An Overview of Empirical Natural Language Processing

Eric Brill and Raymond J. Mooney

■ In recent years, there has been a resurgence in research on empirical methods in natural language processing. These methods employ learning techniques to automatically extract linguistic knowledge from natural language corpora rather than require the system developer to manually encode the requisite knowledge. The current special issue reviews recent research in empirical methods in speech recognition, syntactic parsing, semantic processing, information extraction, and machine translation. This article presents an introduction to the series of specialized articles on these topics and attempts to describe and explain the growing interest in using learning methods to aid the development of natural language processing systems.

One of the biggest challenges in natural language processing is how to provide a computer with the linguistic sophistication necessary for it to successfully perform language-based tasks. This special issue presents a machine-learning solution to the linguistic knowledge-acquisition problem: Rather than have a person explicitly provide the computer with information about a language, the computer teaches itself from online text resources.

A Brief History of Natural Language Research

Since its inception, one of the primary goals of AI has been the development of computational methods for natural language understanding. Early research in machine translation illustrated the difficulties of this task with sample problems such as translating the word *pen* appropriately in "The box is in the pen" versus "The pen is in the box" (Bar-Hillel 1964). It was quickly discovered that understanding language required not only lexical and grammatical information but semantic, pragmatic, and general world knowledge. Nevertheless, during the 1970s, AI systems were developed that demonstrated interesting aspects of language understanding in restricted domains such as the blocks world (Winograd 1972) or answers to questions about a database of information on moon rocks (Woods 1977) or airplane maintenance (Waltz 1978). During the 1980s, there was continuing progress on developing natural language systems using hand-coded symbolic grammars and knowledge bases (Allen 1987). However, developing these systems remained difficult, requiring a great deal of domain-specific knowledge engineering. In addition, the systems were brittle and could not function adequately outside the restricted tasks for which they were designed. Partially in reaction to these problems, in recent years, there has been a paradigm shift in natural language research. The focus has shifted from *rationalist* methods based on hand-coded rules derived to a large extent through introspection to *empirical*, or *corpus-based*, methods in which development is much more data driven and is at least partially automated by using statistical or machine-learning methods to train systems on large amounts of real language data. These two approaches are characterized in figures 1 and 2.

Empirical and statistical analyses of natural language were previously popular in the 1950s when behaviorism was thriving in psychology (Skinner 1957), and information theory was newly introduced in electrical engineering (Shannon 1951). Within linguistics, researchers studied methods for automatically learning lexical and syntactic information from corpora, the goal being to derive an algorithmic and unbiased methodology for deducing the structure of a language. The main insight was to use distributional information, such as the environ-

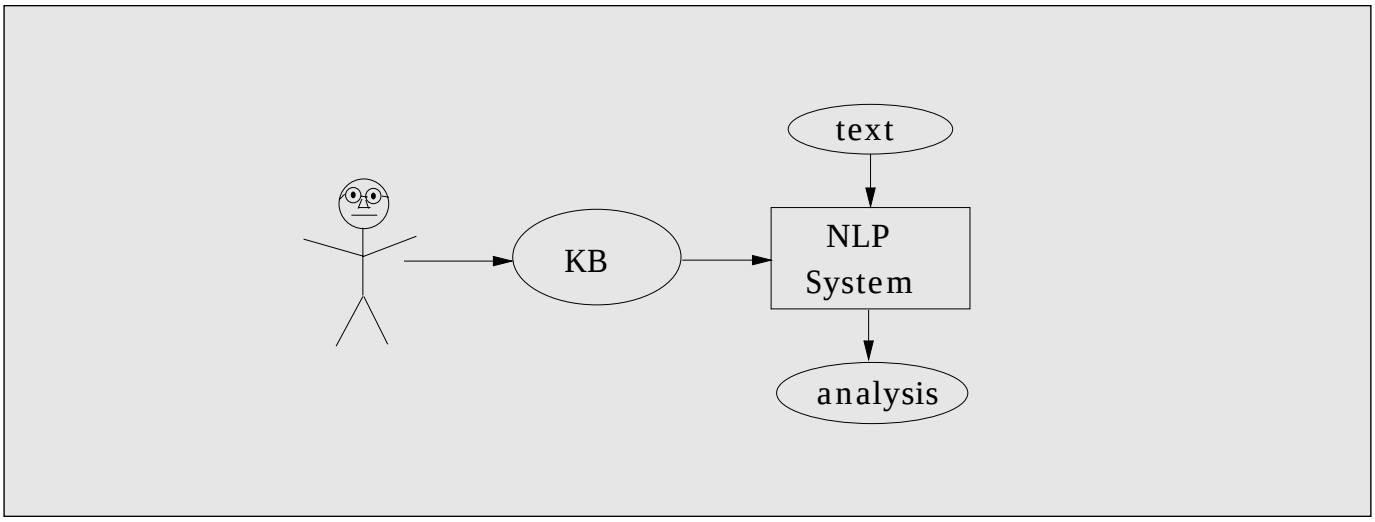

Figure 1. Traditional (Rationalist) Natural Language Processing.

