical parts: a list of attributes (lines 1-13, including comments), and a list of lexicalized phrase candidates to be classified (lines 19-28). The lines that describe the attributes have three parts: the keyword *@attribute*; the name of the attribute, such as

compoundNounModifier; and the type of data, which is a real number for every attribute, except the final one. The list of attributes includes one—*very*—that is more often associated with syntactic phrases than with lexicalized phrases, and four that are usually associated with lexicalized phrases: *department-of*, *topics-in*, *definition-of* and *compound-noun modifier*. Since this ARFF file is used for training the classification algorithm, the attribute *lexicalized* supplies the correct answer key. Lines 19-28 show eight lexicalized- phrase candidates and a list of numbers that represent the raw counts of the citations in the corpus containing the phrase and the attribute in the appropriate syntactic relationship. These lines show that *magnetic resonance*, *artificial intelligence* and *molecular beam epitaxy* co-occur more frequently with the attributes that are usually associated with lexicalized rather than syntactic phrases, which would be predicted by my analysis because these phrases are cited in an engineering thesaurus; the converse is true for *certain browsers*, *high production* and *abundant compounds.*

In Figure 4.1, the input data is of two kinds: numeric and nominal. The numeric data records the counts of citations in the corpus in which the lexicalized phrase candidate and the attribute co-occur in the correct syntactic relationship. The nominal variable *lexicalized?* identifies membership in a category. Since numeric and nominal data can be intermixed in the same ARFF file, the input data can be heterogeneous and thus may also consist of different kinds of numeric measures. In Section 3.6 of Chapter 3, I suggested that a heuristic for distinguishing syntactic from lexicalized phrases for identifying cues in the local context could supplement the log-likelihood measure, which uses different kinds of corpus evidence to identify collocations and may sometimes be accidentally wrong in ways that can be compensated. If the fine-tuning process reveals that log-likelihood increases the accuracy of the classification algorithm, this measure can be added to the ARFF file as an additional attribute, and the result is still conceptually simple. For many problems in computational linguistics, the Witten-Frank software provides relatively few constraints, as long as the input can be represented in the ARFF file format.

The data in this small file raises issues that will be discussed more thoroughly in the next two sections of this chapter. For example, most of the attributes do not categorically identify lexicalized or syntactic phrases—but, as in Yarowsky's study, are only more-or-less correlated with one of the classifications. The co-occurrence of *very* with syntactic phrases may be so high as to be essentially absolute, but its value as an attribute might be limited if it occurs relatively rarely in the data. The resolution of issues like these is part of a fine-tuning process that might result in a startlingly simple, linguistically motivated, data-tested procedure for distinguishing lexicalized from syntactic phrases. Or, since the test is falsifiable, the exercise could reveal that further analysis is required to make the automated classification of noun phrases truly effective.

The next three sub-sections discuss the identification of attributes and describe the procedures for constructing the ARFF file using the software that I discussed in Chapter 2 and input from the engineering corpus and an engineering thesaurus. To make the task manageble, I focus my effort on a subset of the engineering corpus. Approximately two-hundred megabytes, or the first six partitions, are used for analysis and training of the classification algorithm; the seventh partition is used for testing. An important sub-goal of the analysis in the first stage is an assessment of the six partitions of the training corpus to determine whether the attributes are commonly observed and more-or-less evenly distributed. Accordingly, most of the tables in this section show separate tabulations for each partition. If the attributes pass these strict tests, I can be reasonably confident that they appear in the same proportions in the unseen partition that represents the test corpus.

4.2.1. Lexical attributes

Section 3.4 of Chapter 3 argues that writers drop frequent lexical hints that their word choice is not always their own invention. To conduct a computational study of this observation, I restrict my attention to prepositional-phrase complements such as *journal of applied physics* or *bibliography on computer-aided vision*. Examples like these are common in a corpus of engineering text. Moreover, the syntactic relationship between the noun phrases to be classified and the lexical cues, which are underlined in these

examples, can be identified with a reasonable degree of accuracy using the shallow parsing techniqes described in Chapter 2. I ignore other potentially robust lexical cues, such as noun-phrase/acronym pairs, primarily because their identification requires more complex processing than is feasible in this pilot study. For example, in coherent text, the acronym often appears after the lexicalized phrase in parentheses, as in *The U.S. Geological Survey (USGS)*, or in an apposition delimited by commas, as in *Total Quality Management, or TQM*; sometimes the acronym appears at a difficult-to-specify earlier or later point in the discourse. At any rate, since there is no consistent syntactic relationship between the acronym and the noun phrase of interest, the study of acronyms is best left to a more specialized investigation, which would take possible discourse cues into account that are beyond the scope of this study. Bowen, et al. (1996) describes an algorithm that solves part of this problem.

Given these restrictions, lexical attributes are identified in two ways, depending on whether they are positive or negative cues for lexicalized phrases. To identify positive attributes, I need a list of engineering terms that are cited in the engineering corpus, as I discussed in Sections 3.2 and 3.3 of Chapter 3. To execute the study described in this chapter, I obtained a sample of 5,000 common engineering terms that are listed in the Engineering Information Thesaurus (Milstead, et al. 1995) and other subject indexes such as the Dewey Decimal Classification (Mitchell, et al. 19996),[9] which I refer to as the *engineering index* in subsequent discussion. Single words as well as phrases appear in the sample, and I use all of them, on the assumption that since entries in a lexical resource are words that name persistent concepts, there should be no difference in the status of citations such as *aerodynamics* or *aerospace engineering* as persistent names for important concepts in this subject domain. A software program that collects keyword-in-context, or KWIC, citations identified approximately 3,000 sentences from a 50-

[9] To ensure success in the analysis, I needed a sample of engineering vocabulary that was commonly used and generic. This information is readily accessible in the Relative Index portion of Dewey Decimal Classification (Mitchell, et al. 1996) that is devoted to engineering topics and shows a high degree of overlap with the Engineering Information Thesaurus. Permission to use a machine-readable copy of the Dewey Decimal Classification is gratefully acknowledged.

megabyte sample of the engineering corpus that contained the engineering index entries. I inspected these citations to find the patterns of interest, creating the so-called positive lexical cues or attributes.

Negative evidence is more difficult to obtain in an empirical study like this one because I don't have access to intuitions, grammaticality judgments, or a list of syntactic phrases that matches the authority of thesaurus citations for lexicalized phrases. Thus, the syntactic phrases that I use in the analysis portion of this test are a hypothetical subset of the non-lexicalized noun phrases that are the common reflex of syntactic creativity, as I discussed in Section 1.2.1 of Chapter 1. I created the hypothetical syntactic noun phrases from bigrams that consist of nouns modified by context-dependent adjectives, which I identified from output of the noun-phrase parser on the first partition using the criteria listed in Table 3.10 in Section 3.3 of Chapter 3. Table 4.2 shows the flavor of this data, listing the 50 most frequent context-dependent adjectives and the 50 most frequent bigrams formed from these words. Though none of the noun phrases listed in the bottom of Table 4.2 are names of concepts in engineering that would be listed in a thesaurus, this list has a few bigrams that achieve high log-likelihood scores because they are frequent and relatively frozen, such as *first name*, which have a lexicalized status in other domains of interest. More examples are listed in Table 3.11 of Chapter 3.

Context-dependent adjectives

first	certain	considerabl	favorite
different	actual	diverse	false
additional	excellent	corresponding	fifth
full	comprehensive	adequate	consistent
new	entire	acceptable	adjacent
basic	third	enough	explicit
second	extensive	exciting	expensive
appropriate	big	biggest	fascinating
complete	conventional	frequent	convenient
further	accurate	fourth	excessive
available	exact	ethical	crucial
final	essential	extreme	amazing
complex	extra		

Some noun phrases with context-dependent adjectives

further information	first prototype	different ways	complete line
additional information	first half	first place	first meeting
full text	second prototype	full details	basic information
full coverage	complete list	first amendment	fourth quarter

first time	first page	full name	final rule
further details	second edition	first day	first week
different types	first name	basic principles	final manuscript
full range	basic research	second half	additonal support
first step	third quarter	full papers	full spectrum
final report	first year	ethical issues	full list
complex systems	first line	complete listing	comprehensive environment
full time	first quarter	first part	complete manuscript
comprehensive range	extensive use		

Table 4.2 The 50 most frequent context-dependent adjectives and noun-phrase bigrams

The negative lexical cues were identified from the linguistic studies that I reviewed in Chapters 1 and 3 and were subjected to the same constraints of computability that restrict the positive cues. Two lexical cues for syntactic phrases best reflect the spirit of Levy's analysis and are easily computable: the noun-phrase contexts of *–ly* adverbs and *very*. Noun phrases from the engineering corpus that are modified by *–ly* adverbs include *extremely rich collection*, *electronically submitted articles*, *badly referenced nodes*, *incredibly short time*, *fairly general setting* and *frequently used commands*. Noun phrases from the engineering corpus that appear in the syntactic context of *very* include *very first article*, *very complex interface*, *very high speed systems*, *very aggressive drive*, *very well equipped laboratories*, *very graphic presentation* and *very high degree*.

The lexical cues must pass two initial checks: they must be relatively frequent and evenly distributed across the corpus. If they are too infrequent, their value as attributes is severely limited, and they will be automatically eliminated in Step 3 of this study when the classification algorithm is fine-tuned. If they are not evenly distributed, we can't be confident that similar topics are discussed in all partitions, or that the cues represent the common idiom of the engineering community instead of the idiosyncratic language of a single writer or institution. A simple frequency tabulation resolves both doubts.

Table 4.3 lists the most frequent positive lexical cues in the six partitions of the training corpus. The tabulation shows the 15 most common lexical cues in the first partition; 88% of these cues also rank among the top 15 in the remaining partitions and are similar in frequency. Not surprisingly, the most frequent lexical cues in all partitions are fragments of proper names, many of which are built from common nouns; for example, the objects of *center for* include *scientific computing*, *information law and policy*, *automation research*, and *x-ray lithography*. Many lexical cues that may or may

not form proper names are also highly ranked in all partitions; examples include *conference on* and *research in*, which have noun-phrase objects such as *software engineering*, *machine learning*, *discrete mathematics*, *ecosystems and sustainable development*, *computational geophysics*, *magnetic levitation*, *non chlorine-based bleaching*, *science and engineering*, and *theoretical optics*.

	Count-Rank in partition					
Attribute	1	2	3	4	5	6
Department-of	2,425-1	2,426-1	2,091-1	1,910-1	2,348-1	2,258-1
Center-of	736-2	634-5	699-4	612-6	827-4	674-3
Journal-of	711-3	640-4	957-2	833-2	1,169-2	667-4
Institute-of	709-4	833-2	751-3	769-3	908-3	883-2
Conference-of	705-5	579-7	561-7	744-4	768-5	589-6
School-of	696-6	702-3	617-5	630-5	668-6	554-8
Development-of	619-7	585-6	572-6	566-7	541-7	573-7
College-of	517-8	413-9	357-11	440-9	444-9	641-5
Research-in	450-9	447-8	422-8	498-8	482-8	428-10
Society-for	370-10	275-17	289-15	368-12	349-14	317-4
Control-of	335-11	409-10	383-9	335-14	391-10	479-9
Design-of	314-12	285-15	300-13	403-10	385-11	357-12
Application-of	305-13	364-11	257-17	280-17	351-12	334-13
Aspects-of	302-14	306-13	296-14	338-13	350-13	289-17
Association-of	288-15	157-29	184-26	211-24	196-26	166-28

Table 4.3 Distribution of positive lexical cues across the training portion of the corpus

Table 4.4 shows the tabulation of the negative lexical cues across the six partitions of the training corpus. Both negative cues are evenly distributed and nearly as frequent as the most frequent positive cues, though the overall frequency of the syntactic cues is swamped by the lexical cues because only two have been identified. The relatively low absolute frequencies of *–ly* adverbs may be due to the fact that they function more frequently as predicative or preverbal modifiers; examples from the engineering corpus include *...are continuously adjustable*, *...arrived at empirically*, and *...you can easily extend it*.