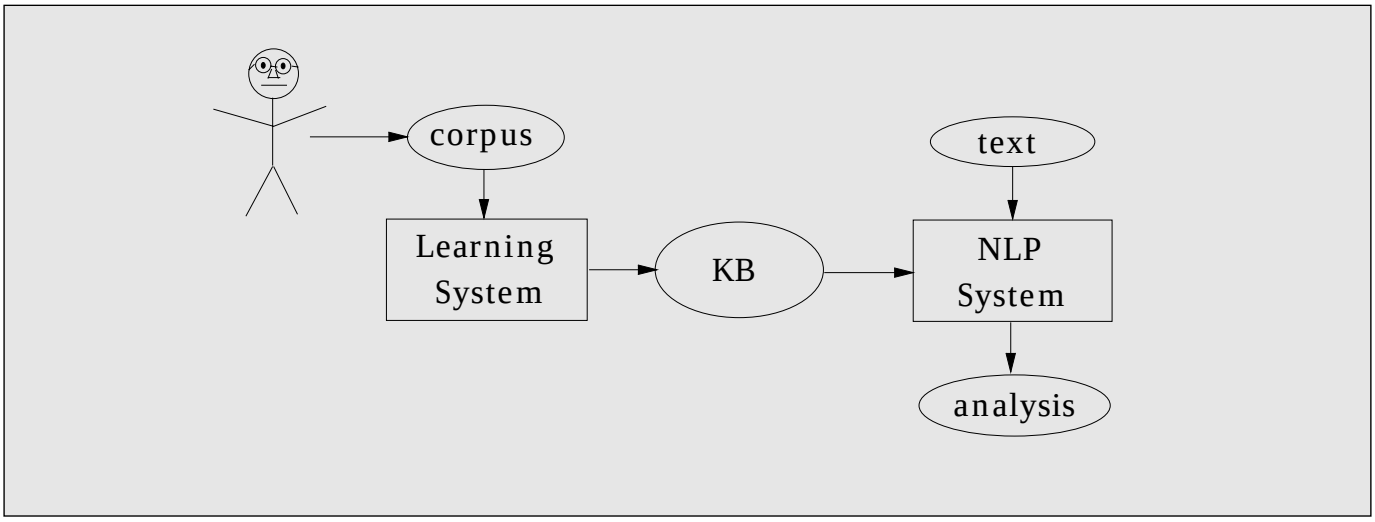

Figure 2. Empirical Natural Language Processing.

ment a word can appear in, as the tool for language study. By clustering words and phrases based on the similarity of their distributional behavior, a great deal could be learned about a language (for example, Kiss [1973], Stolz [1965], Harris [1962], Chatman [1955], Harris [1951], and Wells [1947]). Although the goal of this research was primarily to gain insight into the structure of different languages, this framework parallels that of modern empirical natural language processing: Given a collection of naturally occurring sentences as input, algorithmically acquire useful linguistic information about the language.

Distributional linguistics research began to wane after Chomsky's (1959, 1957) influential work dramatically redefined the goals of linguistics. First, Chomsky made the point that a linguist should not merely be descriptive, discovering morphological, lexical, and syntactic rules for a language, but should turn instead to what he saw as more interesting problems, such as how language is learned by children and what features all languages share in common. These phenomena are far from surface apparent and, therefore, not amenable to a shallow corpus-based study. Second, Chomsky argued against the learnability of language from data, believing that most of language is innate and not learned. Over the years, Chomsky and other generative linguists have uncovered many subtle facts about language that people seem to know somehow and yet that seemingly could not have been learned because of the paucity of data. As researchers discovered that language is much more complex

than was previously thought and as learning the complexities of language from data began to appear hopeless, much of the work on corpus-based–language learning was halted. Chomsky's development of generative linguistics and his critique of existing empirical approaches to language quickly shifted the focus to alternative rationalist methods, with their emphasis on symbolic grammars and innate linguistic knowledge, that is, *universal grammar*. Early AI research in natural language processing adopted this rationalist approach in the sense that it used rule-based representations of grammars and knowledge that were hand coded by the system developer instead of being learned from data.

In the early 1980s, there was some work in automatic induction of lexical and syntactic information from text, based largely on two widely available annotated corpora: the Brown corpus (Marcus, Santorini, and Marcinkiewicz 1993a) and the Lancaster-Oslo–Bergen corpus (Garside, Leech, and Sampson 1987). Empiricism then spread rapidly throughout the natural language–processing community to a large extent the result of seminal research in speech recognition (Waibel and Lee 1990). Like other natural language research of the time, speech research in the 1970s focused on rationalist knowledge-based methods (Klatt 1977; Reddy 1976). However, during the 1980s, research originating at IBM Yorktown resulted in statistical (stochastic) methods based on hidden Markov models (HMMs) that outperformed previous knowledge-based approaches (Rabiner 1989; Bahl, Jelinek, and Mercer 1983). These methods use a probabilistic finite-state machine to model the pronunciation of words and utilize a hill-climbing training algorithm to fit the model parameters to actual speech data. Most current commercial speech-recognition systems use HMMs.

Starting in the late 1980s, the success of statistical methods in speech spread to other areas of natural language processing (Charniak 1993). Much of the initial success came from using the noisy-channel model (figure 3), an approach that had proven highly successful in speech recognition. Basically, the model assumes that language is generated and then passed through a noisy channel, and the resulting noisy data are received. The goal then is to recover the original data from the noisy data. It is fascinating that this simple model has been used successfully in areas of language processing as disparate as spelling correction and machine translation.

One of the first successes of corpus-based learning was in *part-of-speech (POS) tagging*,

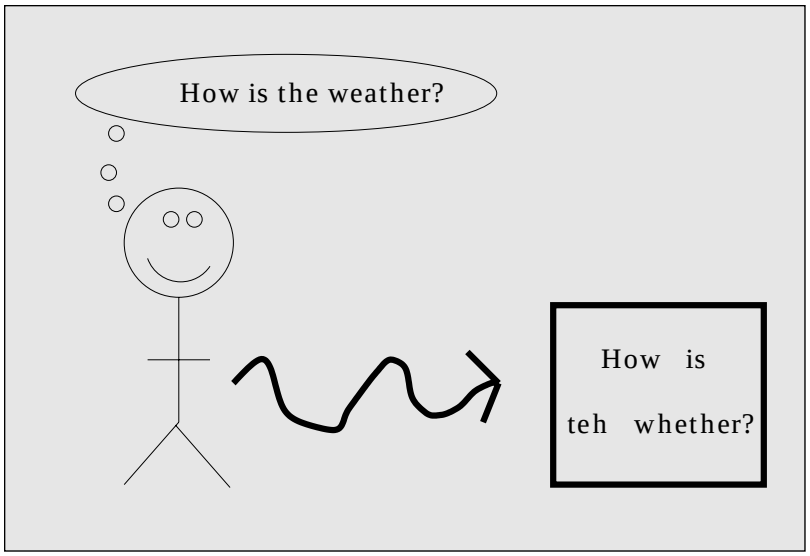

Figure 3. The Noisy-Channel Model.

that is, assigning an appropriate lexical syntactic class (for example, noun, verb, article) to each of the words in a sentence (Merialdo 1994; Church 1988; Bahl and Mercer 1976). A number of techniques can now perform this task at an accuracy close to human performance (>95%), and it is a useful preprocessing step in other tasks such as parsing, speech synthesis, and information retrieval. Another early influential result was that achieved by a statistical approach to machine translation trained and tested on bilingual proceedings of the Canadian parliament (Brown et al. 1990). With the development of *tree banks*, large databases of sentences annotated with syntactic parse trees (Marcus, Santorini, and Marcinkiewicz 1993b), came an increasing body of research on empirical parsing methods, for example, *probabilistic context-free grammars* (PCFGs) (Charniak 1996; Collins 1996; Pereira and Shabes 1992; Lari and Young 1990b). Research on empirical methods is now thriving in a variety of other areas as well, such as word-sense disambiguation (Ng and Lee 1996; Gale, Church, and Yarowsky 1992), prepositional phrase attachment (Hindle and Rooth 1993), semantic analysis (Miller et al. 1997; Zelle and Mooney 1996), anaphora (for example, pronoun) resolution (Aone and Bennett 1995; Cardie 1992), and discourse segmentation (Litman 1996).

In recent years, there have been numerous specialized workshops on issues relating to empirical natural language processing held in coordination with the National Conference on Artificial Intelligence (AAAI-92), the International Joint Conference on Artificial Intelli-

gence (IJCAI-91, IJCAI-95), the AAAI Fall Symposia (1992), the Annual Meeting of the Association for Computational Linguistics (ACL-93–ACL-96), and the International Conference on Computational Linguistics (COLING-96), among others. Last year, the First Conference on Empirical Natural Language Processing (EMNLP-96) was held at the University of Pennsylvania (Brill and Church 1996), and the second EMNLP conference was recently held in coordination with AAAI-97. There are also two relevant special interest groups of the Association for Computational Linguistics: (1) the Special Interest Group on Linguistic Data (SIGDAT): www.cs.jhu.edu/~yarowsky/sigdat.html) and (2) the Special Interest Group on Natural Language Learning (SIGNLL): www.cs.unimaas.nl/signll/.

A final influential change is the desire to move beyond toy domains and systems to more realistic applications and products.

Reasons for the Resurgence of Empiricism

The recent dramatic increase in empirical research has been attributed to various causes. Armstrong-Warwick (1993) mentions that empirical methods offer potential solutions to several related, long-standing problems in natural language processing such as (1) *acquisition,* automatically identifying and coding all the necessary knowledge; (2) *coverage,* accounting for all the phenomena in a given domain or application; (3) *robustness,* accommodating real data that contain noise and aspects not accounted for by the underlying model; and (4) *extensibility,* easily extending or porting a system to a new set of data or a new task or domain.

Automated learning and training techniques allow much of the relevant knowledge to be acquired directly from data rather than laboriously hand coded. If the training data are extensive and represent all the relevant phenomena, empirical methods, which attempt to optimize performance over the complete training set, can help ensure adequate coverage. Unlike symbolic methods that use hard constraints, statistical methods can produce a probability estimate for each analysis, thereby ranking all possible alternatives. This more flexible approach can improve robustness by accommodating noise and always allowing the selection of a preferred analysis even when the underlying model is inadequate.[1] Finally, because empirical methods allow for automatic retraining on additional data or data from a different distribution or a new domain, they can also help improve extensibility.

In addition, Church and Mercer (1993) mention three recent developments that they believe have spurred the resurgence in empiricism: (1) *computing resources,* the availability of relatively inexpensive workstations with sufficient processing and memory resources to analyze large amounts of data; (2) *data resources,* the development and availability of large corpora of linguistic and lexical data for training and testing systems; and (3) *emphasis on applications and evaluation,* industrial and government focus on the development of practical systems that are experimentally evaluated on real data.

Because empirical methods generally require computationally expensive training and testing on large amounts of data, adequate computing resources are needed to support such research. Recently, sufficient computing resources have become available to most researchers. Another requirement of empirical approaches is sufficient linguistic data. For many methods, data also require human annotation with, for example, syntactic tags, parse trees, or word senses. Recently, a number of large corpora have been assembled, appropriately annotated, and made available to the research community. Much of these data are available through organizations such as the Linguistic Data Consortium (LDC) (www.ldc.upenn.edu). Two examples of widely used resources are the Penn tree bank (Marcus, Santorini, and Marcinkiewicz 1993b), which among its data contains syntactic parses for about 40,000 sentences from the *Wall Street Journal,* and WORDNET (Miller 1991) (www.cogsci.princeton.edu/~wn), an English lexical database with word senses linked by kind-of, part-of, and other semantic relations.

A final influential change is the desire to move beyond toy domains and systems to more realistic applications and products. Most industrial research labs have shifted their emphasis from basic to applied research, and government funding agencies have been emphasizing applications and evaluation. The Defense Advanced Research Projects Agency (DARPA) airline travel information system (ATIS) speech program and TIPSTER text program (tipster.org) (including message-understanding conference [MUC]) [Lehnert and Sundheim 1991] and text-retrieval conference [TREC]) were particularly influential in shifting the focus to competitive evaluation of systems on realistic data. These programs developed significant collections of annotated data for testing natural language systems. Because empirical methods can exploit such data for training purposes and thereby automatically tune performance to the given task, they offer a potential advantage in developing competitive systems for such programs. Also, the ability of

empirical methods to quickly develop fairly robust systems that optimize particular numeric criteria for evaluating parsing or retrieval accuracy also give them an edge in such a result-driven environment.

Categories of Empirical Methods

Most of the recent work in empirical natural language processing has involved statistical training techniques for probabilistic models such as HMMs and PCFGs (Charniak 1993). These methods attach probabilities to the transitions of a finite-state machine or the productions of a formal grammar and estimate these probabilistic parameters based on training data. If the training set is preannotated with the structure being learned, learning consists simply of counting various observed events in the training data. If the corpus is not annotated, an estimation-maximization strategy could be used (for example, the forward-backward algorithm for Markov models and the inside-outside algorithm for PCFGs (Baum 1972; Lari and Young 1990a). Novel test examples are then analyzed by determining the most-probable path through the learned automaton or derivation from the learned grammar that generates the given string. Other empirical methods gather and use other statistics such as the frequency of each *n*-word sequence (*n-gram*) appearing in the language.