	Count in partition					
Attribute	1	2	3	4	5	6
very	683	645	592	797	403	550
-ly adverbs	512	531	620	573	489	584

Table 4.4 Distribution of negative lexical cues across the training portion of the corpus

The lexical cues must also be checked to determine whether they generally co-occur with phrases of the appropriate classification. In other words, do the cues that identify syntactic phrases occur primarily with syntactic phrases, and vice-versa? One way to answer this question is to calculate the log-likelihoods of all noun phrases that occur in these contexts. This test is consistent with the widely held belief that sequences of words in lexicalized phrases are relatively frozen and score high on this measure because they represent lexical collocations, as I discussed in Section 1.1.2 of Chapter 1 and Section 2.2.3 of Chapter 2. Of course, this test may sometimes err, but the results represent a valuable rough-cut empirical confirmation of a major hypothesis in this study. Table 4.5 shows that the average log-likelihood of the noun phrases in the contexts of lexical cues is higher than those appearing in the contexts of syntactic cues.

It is possible to object that the high log-likelihood score for lexical contexts is due to the high frequency of *department-of*, which forms many proper names. However, since more than half of the objects of *department-of* are single terms—such as *energy*, *agriculture*, *defense* and *mathematics*, which are not counted in the log-likelihood calculation—this context is not an extreme outlier. A worse problem is that the log-likelihoods of the three classes are not directly comparable because the raw frequencies differ dramatically. The bottom row of Table 4.5 shows the log-likelihood of the noun phrases in one lexical context, *conference-on*, which does not primarily form proper names and closely matches the frequency of *very;* it is still three to four times higher.

	Log-likelihood in partition					
Noun-phrase class	1	2	3	4	5	6
All noun phrases	58.85	57.19	75.65	57.89	61.02	56.36
In lexical contexts	136.76	144.10	124.17	124.77	141.23	125.71
In syntactic contexts	45.79	39.28	60.42	60.86	53.77	43.43
Very	48.21	37.72	63.44	66.12	50.18	44.89
Conference-on	182.41	158.25	180.14	255.21	191.98	220.30

Table 4.5 Average log-likelihoods of noun phrases in lexical and non-lexical contexts

Another way to determine whether the attributes correspond to the correct classifications of the noun phrases is to count the occurrences of lexicalized and syntactic phrases in lexical and syntactic contexts. In other words, do syntactic phrases such as *first step* occur only in syntactic contexts such as *very first step*? And do lexicalized phrases such as *artificial intelligence* occur only in lexical contexts such as *conference-on*? Or can phrases such as *very artificial intelligence* or *conference on first steps* be extracted from the corpus? To make a fair comparison, I consider the results only for *very* and *conference-on*. The noun-phrase samples were balanced by counting half of the lexicalized phrases that matched in the engineering corpus, as well as the engineering index; one third of the hypothetical syntactic phrases were counted. Both samples were obtained by selecting every second or third item, respectively, from an alphabetized list of unique items.

Table 4.6 shows an aggregate count of the relevant data in all six partitions. The proportions of the counts in the contexts and the total counts reflect the fact that the contexts robustly identify the two classes of noun phrases; for example, 796/2,662, or nearly 30%, of the noun phrases that appear in the context of *conference-on* are listed in the engineering index. The distribution is so unbalanced that it is reasonable to wonder why there are any cross-category tabulations at all. Lexicalized phrases appear in syntactic contexts because there are a few bona-fide examples of lexicalized noun phrases that can be modified by *very,* such as *very large scale integration*. In other cases, *very* is used to mark an anaphoric element as in …*the very radiation hazards we had*

anticipated; the underlined phrase appears in the engineering index. The syntactic phrases that appear in lexical contexts reflect the fact that lexicalized noun phrases may sometimes be formed from context-dependent adjectives, as I have defined them. For example, the engineering corpus has several citations of *conference on complex fluids*.

	Syntactic contexts (*very*)	Lexical contexts (*conference-on*)	Total counts of phrase categories
Syntactic phrases	956	12	8,977
Lexicalized phrases	42	796	9,304
Total counts of the two lexical contexts	3,760	2,662	

Table 4.6 Cross-tabulations of noun phrases and contexts

Before leaving the discussion of lexical cues, I need to describe the procedure for identifying noun phrases in the corpus in lexical and syntactic contexts, partly for the sake of completeness and partly because some of the software components are also used by the processes that identify the syntactic cues that I discuss in the next subsection. The process is depicted in Figure 4.2. The text chunker first divides the corpus into sentence-and-clause-delimited segments, using end-of-clause punctuation marks as a cue. The keyword-in-context (KWIC) identifier has access to the list of lexical cues, the lexicalized-phrase candidates that were obtained from the corpus using the procedure summarized in Figure 2.7 in Section 2.3 of Chapter 2, and the list of noun phrases found in the corpus that are listed in the engineering index. With this information, the KWIC identifier reads in one sentence at a time and looks for lexical cues. If a cue is found, it makes a guess at the left and right boundaries of the cue's immediate syntactic context and produces a line in a file that lists the cue and the context. The context contains the lexicalized-phrase candidates that are the focus of further analysis; these are submitted to the part-of-speech (POS) tagger and the noun-phrase parser that I described in Section 2.2 of Chapter 2.

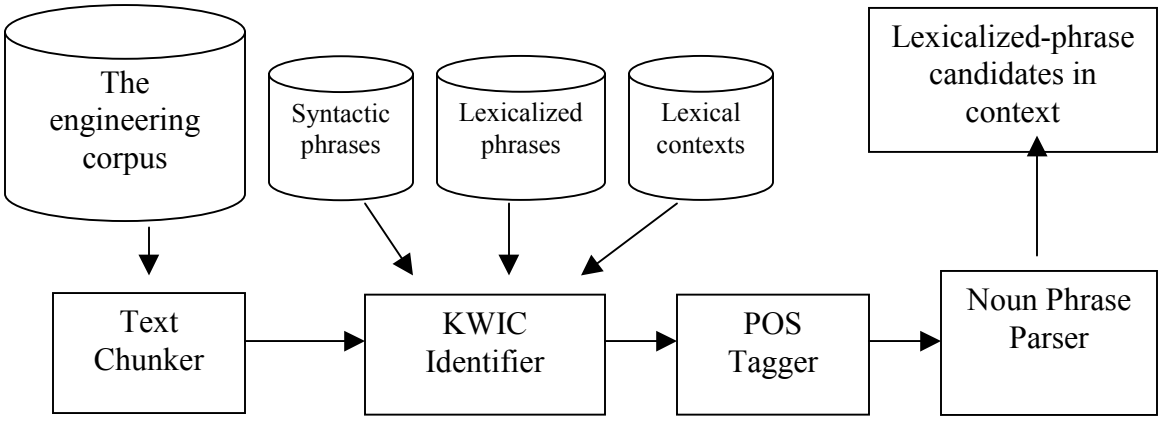

Figure 4.2 Process flow for identifying noun phrases in selected lexical contexts

4.2.2. Syntactic attributes

In Section 3.4.3 of Chapter 3, I argue that the existence of syntactic cues for lexicalized noun phrases in coherent text permits me to generalize the psycholinguistic claims that underlie my analysis beyond a highly specialized corpus of engineering text, which may exhibit peculiar linguistic conventions. Such cues also present an opportunity to pose important questions for computational analysis. For example, how robust are syntactic cues relative to lexical cues for making the distinction between syntactic and lexicalized noun phrases? If the two generic syntactic cues that I have identified are highly reliable, lexicalized noun phrases could be extracted with relatively little effort from a large corpus because there would be little or no need to analyze the language of the text to discover patterns of lexical choice that may vary across subject domains.

4.2.2.1. Conjunctions

As I argued in Section 3.4.3 of Chapter 3, the logic of using conjunctions of noun phrases as a source of evidence for lexical status derives from the hypothesis that, if one element in the list is a single word or a lexicalized phrase, the other elements probably are, too. This hypothesis can be tested on a large corpus using shallow parsing techniques with the process flow summarized in Figure 4.3.

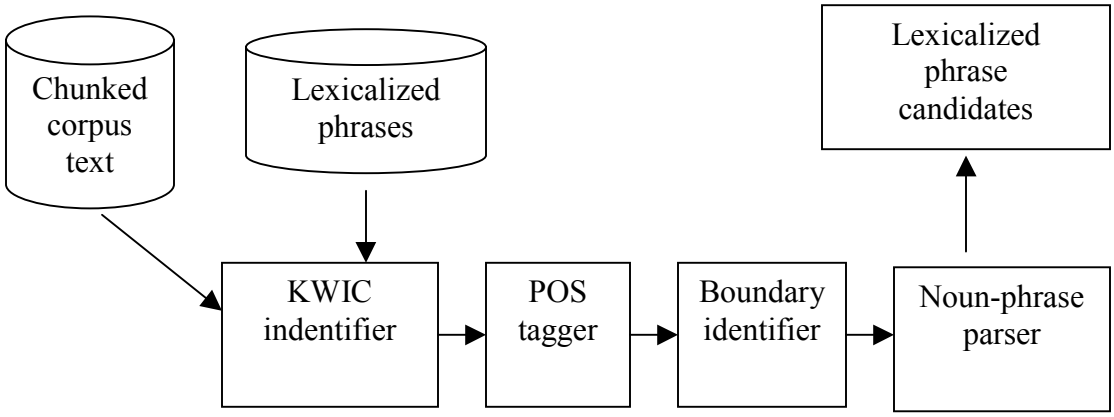

Figure 4.3 Process flow for identifying noun phrases in conjunctions

The KWIC identifier, which is also used to identify the words and phrases in lexical contexts described Section 4.2.1, isolates the sub-corpus of interest for more resource-intensive processes. A chunk of corpus text is selected for further analysis if it contains a conjunction and an entry in the engineering index. But after the first stage of the process depicted in Figure 4.3, there is no guarantee that the dictionary entry is one of the conjuncts, or even that the conjunct is a list of noun phrases. For example, the following sentence is spuriously identified as a candidate because it contains a conjunction and the engineering index entry *information retrieval*: 'A new paradigm for *information retrieval* is based on faceted classification and indexing.' Only examples like those shown in Figure 4.4 are submitted to further analysis.

Key	Sentence
Computer-aided design	The program can be used to aid grid generalization and computer-aided design.
Neural networks	The AI Software Packages section includes subdirectories for artificial life and complex adaptive systems, distributed AI, genetic programming, neural networks, and many others.