However, not all empirical methods use probabilistic models. In the 1970s and early 1980s, there were a number of research projects on learning symbolic grammars and parsers for natural language (Berwick 1985; Langley and Carbonell 1985; Anderson 1977; Reeker 1976). Although the resulting systems demonstrated some interesting principles, they were only tested on small sets of artificial data and never had a major impact on mainstream computational linguistics. Although the current resurgence in empiricism has primarily involved probabilistic methods, symbolic learning techniques are also fairly well represented. Of course, all learning methods are statistical in the sense that they make inductive generalizations from data, and the accuracy of these generalizations can be analyzed using the theory of probability and statistics (Kearns and Vazirani 1994); however, nonprobabilistic methods do not explicitly use probabilities in the representation of learned knowledge. Symbolic learning methods that have been used in recent natural language–processing research include transformational-rule induction (Brill 1995, 1993), decision tree induction (Hermjakob and Mooney 1997; Litman 1996; Aone and Bennett 1995), inductive logic programming (Zelle and Mooney 1996), explanation-based learning (Samuelsson and Rayner 1991; Mooney and DeJong 1985), and exemplar (case-based) methods (Ng and Lee 1996; Cardie 1993). These systems automatically acquire knowledge in some form of rules or simply remember specific past examples and make decisions based on similarity to these stored instances.

A claimed advantage of symbolic methods is that the learned knowledge is more perspicuous and interpretable by humans compared to large matrixes of probabilities. Therefore, symbolic methods could provide more insight into the language-understanding process and allow developers to more easily understand, manipulate, and debug the resulting system. A claimed advantage of probabilistic methods is that they provide a continuous ranking of alternative analyses rather than just a single output, and such rankings can productively increase the bandwidth between components of a modular system. For example, if a POS tagger provides nonzero probabilities for several tags (for example, noun: 0.9, verb: 0.08), then a subsequent probabilistic parser can use these probabilities when weighing decisions about phrasal and sentential structure. Evidence from the parser can confirm the tagger's most probable label (for example, confirm the noun interpretation) or overrule it based on stronger evidence from the larger grammatical context (for example, switch to the initially less likely verb interpretation to form a more coherent parse of the complete sentence).

Another major style of empirical methods is neural network, or connectionist. When neural nets were originally popular in the 1950s, most of the research concerned visual pattern recognition, and language learning was not well represented. However, with the revival of neural nets in the 1980s, applications to language were visible. For example, four of the six chapters in the Psychological Processes section of the Rumelhart and McClelland (1986b) parallel distributed processing volumes (an influential early publication in the resurgence of neural nets) concern models of language processing, specifically, speech, reading, morphology, and sentence analysis. Since then, there has been a range of research on training neural networks to perform various language-processing tasks (Miikkulainen 1993; Reilly and Sharkey 1992). Several neural net methods have successfully been applied to speech recognition (Lippman 1989) and the conversion of text to speech (Sejnowski and Rosenberg

Most of the recent work in empirical natural language processing has involved statistical training techniques for probabilistic models such as HMMs and PCFGs.

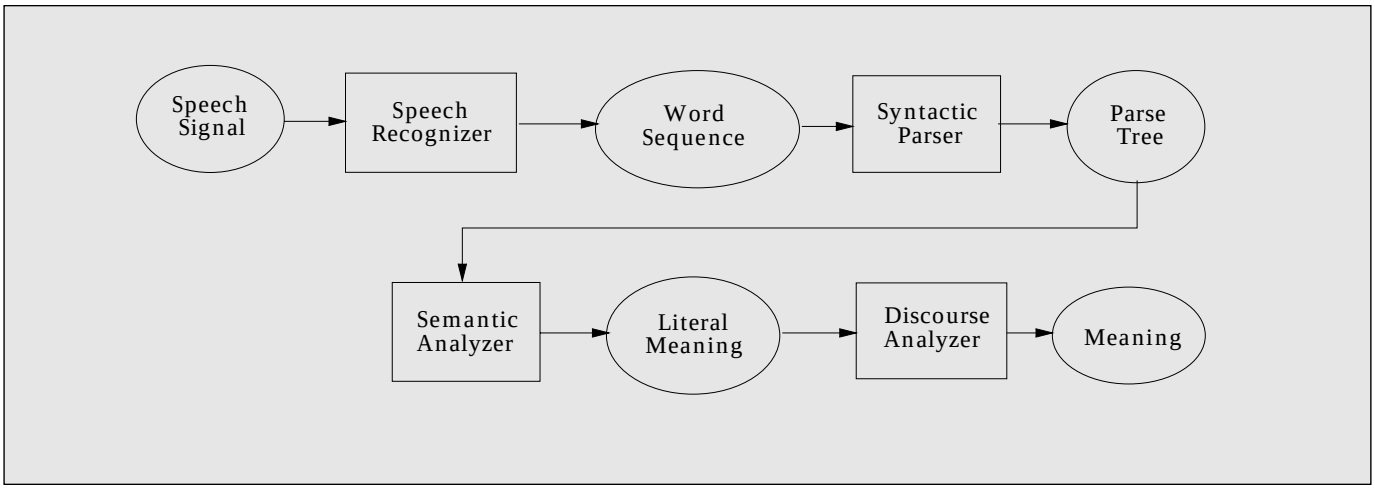

Figure 4. Components of a Natural Language–Processing System.

1987). Other well-studied tasks include the generation of the past tense of English verbs (Plunkett and Marchman 1993; MacWhinney and Leinbach 1991; Rumelhart and McClelland 1986a) and case-role analyses of sentences (Miikkulainen 1996; St. John and McClelland 1990; McClelland and Kawamoto 1986). Typically, neural network research in natural language focuses more on modeling particular psychological or biological phenomena than on producing practical and effective natural language–processing systems. For example, one issue has been modeling the U-shaped learning curve in children's acquisition of English morphology, which refers to the observation that irregulars are first learned correctly, then overregularized, and then finally recorrected. For example, a child uses *went*, then *goed*, then returns to *went*. Because we do not attempt to cover cognitive-modeling aspects of empirical natural language processing in this special issue, neural network research is not reviewed in detail.