Figure 4.4 Conjunctions with lexicalized noun phrases

The conjuncts of interest are identified by locating the boundaries of the conjunction and verifying that it is a list of noun phrases, one of which is a key from the engineering index. This procedure requires part-of-speech-tagged input. Each conjunct in the sentence is located and a pointer first moves leftward, and then rightward, to identify the edges. A binary decision is made for each tagged token to determine whether it can legally appear inside a simplex noun phrase. Admissible tags are *noun*, *adjective*, *determiner*, *comma*, and *conjunction*. The traversal stops when a non-admissible tag is encountered and a boundary is identified. For example, in the second sentence cited in Figure 4.4, one outcome is the fragment *artificial life and complex adaptive systems, distributed AI, genetic programming, neural networks and*, which is flanked on the left by the preposition *for* and on the right by the quantifier *many*. After this fragment is passed through the noun-phrase recognizer, the noun phrases *artificial life*, *complex adaptive systems, distributed AI, genetic programming* and *neural networks* are obtained. When the single terms and dictionary keys are eliminated from the list, the result is the set of noun phrases whose lexical status is to be assigned by the classification algorithm.

This procedure works for a superficial treatment of conjunctions in a large corpus, but much more analysis can be done, especially on reduced conjuncts. The major reason for considering reduced conjuncts is that a proper analysis would increase the number of observations. For example, the process depicted in Figure 4.3 would identify *information*

storage and *retrieval* from the conjoined phrase *information storage and retrieval*, but *information retrieval* is missed, which is potentially damaging to the goals of my study because this phrase is an entry in the engineering index.

The design of a sophisticated conjunction-reduction parser is a non-trivial task, but a simple heuristic that works for two-word cases shows how a corpus-aware solution might look. The procedure starts from part-of-speech-tagged input and examines the two words that immediately flank the conjunction. If their part-of-speech tags match, as they do in the above example because both *storage* and *retrieval* are nouns, the phrase is a conjunction-reduction candidate. The remaining word in the longer half of the conjunction, *information*, is then paired with the word in the shorter half, *retrieval*, and the phrase *information retrieval* is checked against an external list, such as the engineering index, or the noun phrases already collected from a first-pass look at the corpus. If the constructed phrase is found, it is entered into the analysis as though it were a full conjunct. More sophisticated heuristics that work on longer noun phrases require input from a module that assigns internal structure, such as the one I discussed in Section 2.2.4 of Chapter 2 and put to use in the next sub-section, but these interesting and important issues deserve a separate study.

Table 4.7 shows the frequency distributions of unique noun phrases that appear in conjunctions, with and without co-occurring attested lexicalized phrases, for all six partitions. Tabulations of noun phrases that co-occur with the hypthetical syntactic phrases are not shown because the counts are are too small to be meaningful without access to a conjunction-reduction parser. Since syntactic phrases are formed from adjectives that combine freely, the engineering corpus contains many examples of reduced conjuncts with adjective remnants, as in the sentence *This problem…demands development of efficient and accurate algorithms*. The underlined expression is a putative syntactic noun phrase, but it is conjoined with the adjective *efficient*, which would not show up in a count because it is eliminated by the noun-phrase parser as ill-formed. Interestingly, lexicalized phrases are far less affected by this limitation, despite the fact that they may also be realized as adjective-noun bigrams. This observation, as

well as the last line in Table 4.7 showing that only 8 to 9% of all noun phrase tokens are conjoined with entries in the engineering index, suggests that the conjunction context is a potentially powerful selector for lexicalized noun phrases. The counts in this line can be interpreted as a noisy estimate of the lexicalized noun phrases that remain to be discovered.

	Count in partition					
Noun-phrase (NP) location	1	2	3	4	5	6
All NPs	299,883	285,046	287,514	280,626	299,242	296,374
All NPs in conjunctions	68,220	66,549	68,613	72,766	70,776	68,070
NPs conjoined with lexicalized NPs	27,121	24,413	25,854	27,761	26,729	25,362

Table 4.7 Frequencies of unique conjoined noun phrases in six partitions of the engineering corpus

Table 4.8 shows the log-likelihoods for noun phrases in conjunctions in the six partitions of the training portion of the engineering corpus. The subset of the noun phrases conjoined with lexicalized noun phrases has slightly higher log-likelihoods than noun phrases in unfiltered conjunctions, but the log-likelihoods for conjoined noun phrases are generally lower than the average log-likelihood score for all noun phrases. But the log-likelihood score for the list of all noun phrases may be artificially high, in part because it is contaminated with idioms peculiar to Web pages, such as *yellow mountain institute home page*. In the first partition, *home page* appears in nearly 200 noun phrases, has a log-likelihood of 14,181, and does not appear in conjoined contexts. A separate calculation on noun-phrase bigrams reduces the influence of these irrelevant phrases and shows the predicted pattern. This calculation also permits a comparison with the hypothetical syntactic noun phrases; bigrams conjoined with lexicalized noun phrases have log-likelihoods that are approximately four times higher.

Noun-phrase (NP) category	Log-likelihood in partition 1	2	3	4	5	6
All NPs	58.85	57.19	75.65	57.89	61.02	56.36
NPs in conjunctions	53.28	54.18	57.93	55.74	57.11	53.97
NPs conjoined with lexicalized NPs	57.43	60.13	66.60	60.05	60.01	63.09
All NP bigrams	10.13	10.65	10.89	10.87	10.77	10.60
All conjoined NP bigrams	26.18	20.12	20.55	19.68	20.29	20.08
All NP bigrams conjoined with lexicalized NPs	30.11	32.30	32.02	30.17	32.09	31.05
Syntactic bigrams	6.59	8.93	9.04	9.02	8.85	8.84

Table 4.8 Log-likelihoods of conjoined noun phrases

4.2.2.2. Phrase structure

Section 3.4 of Chapter 3 argues that lexicalized phrases often appear in the modifier portion of a compound noun, as in the phrases [[accident investigation] report], [[alternative fuel] program], [[architectural engineering] journal], and [[automatic control] theory]. The underlined expressions represent matches in the engineering index and the brackets show the hierarchical structure of the pharases, as identified from corpus evidence. The major components in the procedure for assigning internal structure to noun phrases are depicted in Figure 4.5. A candidate list of noun phrases is obtained by extracting the master list of noun phrases on the corpus, using a configuration that admits only simplex noun phrases with no embedded prepositions or conjunctions. This list is supplied to a module that uses a variation of Lapata's (1999) algorithm for assigning internal structure using a dictionary and corpus evidence, as I discussed in Section 2.1.4 of Chapter 2. In a noun phrase such as *alternative fuel program*, *fuel* is bracketed with *alternative* if the phrase *alternative fuel* is in the dictionary, or has a higher log-likelihood than *fuel program*. In addition, the outer boundaries are checked for completeness. If corpus evidence suggests that *alternative* usually appears with a modifier, or if *fuel*

usually modifies another noun, the phrase is rejected. Such rejections are frequently encountered in lists of lightly parsed Web text of uncertain quality. Examples from the engineering corpus include *measurement] [frequency range]*, which may consist of incomplete remnants of two noun phrases; and *[picture [home*, which is presumably missing the head noun *page*. Only correctly parsed noun phrases are submitted to further analysis, and from these, the heads and modifiers are extracted and tabulated.

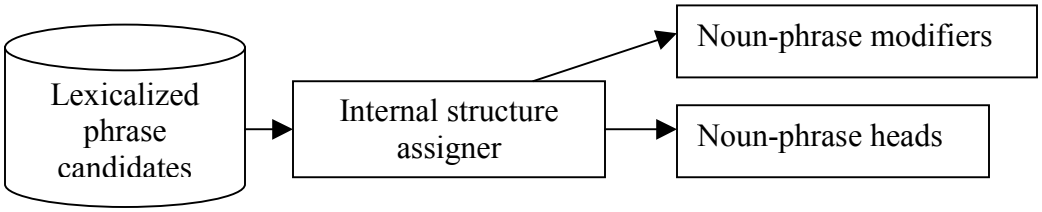

Figure 4.5 Process flow for identifying noun-phrase structure

Tables 4.9 and 4.10 show the frequency tabulations and log-likelihood scores of unique noun-phrase heads and modifiers. Since over 90% of the heads and modifiers are bigrams, only these tabulations are shown, which has the advantage of facilitating a comparison with the data from conjunction contexts that I discussed in the previous subsection. Like conjunctions with lexicalized noun phrases, a correctly defined syntactic position in a compound noun can be interpreted as a filter that isolates noun phrases that are relatively frozen; the log-likelihoods of noun-phrase modifiers are nearly 2.5 times higher than the baseline. Since the similarity of the data for noun-phrase heads and modifiers implies that the head position of a compound noun is also an important location for lexicalized noun phrases, I will use both environments in the remaining analysis.

	Count in partition					
Frequency	1	2	3	4	5	6
All bigrams	178,478	168,937	170,560	160,074	176,808	176,365
Noun-phrase heads	31,162	29,504	29,766	28,291	30,963	15,717
Noun-phrase modifiers	30,841	28,894	28,911	28,575	28,291	15,833

Table 4.9 Raw frequencies of unique noun-phrase heads and modifiers in six partitions of the engineering corpus

	Log-likelihood in partition					
Log-likelihood	1	2	3	4	5	6
All bigrams	10.13	10.65	10.89	10.87	10.77	10.60
Noun-phrase heads	22.07	21.90	22.85	23.76	22.88	31.53
Noun-phrase modifiers	25.19	24.84	26.21	25.76	17.66	35.76

Table 4.10 Log-likelihoods of noun-phrase heads and modifiers

All of the processes described in this section produce lists of noun phrases that have been tagged with a linguistic attribute. The remaining task at this stage of the analysis is the creation of an ARFF file like the one shown in Figure 4.1. The process flow is depicted in Figure 4.6. The ARFF file generator compares the noun phrases to be classified against lists of noun phrases obtained from the linguistic contexts. When a match is found, a count of co-occurrence between a noun phrase and a context is incremented. If the incoming noun phrases are already classified as lexicalized or syntactic, the result is a training ARFF file; otherwise, the result is a test file.

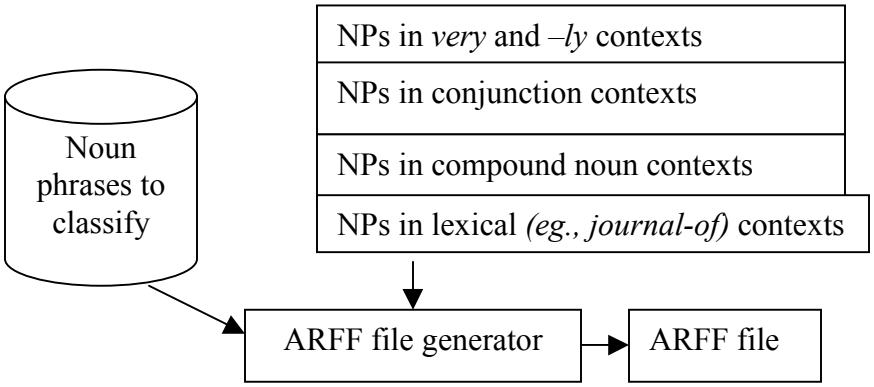

Figure 4.6 Process flow for constructing ARFF files

4.2.3. The linguistic attributes: a summary

To summarize the discussion so far, I have identified a set of positive and negative attributes that are linguistically motivated and bear on the distinction between lexicalized and syntactic noun phrases in a corpus of engineering text. Descriptive statistics suggest that all of the attributes except for *–ly* and *very* specify environments containing noun phrases with high log-likelihood scores, but the hypothesis that these noun phrase are the lexicalized names of concepts in engineering must be tested more rigorously. To accomplish this goal, I use the attributes as input to an algorithm, described in the next section, which classifies a set of noun phrases whose status is already known. In the final section of this chapter, I use the algorithm to classify noun phrases of unknown status and submit the results to verification by human judges.