A different dimension along which empirical methods vary concerns the type of training data required. Many systems use *supervised* methods and require *annotated* text in which human supervisors have labeled words with parts of speech or semantic senses or have annotated sentences with syntactic parses or semantic representations. Other systems employ unsupervised methods and use raw, unannotated text. *Unsupervised learning* is generally more difficult and requires some method for acquiring feedback indirectly, such as assuming that all sentences encountered in texts are positive examples of grammatical sentences in the language. Because annotating corpora is a difficult, time-consuming task, ideally, unsupervised training is preferable. However, the explicit, detailed feedback provided by supervised training generally results in improved performance (Merialdo 1994; Pereira and Shabes 1992).

Finally, it is important to note that traditional rationalist approaches and empirical methods are not incompatible or incommensurate (Klavans and Resnik 1996). Many empirical systems, particularly those using symbolic methods, exploit traditional representations and algorithms and simply replace handcrafted rules with knowledge automatically extracted from data. A number of empirical parsing methods use a fairly standard shift-reduce framework in which parsing operators construct and manipulate constituents stored in a stack (Miikkulainen 1996; Magerman 1995; Zelle and Mooney 1994; Simmons and Yu 1992). Several empirical information-extraction systems simply replace the hand-coded rules and patterns of an existing symbolic system with ones automatically acquired from examples (Soderland and Lehnert 1994; Riloff 1993). Even probabilistic approaches typically divide the understanding process into traditional subtasks, such as maintaining separate syntactic, semantic, and discourse modules with the standard input and output, but with the addition that multiple interpretations are maintained and each assigned a probability (Miller et al. 1996). Integrating the important linguistic insights and rich representations of rationalist approaches with the previously discussed advantages of empirical methods is an important problem that is attracting increasing attention.

Categories of Language Tasks

Understanding natural language is a complex task and involves many levels of processing and a variety of subtasks. The articles in this collection divide the field of natural language processing as follows: speech recognition and spoken-language analysis, syntactic analysis, semantic analysis, discourse analysis and information extraction, and machine translation. Figure 4 illustrates the first four as components of an overall language-understanding system.

Speech Recognition

Speech recognition concerns mapping a continuous speech signal into a sequence of recognized words. The problem is difficult because of the wide variation in the exact pronunciation of words spoken by different speakers in different contexts. Other problems include homonyms (for example, *pair, pear, pare*), other forms of acoustic ambiguity (for example, *youth in Asia* and *euthanasia*), and the slurring of words (for example, *didja*) that happens in continuous speech.

In the article "Linguistic Knowledge and Empirical Methods in Speech Recognition," Andreas Stolcke presents an overview of the current state of the art in speech recognition. Speech recognition has gained great momentum recently, with many commercial products currently available and research systems rapidly becoming more accurate and robust. Stolcke describes some of these emerging systems and presents the technology that is used in virtually all modern speech-recognition systems. In Miller and Chomsky (1963), arguments were presented against the adequacy of Markov models for natural language. For example, no Markov model would be adequate to capture all long-distance dependencies. In the sentence "the boy eats," a *bigram model* (a model where the state remembers the one previous word) would be sufficient to model the relationship between *boy* and *eats*. However, these words can be arbitrarily far apart, as in "The boy on the hill by the lake in our town . . . eats." Although some recent work has been done to create sophisticated models of language able to handle such dependencies, interestingly the bigram and *trigram* (a model based on a two-word history) are the underlying models of language used in virtually all current speech systems and have proven extremely effective despite their obvious deficiencies.

Syntactic Analysis

Syntactic analysis involves determining the grammatical structure of a sentence, that is, how the words are grouped into constituents such as noun phrases and verb phrases. A subtask of syntactic analysis is assigning a part of speech to each word, such as determining that *saw* acts as a noun in "John bought a saw" and as a verb in "John saw the movie." Another problem of ambiguity in syntactic analysis is *attachment*, such as determining in the sentence "John saw the man on the hill" whether *on the hill* modifies the man or the seeing event. Such ambiguity has a tendency to explode combinatorially. A sentence ending in n prepositional phrases such as "on the hill" or "with a telescope" has at least 2^n syntactic analyses depending on how these phrases are attached (Church and Patil 1982). However, because of syntactic and semantic preferences and constraints, people rarely notice this rampant ambiguity and usually quickly settle on an interpretation based on context.

In "Statistical Techniques for Natural Language Parsing," Eugene Charniak presents an overview of current work in machine learning of syntactic information. There have recently been great improvements in such systems, which can be attributed to two things: (1) publicly available tree banks and (2) grammar lexicalization. The Penn tree bank has released a corpus of 50,000 sentences that have carefully been annotated for syntactic structure by hand. This is a great resource for training and also provides the research community with the ability to readily assess the quality of their programs because researchers can train and test on the exact same sentences. PCFGs have been around for a long time. However, they have one big weakness as a tool for modeling language. Given a PCFG, the probability of a particular parse for a sentence is the product of the probability of all rules used to generate the parse. Given the two sentences along with their parses shown in figure 5, assuming *flew* and *knew* are equally likely as verbs, the PCFG would assign equal probability to these two sentences because they differ in how they were generated only by one rule ($VP \longrightarrow$ *flew* versus $VP \longrightarrow$ *knew*).

The insight that has led to vast improvements in parsing is that by lexicalizing the grammar, much more meaningful statistics can be obtained. For example, a lexicalized parse is shown in figure 6. The idea is that all nodes in the tree contain information about the words in the sentence, not just the nodes immediately above words. In the nonlexicalized parses of figure 5, an *S* node is expanded into an *NP* node and a *VP* node. In a lexicalized parse such as that in figure 6, an *S* node expands into an *airplane NP* node and a *flew VP* node. Because the probability of $S \longrightarrow NP$:

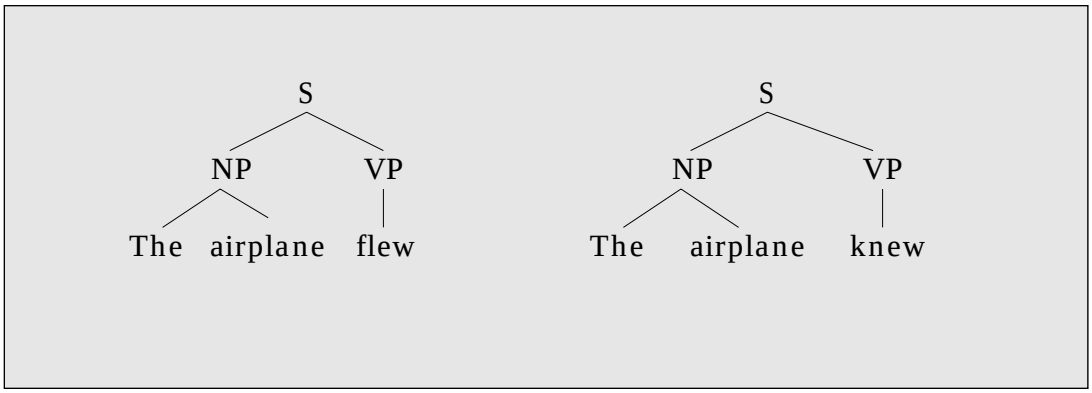

Figure 5. Unlexicalized Parses.