Before I can confidently perform the next step, I need to determine whether the known syntactic and lexicalized phrases actually appear in the engineering corpus in the linguistic environments I have described. To build on an example I cited earlier in this chapter, if *radiation physics,* whose lexical status is unknown, is presumed to be lexicalized because it appears in the lexical environment *department-of*, it must be verified that analogous noun phrases whose lexical status is known, such as *electrical engineering*, also appear in this environment. If they do not, the learning algorithm has

no exemplars to learn from. The noun phrases which are known to be classified as lexical are the entries in the engineering index that match in the corpus. In all tests performed in the next section, I use the matches in five of the six partitions as input; the matches in the sixth partition are set aside for validation. There are a total of 3,576 lexical matches in the first five partitions. The hypothetical syntactic noun phrases are the bigrams formed with context-dependent adjectives, such as *certain browsers*, *further details* and *full papers,* which are described in Section 4.2.1 of this chapter. I also use the syntactic bigrams from five of the six partitions as input. But since syntactic phrases are far more common than lexicalized phrases, I approximate the size of the set of lexical matches by using a random sample of 11% of the total set of syntactic matches, for a total of 3,526 observations.

Table 4.11 shows a count of the unique co-occurrences of the noun phrases in the training corpus with the linguistic attributes. For example, 2,011/3,576, or 56% of the known syntactic noun phrases co-occur at least once with a conjunction, an attribute that identifies lexical contexts, while only 359/3,526, or 10%, of the syntactic noun phrases do. Conversely, *very* and *–ly* adverbs, which are associated with syntactic contexts, co-occur with 85/3,526, or 5%, of the syntactic phrases but with only 46/3,576, or .012%, of the lexicalized phrases.

Attribute	Lexicalized phrases	Syntactic phrases
Conjunction with a lexicalized phrase	2,011	359
Compound-noun modifier	1,409	258
Compound-noun head	1,384	192
Lexical contexts (*journal-of*, etc.)	944	128
Syntactic contexts (*very*, *ly*-adverbs)	46	190

Table 4.11 Counts of training contexts in all six partitions of the engineering corpus

Table 4.11 shows that the attributes co-occur in the predicted patterns with the already-classified noun phrases and permits inferences regarding their relative importance. Perhaps the most significant observation involves the positive cues for lexicalized phrases: syntactic attributes such as conjunction far outweigh the lexical attributes. However, the predictive power of the attributes cannot be calculated from the analytical methods presented in this section, so the analysis must proceed.

4.2. The training phase

In the training phase of a machine-learning study, the data obtained from the analysis is submitted to a classification algorithm, with the ultimate goal of making accurate category assignments on an unseen portion of the corpus. Since the outcome of the previous section is an ARFF file populated with the results of a linguistic analysis of lexicalized and syntactic noun phrases in the engineering corpus, I now have the information required to execute this step. Witten and Frank describe several classification algorithms and make them available in their software package, but for ease of exposition, I restrict my attention to one: the so-called Naïve Bayes algorithm. As Witten and Frank argue, the choice of algorithm is less important than the quality of the attributes in most machine-learning studies, and the Naïve Bayes algorithm provides an intuitively natural formalization of how evidence observed in a corpus of coherent text influences our belief that a given noun phrase should be classified as lexicalized or syntactic.

To illustrate, I have assembled the relevant calculations in Table 4.12, which shows hypothetical data obtained from six lines of the ARFF file depicted in Figure 4.1. For simplicity, I have converted the numeric variables to categorical variables. If the attribute and the noun phrase co-occur at least once in the corpus, *yes* is recorded in the top third of the table; otherwise *no* is. With this data, we can calculate the likelihood that a noun phrase is classified as lexicalized or syntactic by tabulating the co-occurrences of attributes with a given classification. For example, *department-of* occurs with 1/3 of the lexicalized phrases and *very* co-occurs with 2/3 of the syntactic phrases. In this tabulation, shown in the middle section of Table 4.12, numerators of 3 or 0 are

represented in the fraction as 2.99 and 0.01, respectively. Since likelihood is an estimate of a theoretical but unobservable probability, we can't be sure from our small sample that the co-occurrence of a lexicalized-phrase classification with an attribute is ever 100% or 0%, so these cases are represented as hypothetical numbers that are very high or very low.[10] The likelihood that a given phrase is classified as lexicalized or syntactic is expressed as the product of all of the attribute/phrase-class proportions, shown in the third section of Table 4.12. The two likelihoods are converted to probabilities by normalizing them with a denominator that coerces them to sum to one, as shown at the bottom of Table 4.12.

Put in this form, the last two equations in Table 4.12 are straightforward instantiations of Bayes' rule of conditional probability: $pr[H|E] = pr[E|H]pr[H]/pr[E]$. In other words, Bayes' rule expresses the probability of hypothesis H, given the evidence E; or, the probability that a noun phrase will be classified as lexicalized, given its co-occurrence in coherent text with the linguistic attributes I have identified in this chapter. This can be calculated as a product of the probabilities that the phrase is observed in contexts that favor lexicalized noun phrases, normalized by the sum of the probabilities of the observations. For example, if the data in Table 4.12 for *artificial intelligence* is substituted into the equation for Bayes' rule, the probability that it would be classified as lexicalized is approximately 92%. Since we already know the correct classification of *artificial intelligence* because it is cited in the engineering index, this result might seem uninformative. But the algorithm that encodes Bayes' rule is powerful because it uses this information to classify noun phrases whose status is unknown. For example, if *radiation physics* co-occurs with many of the attributes that are observed with *artificial intelligence*, it would also be classified as lexicalized with a high probability.

However, I must point out that the probabilities in Table 4.12 are unrealistically high because the co-occurrences of the attributes with the classifications are not as robust as this data suggests. Table 4.11 shows the real pattern of co-occurrences, on which the results reported in the rest of this section are based.

[10] In the real data, this adjustment is made to every observation. This is the so-called Laplace estimator, which is described in Manning and Schutze (2000:202).

	Attribute					
Phrase	*department of*	*def'n of*	*topics in*	*very*	Compound modifier	Lexicalized?
certain browsers	no	no	no	no	no	no
abundant compounds	no	no	no	yes	no	no
civil engineering	yes	no	no	yes	yes	yes
remote sensing	no	yes	yes	no	yes	yes
high production	no	no	no	yes	yes	no
artificial intelligence	no	yes	yes	no	yes	yes

Attribute	Co-occurrence with Lexicalized	Syntactic attributes
department-of	1	0.01
definition-of	2	0.01
topics-in	2	0.01
very	1	2
compound-noun modifier	2.99	1

Likelihood of:
lexicalized classification: 1/3 x 2/3 x 2/3 x 1/3 x 2.99/3 = 0.047
syntactic classification: 0.01/3 x 0.01/3 x 0.01/3 x 2/3 x 1/3 = 0.000000082

Probability of:
lexicalized classification: .047/(0.047 + 0.000000082) = approx. 99.9999%
syntactic classification: 0.000000082/(0.047 + 0.000000082) = approx. 0.0001

Table 4.12 Hypothetical co-occurrences of attributes with training data

So far, I have described how the encoding of Bayes' rule enables the Naïve Bayes learning algorithm to learn. But why is it naïve? A detailed answer to this question would take us far afield into statistical theory. But the short answer is that Bayesian inference works best when the attributes identified in the analysis are independent of each other and, at the outset, we naively assume that they are. Since performance is degraded if the attributes are not independent, part of the fine-tuning process, which I will discuss later in this section, is to eliminate those that cause trouble.

I mention this issue not only because it explains the name of the learning algorithm that I use in this study, but also because it suggests how the enterprise of data mining is different from the execution of a typical laboratory experiment. For example, a

team of experimental psycholinguists might design an experiment using small sets of carefully balanced variables. Once they collect the data, they test the hypothesis that motivated the study using statistics that are appropriate for the design of the experiment. The outcome is an inference about a population from which the data was presumed to be a sample, which either supports or fails to find evidence for the hypothesis. By contrast, a data-mining study is motivated by the desire to discover regularities in an existing body of data, which may be unruly or dirty from the perspective of an experimental psycholinguist. As in a controlled experiment, the goal of a data-mining study is to find order. Howver, the end result is not an inference about a population, but useful new information.

Accordingly, a data-mining study requires different methods of evaluation. My evaluation follows established procedure and consists of three steps. First, I perform a sanity check on the design of the study by assuming that the training data can also be interpreted as hypothetical test data. This is the so-called cross-validation test, which is one output of the Witten-Frank software. In this simplest form of evaluation, I can test the effectiveness of the attributes at predicting classifications in the best-case scenario and optimize the fit of the attribute set to the mathematical model encoded in the learning algorithm. In a slightly more abstract test, the algorithm classifies additional noun phrases of known status, which were extracted from the first six partitions of the corpus but were not used as training data. When I assembled Table 4.11, which shows the co-occurrences of noun phrases and linguistic attributes in the training set, I counted only the classified noun phrases in the first five partitions. The classified noun phrases in the sixth partition are reserved for this test. In the third evaluation, a new partition of the engineering corpus is processed. This partition presumably also contains known citations of lexicalized and hypothetical syntactic noun phrases, which can be used to check the performance of the classification algorithm. But a more important test is the evaluation by expert human judges of the classifications assigned to noun phrases of unknown status. This is the useful outcome of my data-mining study, which simulates the research

goal that I stated at the beginning of Chapter 3: to provide a list of words that a human editor would consider appropriate for inclusion in a new edition of a dictionary of engineering terms.

Before reporting the results, I need to describe an adjustment to the input data that is already implicit in the transformation of the ARFF file shown in Figure 4.1 to the fragment depicted in Table 4.12. Figure 4.1 shows numeric values for the co-occurrences of linguistic attributes and noun phrases, which correspond to the observations in the corpus; in Table 4.12, this information is represented as nominal data—in this case, tabulations of *yes* and *no*. In the tests reported below, the data has been 'discretized' using a function available in the Witten-Frank software package. In this transformation, the numeric scale implicit in the raw data for each numeric attribute has been aggregated into a small number of discrete bins, ranging from three to seven, each of which represents a range of numeric values in the scale. Thus the definitive representation of the data is more fine-grained than what is implied by Table 4.12, where the observations have been reduced to two bins for each attribute. Discretization has the effect of minimizing the influence of very large numbers and permits easier comparison of data from different sources in the corpus.

Table 4.13 shows the cross-validation results for all of the linguistic variables that I have studied in this dissertation. As implied in Table 4.11, they are organized into five groups of attributes: conjunctions in lexical environnments, compound noun heads, compound noun modifiers, noun-preposition collocations such as *journal-of*, and adverbial modifiers that favor syntactic phrases, such as *very* and *ly*-adverbs. Overall, the accuracy rate is 76% and the algorithm is better at classifying syntactic than lexicalized phrases.

The data I have presented in this chapter and Chapter 3 can be used to explain why. The linguistic attributes function in the classification algorithm as near-categorical variables that have a few exceptions, but not enough to be disastrous for an appropriately formalized analysis. The relatively low accuracy score for the lexicalized phrases implies that additional linguistic analysis must be done because some lexicalized phrases do not appear in the environments that I have identified and so cannot be distinguished from

syntactic phrases by the currently encoded evidence. But the high accuracy score for the syntactic phrases implies that the test can perform the classification when the variables are observed and that it is conservative, erring more in recall than precision. Thus, it promises to accomplish the goal I described at the beginning of Chapter 3: to present an editor with a list of candidates for inclusion in an engineering dictionary that is low in noise, at the possible expense of missing some bona-fide candidates. Considering that lexicalized phrases are sparse in a corpus of coherent text and that this relatively simple pilot test has only five sets of linguistic attributes, the results support the conclusion that lexicalized and syntactic phrases are not randomly distributed, but appear in privileged environments that can be identified by careful linguistic analysis.

	Classifications		Total correct
Training terms	Syntactic	Lexicalized	classifications (%)
Syntactic	3,243	282	92
Lexicalized	1,414	2,162	60
Overall			76.3

Table 4.13 Cross-validation results for all linguistic variables

To provide support for some of the arguments I have made in this dissertation, I also need to assess the relationship of the linguistic variables to the log-likelihood measure of statistical collocation. My arguments embody two claims. First, log-likelihood scores correlate positively with the distinction between lexicalized and syntactic phrases: lexical contexts generally harbor noun phrases with high log-likelihood scores, while the opposite is true of syntactic contexts. Second, the attributes supplied to the machine-learning algorithm that identify linguistic contexts can compensate for the misleading output of this measure that results when lexicalized phrases score too low because they are made up of words that are common in the subject domain, or when syntactic phrases score too high because they may be idiomatic or stereotypical expressions.

The simplest test of these claims is a second baseline test on the same input files that has a discretized form of the log-likelikhood score as the only attribute. But the cross-validation results in this test are strongly influenced by the selection of the phrases that are not lexicalized. The accuracy rate of the classification may be close to the results reported in Table 4.13 because the syntactic phrases, which are also a product of my linguistic analysis, have lower-than-average log-likelihood values. This has the effect of making the positive correlation to lexical status even stronger. A more realistic baseline would embody none of the linguistic analysis that I have discussed in this dissertation. It might compare noun phrases that are presumed to be lexicalized, perhaps with the feedback of human judges, against a background of noun phrases that have been selected randomly. In the experiments I performed that simulated these conditions, the overall cross-validation scores ranged from 62%-79%, depending on how the negative instances were sampled.

It is conceptually simpler to add the log-likelihood score to the list of linguistic attributes in the ARFF file that generated Table 4.13. The results of this experiment are shown in Table 4.14. The addition of the log-likelihood score improves the performance of the classification algorithm by 10% for lexicalized phrases and approximately 3% for syntactic phrases. This result is perhaps due to the fact that, with the addition of the log-likelihood score, every noun phrase to be classified now has at least one value on an attribute that is relevant to the classification. When the log-likelihood score for a lexicalized phrase that appears in the linguistic environments I have studied is low—as it is for *methanol fuels, magnetic shielding, machine parts*, *metamorphic rocks* and *catalytic cracking*—the linguistic attributes can correct the classification. Analogously, the classification algorithm correctly fails to classify the syntactic noun phrase *complete guide* as lexicalized, despite its relatively high log-likelihood score, because it does not appear in the linguistic contexts that favor lexicalized phrases. But the log-likelihood score can sometimes compensate when the lexicalized-phrase candidate does not appear in the linguistic environments because of the usually high correlation between high measures of statistical association and lexical status.

Ironically, the arguably modest 10% improvement in the classification of lexicalized phrases when log-likelihood is an attribute can support the arguments made by Church and Hanks (1990) and Smadja (1993) that sequences of words with high scores on statistical measures of association are true lexical collocations. I have shown that I can identify essentially the same set of word sequences in a computational study derived solely from linguistic arguments. If I am correct, the researcher now has an alternative way of identifying this kind of lexical information, without having to wade into the technical difficulties of calculating association statistics on linguistic data, which I discussed in Section 2.2.3 of Chapter 2.

	Classifications		Total correct
Training terms	Syntactic	Lexicalized	classifications (%)
Syntactic	3,349	177	94.98
Lexicalized	1,062	2,514	70.03
Overall			86.12

Table 4.14 Cross-validation results for linguistic variables and log-likelihood

Before considering the other methods of evaluation, I need a better understanding of the individual attributes, which can be obtained from a closer examination of the cross-validation results. By systematically subtracting each attribute from the ARFF file and re-running the classification experiment, I can observe whether the performance of the classification algorithm is degraded by the missing information and hence determine whether the attribute contributes to an accurate classification. The results of these tests are summarized in Table 4.15. The data in this table support the conclusion that none of the attributes should be eliminated from the study because all are needed to obtain the best results on lexicalized phrases. In other words, none of the classification scores for lexicalized phrases in Table 4.15 exceed the one reported in Table 4.13. The lexical contexts are more valuable than the syntactic contexts because accuracy is degraded most severely when these contexts are missing. The conjunction and compound-noun contexts introduce noise that slightly degrade the performance of the classifier on syntactic phrases, but not severely enough to argue for their exclusion. The attribute derived from

very and *–ly* adverbs mitigates this problem slightly, but has no impact on the classification of lexicalized phrases, which implies that these contexts function as a robust linguistic test that identifies syntactic phrases, just as Judith Levy (1978) argued.

In future machine-learning studies, references to external knowledge sources such as the Engineering Information Thesaurus might not be necessary. If attributes are restricted contexts such as *topics-in* and *journal-of,* the external lexical resource is not needed because the classification algorithm does not literally refer to them. These contexts were identified through the linguistic analysis that precedes the training step and it is likely that such contexts also identify lexicalized noun phrases in document collections on topics other than engineering. The compound-noun and conjunction contexts also do reasonably well at performing the classification, though the scores on these attributes can be obtained only with an algorithm that makes literal reference to a list of dictionary entries from an external source, as I described in Sections 4.2.2.1 and 4.2.2.2 of this chapter. Without such references, the performance of the compound-noun and conjunctions attributes in the classification task would be slightly noisier than the results I have reported here.

Training terms	Syntactic	Lexicalized	Classifications (%)
No conjunction contexts			
Syntactic	3,290	236	93.33
Lexicalized	1,121	2,455	68.65
Overall			82.09
No compound-noun contexts			
Syntactic	3,420	156	97.02
Lexicalized	1,091	2,485	69.49
Overall			81.94
No lexical contexts			
Syntactic	3,314	212	93.98
Lexicalized	1,259	2,317	64.79
Overall			80.64
No *very*, *-ly* adverb contexts			
Syntactic	3,279	247	93.02
Lexicalize	1,072	2,504	70.02
Overall			82.60

Table 4.15 The relative contributions of attributes to the cross-validation results

With this information, I can now perform the second evaluation on the training data. Recall that the syntactic and lexicalized phrases in the last partition of the training corpus, representing one-sixth of the total, were held back from the training set. This data gives us the opportunity to perform an evaluation on unseen data without the need for human judges. The results are shown in Table 4.16. They are slightly higher than the results on the training data, the first hint that the trained classification algorithm can be successfully applied to novel phrases.

	Classifications		Total correct
Training terms	Syntactic	Lexicalized	classifications (%)
Syntactic	623	28	95.81
Lexicalized	160	532	76.87
Overall			87.30

Table 4.16 Performance on unseen data of known status in the training corpus

More significantly, the classification algorithm can be applied to noun phrases in the training corpus whose status is not known. Some of this output is listed in Table 4.17. This hand-selected list is a preview of the data that will be presented to human judges in the final test, but it makes the discussion in this section concrete and illustrates some problems. For the most part, the noun phrases classified as lexicalized are suggestive of engineering topics, except for place names such as *south america*. The algorithm classified as lexicalized some noun phrases that are listed in the engineering index but were not included in the training data. The phrases preceded by asterisks do not appear in the engineering index and would be identified by this exercise as candidates for inclusion in the next edition. Among the noun phrases classified as syntactic are genuine syntactic phrases such as *effective strategy* and *three professors*. The output tagged with this classification also includes parsing errors such as *orthopedics orthopedics* and phrases that should probably be classified as lexicalized, such as *garbage collection algorithm,* but they are not cited frequently enough in the contexts identified by the attributes that trained the classifier.

Lexicalized phrases	Syntactic phrases
*design tools	pleasant cornish weather
*server software	effective strategy
*test instrumentation	regular columns
linear algebra	huge tome
greenhouse effect	reasonable approaches
*gas turbine engine	garbage collection algorithm
*circuit simulation	dynamical effects
noise control	three professors
nervous system	national advocacy
electromagnetic compatibility	individual entrants
steam traps	payment options
carbon monoxide	ideal vehicle
*internet services	entire human body
fiber optics	senator nickles
information technology	solvent acts
*fault location	web info
*south america	orthopedics orthopedics

Table 4.17 Some new classifications in the training corpus

4.1. The test phase

In the final evaluation, I process a new 30-megabyte partition of the engineering corpus using all of the software described in this chapter and Chapter 2. The outcome is a complete list of simplex noun phrases in the partition, as well as the noun phrases in the partition that occur in the syntactic environments of the attributes described in this chapter: conjunction contexts, lexical contexts, noun-phrase head and modifier contexts and *–ly* and *very* adverb contexts. These lists can be submitted to the ARFF-file generator, depicted in Figure 4.6, for the creation of the test run. For this test, I restrict my attention to the 157,000 noun-phrase bigrams because log-likelihood is one of the effective attributes identified in the training run, and it is calculated differently for bigrams and longer phrases, as I discussed in Section 2.1.3 of Chapter 2. Moreover, since the selections from the list are presented to human judges, a consistent length removes a potential source of response bias.

On a test run, the Witten-Frank software returns two results: a classification and a confidence measure. Before I discuss the results with human judges, I want to take a closer look at this information because it sheds light on the behavior of the attributes. Table 4.18 shows a random sample of classified bigrams that were presented to the

human judges and three levels of confidence returned by the training algorithm for each classification: the highest, the lowest, and the mean. The bigrams in the top third of the table are maximally distinct from each other; the entries in top left cell are, for the most part, the established names of concepts in engineering, while those in the top right cell are syntactic phrases. The bigrams in the upper left quadrant produce high log-likelihood scores, as well as high scores on most of the linguistic attributes I have described in this chapter, while the bigrams in the upper right quadrant fail on all of these measures, except possibly for high scores on the *very* and *-ly* attribute. In the middle of the list, we observe the effects of the linguistic attributes because very few of the bigrams that produce average levels of confidence have high log-likelihoods. Bigrams are classified as lexicalized with some measure of confidence if they appear in two or more linguistic contexts that favor lexicalized noun phrases. The bigrams at the bottom of the table are less distinct, but the classification algorithm is still forced to make a decision, which hinges on the reliability of conjunction versus the lexical contexts like those listed in Table 4.3. If the bigram appears in these lexical contexts, but has no other positive scores for attributes that identify lexicalized noun phrases, it is classified as lexicalized. But if it appears only in conjunction contexts, it is classified as syntactic. This data suggests an interpretation of the conjunction feature that may be used to guide future work: it has the potential to introduce noise by falsely classifying syntactic phrases as lexicalized, but it can't do so in the presence of the other attributes.