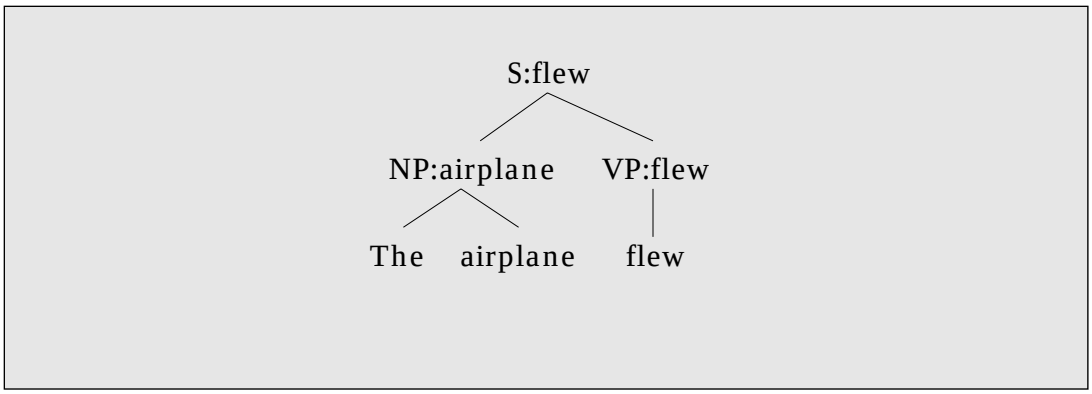

Figure 6. A Lexicalized Parse.

airplane VP : *flew* will be greater than that of *S* $\rightarrow$ *NP* : *airplane VP* : *knew* (airplanes fly, they don't know), the grammar will now give higher probability to the more likely sentence.

Semantic Analysis

Semantic analysis involves mapping a sentence to some sort of meaning representation, for example, a logical expression. In "Corpus-Based Approaches to Semantic Interpretation in Natural Language Processing," Hwee Tou Ng and John Zelle describe recent empirical work on two important subtasks of semantic analysis: (1) word-sense disambiguation and (2) semantic parsing. *Word-sense disambiguation* roughly means deciding which of the possible meanings for a word is correct in a particular context. A classic example is determining whether *pen* refers to a writing instrument or an enclosure in a particular sentence such as "John wrote the letter with a pen" or "John saw the pig in the pen." This step is vital for natural language understanding. It is also important for machine translation, where different senses of a word in one language are translated differently. In speech synthesis, one needs to know the sense of certain words to determine the correct pronunciation (for example, bass guitar versus bass fishing).

Part of semantic parsing involves producing a *case-role* analysis, in which the semantic roles of the entities referred to in a sentence, such as *agent* and *instrument*, are identified. When building a natural language front end to a database, the ability to map a sentence to a formal query language, such as Prolog or SQL, is particularly useful. Activity in semantic parsing has been stimulated recently, in part because of various DARPA-sponsored projects. In one particular project, ATIS, the goal is to convert sentences uttered to a travel agent into SQL queries that can then be used to automatically retrieve the desired information.

Discourse Analysis and Information Extraction

Claire Cardie's article, "Empirical Methods in Information Extraction," describes recent work in discourse analysis. *Discourse analysis* involves determining how larger intersentential context influences the interpretation of a sentence. Her article covers two important sub-

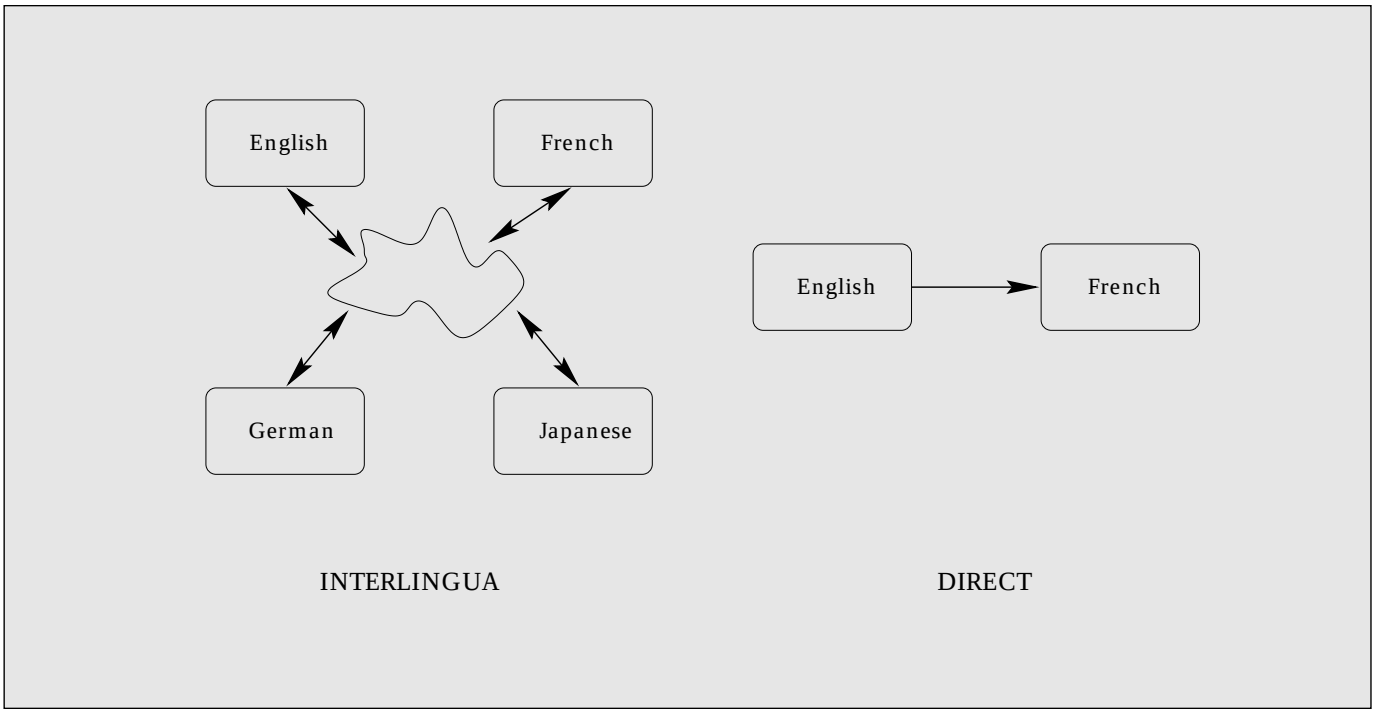

Figure 7. Two Approaches to Machine Translation.

tasks in discourse analysis: (1) coreference resolution and (2) information extraction. *Coreference resolution* is the task of determining what phrases in a document refer to the same thing. One aspect of this problem is pronoun resolution. For example, in the sentence "John wanted a copy of Netscape to run on his PC on the desk in his den; fortunately, his ISP included it in their startup package," a pronoun-resolution algorithm would have to determine that *it* refers to a copy of Netscape rather than PC, desk, or den. More generally, to do successful text analysis and understanding, one needs to identify all noun phrases that corefer within a discourse. For example, in the sentence "Ford bought 100 acres outside Nashville; the company will use the land to build a factory," it must be determined that *Ford* and *the company* corefer, as do *100 acres* and *the land*.

Information extraction is the task of locating specific pieces of data from a natural language document and has been the focus of DARPA's MUC program (DARPA 1993; Lehnert and Sundheim 1991). For example, consider analyzing a message from the newsgroup misc.jobs.offered to extract the employer's name, the location, the type of job, the years of experience required, and so on. The information extracted from a collection of messages could then be stored in a database with fields for each of these slots. Typically, text is first linguistically annotated, and then extraction rules are used to map from annotated text to slot filling. Until recently, these rules were written by hand. By using machine-learning techniques, extraction rules can be learned automatically and achieve performance close to the best manually constructed systems.

Machine Translation

Machine translation involves translating text from one natural language to another, such as translating English to Japanese, or vice versa. One approach uses simple word substitution with some changes in ordering to account for grammatical differences; another is to translate the source language into an underlying meaning representation, or *interlingua*, and then generate the target language from this internal representation. By taking two tightly coupled corpora such as translated books, or loosely coupled corpora such as two stories about the same topic and in different languages, and aligning these corpora to determine which words in one language correspond with which words in the other, a great deal can be learned automatically about word-to-word translations and the relationships between syntactic structures in the two languages (figure 7). Kevin Knight's article "Automating Knowledge Acquisition for Machine Translation" describes recent corpus-based approaches to automatic training of machine-translation programs.

Conclusion

One of the biggest challenges in natural language processing is overcoming the linguistic knowledge-acquisition bottleneck: providing the machine with the linguistic sophistication necessary to perform robust, large-scale natural language processing. Until recently, the only method was to manually encode this linguistic knowledge. Because of the apparent immense complexity of human language, this task has proved to be difficult. Recently, a number of exciting results have shown the feasibility of learning linguistic knowledge automatically from large text corpora. The effectiveness of this approach has been demonstrated across the spectrum of natural language–processing tasks, from low-level tasks such as part-of-speech tagging, word-sense disambiguation, and parsing to high-level tasks such as speech recognition, machine translation, and information extraction. The need for robust language technology is rapidly growing. As computers become ubiquitous in our society, natural language and speech interfaces can make the power of these machines accessible to everybody. With the explosion of online text, being able to effectively sift through these data and extract important facts from them becomes more and more crucial to our productivity. Companies have to rapidly produce reports and manuals in many languages. Hopefully, with the continued development of powerful learning algorithms, faster computers, and ever-growing amounts of online text for training, empirical methods can provide the means to overcome the linguistic knowledge-acquisition bottleneck and make advanced language processing a reality. We hope the articles in this special issue will give the reader an enjoyable sampling of this exciting field.

Acknowledgments

The preparation of the article was partially supported by the National Science Foundation through grants IRI-9310819 to Mooney and IRI-9502312 to Brill. We are grateful to the authors who contributed articles to this special issue.

Notes

1. An ordered set of learned rules can also be utilized to ensure a preferred analysis (Brill 1995).

References

Allen, J. F. 1987. *Natural Language Understanding*. Menlo Park, Calif.: Benjamin/Cummings.

Anderson, J. R. 1977. Induction of Augmented Transition Networks. *Cognitive Science* 1:125–157.

Aone, C., and Bennett, S. W. 1995. Evaluating Automated and Manual Acquisition of Anaphora Resolution Strategies. In Proceedings of the Thirty-Third Annual Meeting of the Association for Computational Linguistics, 122–129. Somerset, N.J.: Association for Computational Linguistics.

Armstrong-Warwick, S. 1993. Preface. *Computational Linguistics* 19(1): iii–iv.

Bahl, L. R., and Mercer, R. L. 1976. Part-of-Speech Assignment by a Statistical Decision Algorithm. In Abstracts of Papers from the IEEE International Symposium on Information Theory, 88–89. Washington, D.C.: IEEE Computer Society.

Bahl, L. R.; Jelinek, F.; and Mercer, R. 1983. A Maximum-Likelihood Approach to Continuous Speech Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 5(2): 179–190.

Bar-Hillel, Y. 1964. *Language and Information*. Reading, Mass.: Addison-Wesley.

Baum, L. 1972. An Inequality and Associated Maximization Technique in Statistical Estimation for Probabilistic Functions of a Markov Process. *Inequalities* 3:1–8.

Berwick, B. 1985. *The Acquisition of Syntactic Knowledge*. Cambridge, Mass.: MIT Press.

Brill, E. 1995. Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part-of-Speech Tagging. *Computational Linguistics* 21(4): 543–565.