	Classification	
Confidence	Yes	No
High	automatic control	high percentage
	safety engineering	sensitive instruments
	decision analysis	low content
	virtual reality	broad introduction
	computer sciences	latest mst
	risk management	photomultiplier tube
Medium	soil contamination	click pay
	environmental geosciences	production industry
	adaptive structures	harmful ones
	photonic technology	geology sofia
	rapid fabrication	ship production
	semiconductor equipment	university users
Low	public economy	test rig
	epitaxial structure	low number
	human virology	aquatic ecology
	land subsidence	drainage improvement
	structural theory	specifications petroleum
	pellet size	powder atomization

Table 4.18 Classifications in the test data

For the test with human judges, I created a random sample of twenty bigrams from each of the six classifications shown in Table 4.18 and presented it to judges. Seven judges, all of whom have at least a bachelor's degree in engineering or who work as an engineer, were presented with a printed list of 120 bigrams that was introduced with the following instructions: 'Below is a list of 120 two-word phrases. If the phrase is the name of a concept or object in engineering, or a field of study that would be affected by the work of engineers, put a check mark beside it. In other words, identify the phrases that you would expect to find in an index about engineers and engineering.'

The results of this simple paper-and-pencil test are shown in Table 4.19. In general, the performance by human judges is predicted by the confidence measures supplied by the trained classification algorithm. For the lexicalized phrases, the algorithm's confidence score correlates positively with human judgments. Noun phrases that achieve the highest confidence scores, such as *automatic control*, *risk management*, and *decision analysis* are the same phrases that professional engineers have heard of. The performance of the judges matches the classification algorithm less reliably on the noun phrases to which the algorithm assigns low confidence scores, such as *epitaxial*

structure and *land subsidence*, perhaps because these phrases have lower frequency or refer to leading-edge concepts that may require specialized expertise to recognize. The overall agreement with the classification algorithm is higher for the 'yes' votes than the 'no' votes, which supports the claim that I made in the discussion of the training data that the algorithm is conservative and hence skewed toward maximizing precision over recall. For example, the noun phrase *photomultiplier tube* received 'yes' votes from all seven judges despite its classification as 'high-no' by the algorithm. But noun phrases such as *drainage improvement* and *powder atomization* are assigned a low-no classification by the algorithm because they appear frequently as noun-phrase heads or modifiers, or in conjunctions with known lexicalized noun phrases, and many judges identify them as names of engineering concepts.

To support an emerging standard in corpus linguistics studies, I must also submit the data in Table 4.19 to the Kappa statistic. As Carletta (1996) argues, Kappa, or K, is a measure of agreement in classification tasks and encodes a correction for the possibility that the judges' performance may be random. Formally, K = p(A) – P(E)/1-p(E), where p(A) is a count of observed, or actual, agreements and p(E) is the number of agreements that would be expected by chance. K = 1 if there is complete agreement and 0 if there is only chance agreement. Since only p(A) is represented in Table 4.19 and p(E) is an error term, the percentages in the rightmost column represent an over-estimate of K. For all of the observations in Table 4.17, K = .61,[11] which rises to .63 if the borderline low-no and low-yes categories are eliminated. Standards of acceptance for values of K vary according to the subfield of computational linguistics, but .61 and .63 are slightly low for a mature area of study, as Carleta argues, where Kappa scores of .7 and above are reported. Nevertheless, they represent a detectable effect, which is a reasonable outcome for a pilot study. Agreements between the classification algorithm and the judges can probably be substantially increased by constraining the subject of the corpus. Many of the experts who participated in my study expressed a lack of confidence that they had mastered all of the topics represented in the test.

[11] In a confusion matrix, such as the top half of Table 4.17, the diagonals represent the agreements; the lower-left and upper right quadrants are the disagreements. K is calculated on this data as follows: 2(352*324 – 68*96)/((352+96)*420 + (68+324)*420) = .61

	Judges		
Algorithm	Yes	No	Agreement
Yes	352	68	352/420 (83%)
No	96	324	324/420 (77%)
Agreements of judges and confidence levels reported by the algorithm levels			
High yes	130	10	130/140 (92%)
Medium yes	118	22	118/140 (84%)
Low yes	104	36	104/140 (74%)
High no	34	106	106/140 (76%)
Medium no	14	126	126/140 (90%)
Low no	48	92	92/140 (66%)

Table 4.19 Agreements between the classification algorithm and human experts

Following Carleta's recommendations, I also calculated Kappa scores for all pairs of judges. The results, shown in Table 4.20, show a high level of inter-judge agreement, since only 3 of the 21 pairs have Kappa scores below .60, while 15 pairs have Kappa scores that at or above .68. Indeed, the agreement among the judges is higher than the pooled agreement of the human judgments with the classification algorithm, which suggests that the task made sense and was interpreted in a consistent way by the seven judges. Failure analysis of individual items suggests three reasons for divergence with the classification algorithm. First, the classification algorithm gave an incorrect classification for some genuine engineering concepts, such as *photomultiplier tube*. Second, the classification algorithm identified phrases such as *aquatic ecology* and *environmental geosciences* as lexicalized, but some of the judges believed that they were names of concepts in science, not engineering. Finally, the test set contained some mis-parsed phrases such as *specifications petroleum*, which the algorithm correctly classified as not lexicalized. But this phrase was selected by four judges, presumably because of the word *petroleum*, which spuriously suggets an engineering topic.

Judges	1	2	3	4	5	6	7
1	--	.599	.705	.679	.695	.932	.793
2		--	.639	.740	.705	.668	.599
3			--	.770	.672	.705	.575
4				--	.772	.742	.615
5					--	.771	.707
6						--	.864
7							--

Table 4.20 Kappa scores for all pairs of human judges

4.5. Summary and conclusions

The study described in this chapter accomplishes several goals. First, I have shown that, even with relatively noisy heuristic processing, linguistically natural evidence that is well-chosen can pass through many layers of rigorous empirical tests and go far toward solving the central problem of this dissertation. I have also established a simple testbed that can accept much more input from linguistic analysis. Second, the results reported in this chapter support the claim that the task of distinguishing lexicalized from syntactic noun phrases can be studied productively as a machine-learning problem that is closely related to the problem of word-sense disambiguation. Taken together with the arguments presented in Chapter 3, I have shown that a traditional linguistic analysis, which obtains insights primarily from introspection, can be supported, broadened and deepened by a computational analysis derived from a large corpus of coherent text.

CHAPTER 5

CONCLUDING REMARKS

5.0. Syntactic and lexicalized noun phrases in coherent text

At first encounter, the problem I addressed in this dissertation appears to be elusive. While most noun phrases in speech and writing are the throw-away reflex of syntactic creativity, some masquerade as names of persistent concepts and have acquired word-like qualities. Is it possible to tell the difference?

Traditional linguistic scholarship is only guardedly optimistic about the prospects of making the distinction. Many scholars agree that such names exist and figure prominently in lexicons of technical topics. But theoretical linguists concluded that no reliable formal criteria can be identified for separating them from syntactic phrases because speakers and writers can use any linguistic means at their disposal to create an expression, which may become lexicalized through repeated use. Notable examples of lexicalized noun phrases include compound nouns such as *eggplant* and *garbage man*; noun-adjective phrases such as *blackbird* and *high school*; or phrases with occasionally more complex syntactic forms, such as *very important person*, *frequently asked questions* and *kiss-me-under-the-garden-gate*. Such phrases are considered to be lexicalized because they are the usual and expected names of concepts, just as single words are. To refer to a tomato by any word or phrase other than *tomato* is to risk being labelled a less-than-proficient speaker of American English; the same can be said of *high school* or *garbage man*. Computational linguists enriched the theoretical linguist's account with an important observation: not only is the referent of a lexical phrase persistent, but so is the form, and this persistence can be quantified by statistical measures of association in a

sufficiently large corpus. But the computational linguists' account has significant gaps, and the output of their processes must be reviewed by experts in lexicography, terminology, or a particular subject domain.

My research attempts to fill some of the gaps in the computational analysis by developing the argument that syntactic and lexicalized noun phrases have different distributions in a corpus of coherent text. To achieve analytical rigor, I obtained most of my data from a large corpus of engineering documents, but I believe that the argument can be applied to other subject domains. In fact, this line of inquiry originated with Judith Levy, who did not restrict herself to engineering terminology when she observed that syntactic noun phrases may be modified by *very* and *–ly* adverbs but lexicalized noun phrases usually cannot. In the engineering corpus that I studied, this observation accounts for the fact that judges reject the combination of *very* with lexicalized phrases, as in **very electrical engineering* and **very magnetic resonance*; but the syntactic phrases *very well-equipped laboratories* and *very complex interfaces* are acceptable and even attested in the corpus. By contrast, lexicalized noun phrases appear in syntactic, lexical and discourse contexts that create an expectation for a conventional name. For example, if I say *professor of...*, my behavior would run counter to expectation if I did not complete the phrase with a familiar name of a field of study that could be found in a dictionary or college catalog, such as *economics*, *comparative literature*, or *artificial intelligence*. Section 3.4.1 Chapter 3 cites other contexts that favor words or lexicalized phrases. The details of the argument may be new, but all of the evidence is consistent with the observation made by linguists working in the 1960s and 1970s that syntactic noun phrases describe temporary relationships, while lexicalized noun phrases are names that have survived a single context of use.

What emerges from my analysis is a set of lingistic tests for identifying syntactic and lexicalized noun phrases from a corpus. Traditional scholars of syntax and word-formation processes were looking for such tests, too, but they were discouraged by the fact that all of the tests have exceptions. Yet this result should not be surprising, considering that the distinction between the two classes of phrases is not deeply encoded in grammar, but has roots in the sociology of language use, which may be subject to

variation. But it does require a change of methodology. Instead of evaluating linguistic tests by our ability to generate counterexamples, we obtain observations from a corpus of coherent text. Tests developed by computer scientists who specialize in machine learning can determine whether the evidence more-or-less robustly predicts whether a given noun phrase is syntactic or lexical. This may seem like a radical change in methodology, but it respects the 'fuzzy' quality of the data that nearly all scholars who have worked on the problem have observed and produces a clean framework that is receptive to more input from linguistic analysis. When I started this project I did not expect to discover a nearly seamless transition from Alice Morton Ball's classic 1941 study of English compounds such as *lighthouse* and *shellfish* to my machine-learning study of engineering terminology harvested from the Web sixty years later. There is no question that a period of analysis must precede such a study, and linguistic analysis has proven to yield reliable insights that can be formalized and projected into a body of text that is too large to inspect.

The main conclusion of this dissertation is that lexicalized noun phrases are not randomly distributed in coherent text, but instead appear in restricted syntactic and lexical environments that can be identified through careful linguistic analysis. In practical terms, this means that if the task is to supply an editor of a specialized dictionary with candidates for a new edition, a software program can find them in a text by intelligently skipping around, in the tradition of data-mining studies based on knowledge-poor computational techniques, without having to calculate statistical measures of association, which can sometimes be unreliable. Philosophically, this conclusion implies that speakers and writers have a larger-than-expected lexicon, with many entries that look like the ordinary noun phrases used in descriptions of temporary circumstances, which they frequently consult to construct sentences that are in line with the expectations of readers and listeners.

5.1. A theory of common usage

In Chapter 1, I cited a position paper by Steven Abney, who argued that empirical studies of language are valuable because they provide data-tested input to a theory of common usage. So, what are lexicalized noun phrases like? Previous linguistic studies have speculated that *lighthouse* is a fairly typical example. This noun-noun compound is so persistent that it is no longer written as two words, and as a referent to the tall, cylindrical structure that guides ships to the coastline, it seems only distantly evocative of houses. But but there aren't many lighthouses in engineering text. Perhaps *database* followed a similar evolutionary path, starting out as *data base* and morphing to *data-base* before ending up as a single word, whose transparent meaning 'base of data' does not fully evoke the current usage that implies a structured repository of machine-readable information about a single topic. But there are two reasons why *lighthouse* and *database* are not typical of the lexicalized noun phrases found in engineering text.

First, the textbook cases of compound nouns originate from short and relatively frequent monosyllabic words and can still be easily parsed by human readers when they are written as a single word. But I predict that most of the citations in the Engineering Index Thesaurus, or the noun phrases identified by my software in the Engineering corpus as lexicalized, will never achieve the appearance of uncontroversial words, no matter how old or frequently used they are. The texts I have analyzed have no citations of *electricalengineering*, *airtrafficcontrol*, *magneticresonanceimaging*, or s*tatisticalthermodynamics*. There is probably no extraordinary pressure to represent these words as single tokens, except in unusual contexts where typographical white space is a technical liability—as it is in the names of Web addresses, which sometimes take the form of collocations such as *iknowican*, *howstuffworks*, *georgewbushforpresident*, or lexicalized noun phrases like those that are the object of my study.

Second, it is not obvious that many of the lexicalized noun phrases I have identified in the Engineering corpus have lost much of their semantic transparency, at least to the extreme that *lighthouse* has. In my study, I have departed from tradition by making no reference to semantic opacity as a defining criterion of lexicalized noun phrases. I have seen no need to, partly because metalinguistic cues that identify their

lexical status are frequent enough in a corpus of coherent text to be the object of rigorous study, and partly because the theoretical issues regarding semantic composition remain controversial. Consider, for example, the phrase *artificial intelligence*. Computer programs that mimic the cognitive processes of humans can certainly be characterized as intelligence that is artificial, but the phrase *artificial intelligence* does not convey the fact that this is also the name of an established area of study in computer science. As I argued in Chapter 1, perhaps a small degree of semantic transparency is lost as soon as a phrase is adopted as a name. Though the phrase is now ambiguous between a name and a description, the name persists because the act of naming is psychologically difficult, and a name that depicts salient characteristics of a concept eases the burden of processing for both speaker and listener. As a result, further semantic drift may not be observable for a long time, if ever.

Because most lexicalized noun phrases are probably not destined to assume what many scholars would argue to be the two defining characteristics of words, the research issues surrounding the status of *integrated circuits*, *internet services*, *water supply systems*, and *garbage collection algorithm* will remain unresolved. But if, as I have claimed, such phrases are words, we should be able to notice side effects in the lexicon, and I believe we can.

One consequence is, simply, that the lexicon might be much bigger than we have previously assumed. This can be studied as a psycholinguistic issue, as I hinted in Chapter 3. If hesitations in speech occur before and after phrases such as *rights mangement*, but not in the middle, then we might reasonably think of this as a lexical chunk that is as solid as *eggplant*. And if failures of communication occur when a perfectly transparent phrase such as *legal deposit* is not understood and a speaker refers to it as 'vocabulary,' then this must be a word that names a persistent concept, not a description whose meaning is clear from immediate experience.

In the study of collocations in the mental lexicon, foreign-language teachers are on the forefront. As a student of German in the 1970s, I had to memorize long lists of 'separable-prefix verbs' such as *ab-hacken (to tear apart)*. Foreign-language teachers

now routinely comb the research results of corpus linguists for lists of collocations like the ones that can be extracted from coherent text with the software I have developed because they understand that, to become fluent speakers, students must confront recurrent sequences of words as lexical knowledge to be memorized, and not to be re-composed on the fly as needed.

Is this enlarged lexicon just a big list, or are the entries connected by lexical relations? Of course, the more interesting possiblity is the second one. And the authors of the Engineering Information Thesaurus, at least, believe that the lexicalized noun phrases in engineering enter into hypernomy/hypony relationships with single words, as in *abutments/bridge components*, as well other noun phrases, as in *acoustic microscopes/imaging techniques*. But logically possible relationships showing relative degrees of abstraction may not require that the phrases in the relationship be lexicalized because the same relationship connects *cars* and the ordinary syntactic phrases *green cars*, *expensive cars* or *big cars*. Pairs of words and phrases in such relationships that are undeniably lexicalized are members of word associations such as *hot/cold* and *thick/thin*. Unfortunately, as any student of psycholinguistics knows, word associations are positively correlated with word frequency, and if lexicalized noun phrases are words, they are not the highly frequent ones that would be likely to participate in such relationships.

But consider the snippet from a letter that a cancer patient's family physician received when the patient returned from a visit to an oncologist: 'Since the lesion appears to be completely excised at this time and is unlikely to have an impact on her overall survival, I have not scheduled a return appointment.' Is this good news? The key to understanding this sentence is the phrase *overall survival*, which we can analyze by processing a corpus of medical texts about oncology harvested from the Web in 1999 using the software that I have described in this dissertation. Table 5.1 lists some citations.

1. Early trials suggested disease-free and overall survival benefit for node-negative patients.

2. CMFPT, given for 1 year, failed to improve either disease-free or overall survival compared surgery alone.

3. No differences in disease-free survival and overall survival were observed in these two treatment arms.

4. Throughout 5 years of followup, the chemotherapy plus tamoxifen regimen resulted in a 91% disease-free survival and a 96% overall survival.

5. There was, however, no significant difference in the disease-free, distant disease-free, or overall survival in the patients receiving preoperative chemotherapy as compared to those receiving postoperative chemotherapy.

Table 5.1. Citations of *overall survival* in a corpus of medical text

In all of the citations except for Sentence 5, *overall survival* is conjoined with *disease-free survival*, which reveals volumes to the perceptive linguist. First, we can infer that *overall survival* is a collocation because the same expression appears in the oncologist's letter as well as in a random collection medical research articles discovered on the Web by an automated process. If so, we can infer that *disease-free survival* is probably a collocation, too, because it is conjoined with a lexicalized noun phrase, which also implies that it has a related meaning. Moreover, Sentence 1 shows that *overall survival* and *disease-free survival* both appear as modifiers of a compound noun, forming the expressions *overall survival benefit* and *disease-free survival benefit*. Sentence 4 provides further evidence that *overall survival* and *disease-free survival* are lexicalized and drops a hint at how the meanings of the two phrases differ: overall survival is not as hard to obtain in a study of medical treatment regimens as disease-free survival. On the basis of this evidence, I would claim that *overall survival* and *disease-free survival* are highly associated lexicalized noun phrases in the oncologist's mental lexicon, and the oncologist's choice of the first expression in the letter about the patient permits the inference that the lesion could recur but is not life-threatening. This example also shows that, when laymen have access to the conventional names of concepts in an unfamiliar subject, they have a powerful tool for unlocking its secrets.

5.2. Some extensions

One goal of this research was to establish support for the claim that noun phrases can be classified as syntactic or lexicalized, based on linguistically sophisticated evidence found in a corpus of coherent text. This goal was stated in general, and perhaps vague, terms for two reasons. First, it was not obvious at the outset that lexicalized noun phrases are sufficiently stable and persistent to be easily distinguishable from syntactic phrases, especially in a technical subject domain that is constantly evolving and for which non-specialists have no clear intuitions. Second, the goal was operationalized by assuming that the task is to automate the discovery of new terminology that could be added to a dictionary of engineering terms, whose entries may encompass many semantic classes, including names of sub-disciplines such as *mathematical sciences*, *earth sciences* and *computer science*; or names of substances, such as *zeolite-encapsulated iron* and *mercury cadmium*.

Of course, the methodology that I used in this dissertation could be applied in more focused studies of lexicalized noun phrases that belong to a single semantic class. In Section 3.4.1 of Chapter 3, I mentioned one study that is already underway devoted to the automatic discovery of the hierarchical structure of discipline names. *Engineerin*g is more abstract than *environmental technology*, which is, in turn, more abstract than *agribiodiversity assessment*. And it may not be coincidental that the first appears as the object of *careers in…,* while the second and third are the objects of *workshop on…,* suggesting that only the last two phrases name concepts that are close to the leading edge of knowledge in this field of study.

As another example, the computational techniques used in my study can extract names of machines and manufactured artifacts, which are abundant in a corpus of engineering text. The object of *manufacturer-of* includes noun phrases such as *bjork-shiley heart valve*, *ozone water treatment equipment*, *engineered industrial products*, *biological wastewater treatment equipment*, *screening equipment*, *filter presses*, *sludge dryers*, *the vortisand ultrafine water filtration system*, *memory upgrade modules*, and *auto speaker cabinets*. For ease of computation, I restricted my attention to a small number of syntactic and lexical contexts in this dissertation, but an analysis that does not

extend to the syntactic objects of verbs is arbitrarily restricted. After all, rich contexts for lexicalized noun phrases such as *study of*, *design of*, *application of,* and others listed in Table 4.2 in Chapter 4, are nominalizations of common verbs and many of the same phrases can be observed in the corpus as objects of the corresponding verbs.

5.3. Future prospects

The research reported in this dissertation has so many potential applications that I was motivated to conduct my study by a sense of urgency. It is a truism in the computational linguistics community that vocabulary acquisition is a bottleneck that impedes progress in the development of sophisticated language understanding systems. As a result, nearly 50% of the papers presented at recent international computational linguistics conferences have been devoted to the problem of lexical knowledge acquisition.

But the output of the software I have described is valuable even if it is not made available to a larger natural language processing system. As an organizing theme, I focused my research on the practical application of creating an automated tool that assists a lexicographer in selecting candidates for a future edition of an engineering dictionary. But lexicographers are not the only potential beneficiaries. Lexicalized noun phrases are so important for the effective use of Internet search engines that a recently published book (MacDonald and MacDonald 2001) contains nothing more than lists of words and phrases that the authors judge to be effective search terms related to various subjects. The section on engineering lists highly abstract lexicalized phrases such as *systems engineering* and *automatic control*, but none of the entries come close to the specificity of *digital watermarking*, which my software discovered as the object of the lexical cue *bibliography-on*. When *digital watermarking* is used as a query in an Internet search engine, it yields just five documents, all on the same subject, one of which is an annotated bibliography[12] that defines the phrase and describes its connection to steganography, cryptography and data hiding. Lexicalized noun phrases are effective search terms because they have established, stable referents in a single subject domain.

[12] Accessible at: <http://www.jjtc.com/Steganography/>

But syntactic phrases are useless in this task, just as they would be uninformative clutter in a dictionary. But when the syntactic phrase *comprehensive environment* is used as a search term, it yields 1200 matches on no particular subject.

Nevertheless, I am not optimistic that the extraction of lexicalized noun phrases from stores of coherent text will be a fully automated process in the foreseeable future. I have identified many points in my analysis where important details have been glossed over. Until more effective solutions to these problems are implemented, the output will appear to be noisy to the intended audience. As a result, human involvement will be necessary for guiding the analysis, or for selecting and evaluating the output. But the identification of lexicalized noun phrases remains an enduring subject for basic research.

BIBLIOGRAPHY

Abney, Steven. 1996. Statistical methods in linguistics. In Klavans, Judith and Philip Resnik (eds.), *The Balancing Act*: *Combining Symbolic and Statistical Approaches to Language*, pp. 1-26. Cambridge, MA: MIT Press.

Abney, Steven. 1996. Part-of-speech tagging and partial parsing. In Young, S. and G. Bloothooftt, (eds.). *Corpus-Based Methods in Language and Speech Processing*, pp. 118-136. Dordrecht, The Netherlands: Kluwer Academic Publishers.

Ball, Alice Morton. 1941. *Compounding in the English Langauge.* New York: The H.H Wilson Company.

Bourigault, Didier. 1992. Surface grammatical analysis for the extraction of terminological noun phrses. In *Proceedings of the Fourteenth International Conference on Computational Linguistics (COLING-92).* Nantes, France.

Bowden, Paul, Lindsay Evett and Peter Halstead. 1998. Automatic acronym acquisition in a knowledge extraction program. In Bourigault, D., C. Jaquemin, and M. L'Homme, (eds.), *Computerm '98: First Worshop on Computational Terminology: Proceedings of the Workshop*, pp. 43-49. Montreal.

Brill, Eric. 1995. Transformation-based error-driven learning and natural language processing: A case study in part of speech tagging. *Computational Linguistics*. 21:543-565. Accessible at <http://www.cs.jhu.edu/~brill/papers.html> .

Busa, Federica and Michael Johnston . 1996. Cross-linguistic semantics for complex nominals in the Generative Lexicon. AISB Workshop on Multilinguality in the Lexicon, Sussex, England. Accessible at: http://www.cs.brandeis.edu/~johnston/home.html

Carletta, Jean. 1996. Assessing agreement on classification tasks: the kappa statistic. *Computational Linguistics*. 22: 249-254. Accessible at: <http://www.cogsci.ed.ac.uk/~jeanc/squib.pdf>

Carroll, John M. 1985. *What's in a Name? An Essay on the Psychology of Reference*. New York: W.H. Freeman.

Chan, Lois Mai. 1995. *Library of Congress Subject Headings.* Englewood, Colorado: Libraries Unlimited.

Charniak, Eugene. 1996. *Statistical Language Learning.* Cambridge, MA: MIT Press.

Chomsky, Noam. 1963. Current issues in lingistic theory. In Fodor, Jerry and Jerrold Katz, eds. *The Structure of Language: Readings in the Philosophy of Language*. Englewood Cliffs, New Jersey: Prentice-Hall.

Church, Kenneth. 1988. A stochastic parts program and noun phrase parser for unrestricted text. *Proceedings of the 2nd Conference on Applied Natural Language Processing*, pp. 136-143.

Copestake, Ann and Alex Lascarides. 1997. Integrating symbolic and statistical representations: the lexicon-pragmatics interface. *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, pp. 136-143. Madrid, Spain.

Daille, Beatrice. 1996. Study and implementation for combined techniques for automatic extraction of terminology. In Klavans, J. and P. Resnik (eds.), *The Balancing Act*: *Combining Symbolic and Statistical Approaches to Language*, pp. 49-66. Cambridge, MA: MIT Press.

Dillon, Martin and L. K. McDonald. 1983. Fully automatic book indexing. *Journal of Documentation*. 39:1-17.

Downing, Pamela. 1977. On the creation and use of English compound nouns. *Language*. 53:810-842.

Dowty, David. 1979. *Word Meaning and Montague Grammar: The Semantics of Verbs and Times in Generative Semantics and in Montague's PTQ*. Boston, MA: D. Reidel.

Dunning, Ted. 1993. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*. 19:61-74.

Fellbaum, Christiane, (ed). 1998. *WordNet: An Electronic Lexical Database.* Cambridge, MA: MIT Press.

Gleitman, Lila and Henry Gleitman. 1970. *Phrase and Paraphrase: Some Innovative Uses of Language.* New York: W. W. Norton.

Godby, Carol Jean. 2001. On the subject of subjects. Paper accepted for presentation at the Seventh International International Society of Knowledge Organization (ISKO) Conference: "Challenges in Knowledge Representation and Organization for the 21st Century: Integration of Knowledge across Boundaries."

Godby, Carol Jean and Ray Reighart. 2001. Terminology identification in a collection of Web resources. In K. Calhoun and J. Riemer (eds.), *CORC: New Tools and Possibilities for Cooperative Electronic Resource Description*, pp. 49-66. New York: Hayworth Press.

Godby, Carol Jean and Ray Reighart. 1999. The WordSmith indexing system. *The Annual Review of OCLC Research* Accessible at: http://www.oclc.org/research/publications/arr/1998/godby_reighart/wordsmith.htm

Halliday, M.A.K. and Ruqaya Hasan. 1976. *Cohesion in English.* London: Longman.

Hays, W.L. 1981. *Statistics*, 3rd edition. New York: Holt, Rinehart and Winston.

Hearst, Marti. 1998. Automated discovery of WordNet relations. In Fellbaum, Christiane, (ed.) *WordNet: An Electronic Lexical Database*, pp. 131-151. Cambridge, MA: MIT Press.

Hindle, Donald. 1994. A parser for text corpora. In Zampolli, A., (ed.), *Computational Approaches to the Lexicon*, pp. 103-151. New York: Oxford University Press.

Hindle, Donald and Mats Rooth. 1993. Structural ambiguity and lexical relations. *Computational Linguistics*. 19: 103-120.

Hornby, S. 1974. *Oxford Advanced Learner's Dictionary*. Oxford: Oxford University Press.

Ide, Nancy and Jean Veronis. 1998. Introduction to the special issue on word sense disambiguation: the state of the art. *Computational Linguistics.* 24:1-40.

Johnston, Michael, Branimir Boguraev and James Pustejovsky. 1995. The acquisition and interpretation of complex nominals. *Working Notes of AAAI Spring Symposium on the Representation and Acquisition of Lexical Knowledge*, Stanford University, Palo Alto, California.

Justeson, John S., and Slava M. Katz. 1995. Technical terminology: some linguistic properties and an algorithm for identification in text. *Natural Language Engineering*. 1:9-27.

Kilgariff, Adam. 1997. I don't believe in word senses. *Computers and the Humanities*. 31:91-113.

Kilpatrick, James. 1999. Nouns sound awkward when masquerading as adjectives. *Columbus Dispatch*, May 19, Section E, p.4 .

Klavans, Judith and Philip Resnik (eds.) 1996. The Balancing Act: Combinig Symbolic and Statistical Approaches to Language. Cambridge, MA: MIT Press.

Koch, Traugott. 1998. Browsing and searching Internet resources: subject based, robot-generated search services. Accessible at: < http://www.ub2.lu.se/nav_menu.html#rosubj>.

Kucera, Henry and W. Nelson Francis. 1967. *Computational Analysis of Present-Day American English.* Providence, RI: Brown University Press.

Lapata, Maria. 1999. Acquiring lexical generalizations from corpora: a case study for diathesis alternations. *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pp. 397-404.

Lauer, Mark. 1995. Corpus statistics meet the noun compound: some empirical results. *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*, pp. 47-54.

Levi, Judith. 1978. *The Syntax and Semantics of Complex Nominals*. New York: Academic Press.

Lin, Dekang. 1999. Automatic identification of non-compositional phrases. Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics, pp. 317-324.

Lyons, John. 1977. *Semantics, Volume 2*. Cambridge: Cambridge University Press.

MacDonald, Randall and Susan Priest MacDonald. 2001. *Successful Keyword Searching: Initiating Research on Popular Topics Using Electronic Databases*. Westport, Conn: Greenwood Press.

Manning, Christopher and Hinrich Schütze. 1999. *Foundations of Statistical Natural Language Processing*. Cambridge, MA: MIT Press.

Marchand, Hans. 1969. *Categories and Types of Present-day English Word Formation*. Munich: C.H. Beck'sche Verlagsbuchhandllung.

Mitchell, Joan, Julianne Beall, Winton Matthews and Gregory New, eds. 1996. *Dewey Decimal Classification and Relative Index: Devised by Melvil Dewey*. Albany, NY: Forest Press.

Mitra, M., C. Buckley, A. Singhal, and C. Cardie, 1997. An analysis of statistical and syntactic phrases. *The RIAO 97 Proceedings*, pp. 200-214. Montreal.

Monaghan, Peter. 2001. Literary Lists Are (1) Interesting (2) Important (3) Everywhere. *The Chronicle of Higher Education*. September 28.

Milstead, Jessica, ed. 1995. *Engineering Information Thesaurus*. Revised, 2nd Edition. Hoboken, NJ: Engineering Information, Inc.

Murphy, Gregory. 1988. Comprehending complex concepts. *Cognitive Science*. 12:529-562.

NISO. 1993. *ANSI/NISO Z39.19-1993 Standard. Guidelines for the Construction, Format, and Management of Monolingual Thesauri, American National Standard*. NISO Press: Bethesda, Maryland, 1993.

Pustejovsky, James. 1995. *The Generative Lexicon*. Cambridge, MA: MIT Press.

Schone, Patrick and Daniel Jurafsky. 2001. Is knowledge-free induction of multiword unit dictionary headwords a solved problem? *Proceeding of the Conference on Empirical Methods in Natural Language Processing*, pp. 100-108. Pittsburgh, PA.

Smadja, Frank. 1993. Retrieving collocations from text: Xtract. *Computational Linguistics.* 19:143-177.

Turney, P. 2000. Learning algorithms for keyphrase extraction. *Information Retrieval.* 2(4): 303-336. Accessible at: http://extractor.iit.nrc.ca/publications/IR2000.pdf

Turney. P. 1997. Extraction of keyphrases from text: evaluation of four algorithms. National Research Council of Canada, report ERB-051.

Wacholder, Nina. 1998. Simplex NPs clustered by head: a method for identifying significant topics within a document. *The Computational Treatment of Nominals: Proceedings of the Workshop*, pp. 70-79. Montreal.

Witten, Ian and Eibe Frank. 2001. *Data Mining*: *Practical machine learning tools and techniques with Java implementations.* New York: Morgan Kauffman Publications.

Yarowsky, David. 1994. Decision lists for lexical ambiguity resolution: application to accent restoration in Spanish and French. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*, pp. 88-95.

Xhai, Chengziang. 1997. Exploiting context to identify lexical atoms—a statistical view of linguistic context. Manuscript. Available from cmp-lg/9701001.

Zhou, Joe and Dapkus, Peter. 1995. Automatic suggestion of significant terms for a predefined topic. In *Proceedings of the Third Workshop on Very Large Corpora*, pp.131-147.

Zipf, H.P. 1949. *Human Behavior and the Principle of Least Effort*. Cambridge, Massachusetts: Addison-Wesley.