Brill, E. 1993. Automatic Grammar Induction and Parsing Free Text: A Transformation-Based Approach. In Proceedings of the Thirty-First Annual Meeting of the Association for Computational Linguistics, 259–265. Somerset, N.J.: Association for Computational Linguistics.

Brill, E., and Church, K., eds. 1996. Paper presented at the Conference on Empirical Methods in Natural Language Processing, 17–18 May, University of Pennsylvania.

Brown, P.; Cocke, J.; Della Pietra, S.; Della Pietra, V.; Jelinek, F.; Lafferty, J.; Mercer, R.; and Roossin, P. 1990. A Statistical Approach to Machine Translation. *Computational Linguistics* 16(2): 79–85.

Cardie, C. 1993. A Case-Based Approach to Knowledge Acquisition for Domain-Specific Sentence Analysis. In Proceedings of the Eleventh National Conference on Artificial Intelligence, 798–803. Menlo Park, Calif.: American Association for Artificial Intelligence.

Cardie, C. 1992. Learning to Disambiguate Relative Pronouns. In Proceedings of the Tenth National Conference on Artificial Intelligence, 38–43. Menlo Park, Calif.: American Association for Artificial Intelligence.

Charniak, E. 1996. Tree-Bank Grammars. In Proceedings of the Thirteenth National Conference on Artificial Intelligence, 1031–1036. Menlo Park, Calif.: American Association for Artificial Intelligence.

Charniak, E. 1993. *Statistical Language Learning*. Cambridge, Mass.: MIT Press.

Chatman, S. 1955. Immediate Constituents and Expansion Analysis. *Word* 11.

Chomsky, N. 1959. Review of Skinner's Verbal Behavior. *Language* 35:26–58.

Chomsky, N. 1957. *Syntactic Structures*. The Hague, The Netherlands: Mouton.

Church, K. 1988. A Stochastic Parts Program and Noun Phrase Parser for Unrestricted Text. In Proceedings of the Second Conference on Applied Natural Language Processing, 136–143. Somerset, N.J.: Association for Computational Linguistics.

Church, K., and Mercer, R. L. 1993. Introduction to the Special Issue on Computational Linguistics Using Large Corpora. *Computational Linguistics* 19(1): 1–24.

Church, K., and Patil, R. 1982. Coping with Syntactic Ambiguity or How to Put the Block in the Box on the Table. *American Journal of Computational Linguistics* 8(3–4): 139–149.

Collins, M. J. 1996. A New Statistical Parser Based on Bigram Lexical Dependencies. In Proceedings of the Thirty-Fourth Annual Meeting of the Association for Computational Linguistics, 184–191. Somerset, N.J.: Association for Computational Linguistics.

DARPA. 1993. *Proceedings of the Fifth DARPA Message-Understanding Evaluation and Conference*. San Francisco, Calif.: Morgan Kaufman.

Gale, W.; Church, K.; and Yarowsky, D. 1992. A Method for Disambiguating Word Senses in a Large Corpus. *Computers and the Humanities* 26:415–439.

Garside, R.; Leech, G.; and Sampson, G. 1987. *The Computational Analysis of English: A Corpus-Based Approach*. London: Longman.

Harris, Z. 1962. *String Analysis of Language Structure*. The Hague, The Netherlands.: Mouton.

Harris, Z. 1951. *Structural Linguistics*. Chicago: University of Chicago Press.

Hermjakob, U., and Mooney, R. J. 1997. Learning Parse and Translation Decisions from Examples with Rich Context. In Proceedings of the Thirty-Fifth Annual Meeting of the Association for Computational Linguistics, 482–489. Somerset, N.J.: Association for Computational Linguistics.

Hindle, D., and Rooth, M. 1993. Structural Ambiguity and Lexical Relations. *Computational Linguistics* 19(1): 103–120.

Kearns, M. J., and Vazirani, U. V. 1994. *An Introduction to Computational Learning Theory*. Cambridge, Mass.: MIT Press.

Kiss, G. 1973. Grammatical Word Classes: A Learning Process and Its Simulation. *Psychology of Learning and Motivation* 7.

Klatt, D. H. 1977. Review of the ARPA Speech-Understanding Project. *The Journal of the Acoustical Society of America* 62(6): 1324–1366.

Klavans, J., and Resnik, P., eds. 1996. *The Balancing Act: Combining Symbolic and Statistical Approaches to Language*. Cambridge, Mass.: MIT Press.

Langley, P., and Carbonell, J. 1985. Language Acquisition and Machine Learning. In *Mechanisms of Language Acquisition*, ed. B. MacWhinney, 115–155. Hillsdale, N.J.: Lawrence Erlbaum.

Lari, K., and Young, S. J. 1990. The Estimation of Stochastic Context-Free Grammars Using the Inside-Outside Algorithm. *Computer Speech and Language* 4:35–56.

Lehnert, W., and Sundheim, B. 1991. A Performance Evaluation of Text-Analysis Technologies. *AI Magazine* 12(3): 81–94.

Lippman, R. P. 1989. Review of Research on Neural Nets for Speech. *Neural Computation* 1(1).

Litman, D. J. 1996. Cue Phrase Classification Using Machine Learning. *Journal of Artificial Intelligence Research* 5:53–95.

McClelland, J. L., and Kawamoto, A. H. 1986. Mechanisms of Sentence Processing: Assigning Roles to Constituents of Sentences. In *Parallel Distributed Processing, Volume 2*, eds. D. E. Rumelhart and J. L. McClelland, 318–362. Cambridge, Mass.: MIT Press.

MacWhinney, B., and Leinbach, J. 1991. Implementations Are Not Conceptualizations: Revising the Verb Model. *Cognition* 40:291–296.

Magerman, D. M. 1995. Statistical Decision Tree Models for Parsing. In Proceedings of the Thirty-Third Annual Meeting of the Association for Computational Linguistics, 276–283. Somerset, N.J.: Association for Computational Linguistics.

Marcus, M.; Santorini, B.; and Marcinkiewicz, M. 1993. Building a Large Annotated Corpus of English: The Penn Tree Bank. *Computational Linguistics* 19(2): 313–330.

Merialdo, B. 1994. Tagging English Text with a Probabilistic Model. *Computational Linguistics* 20(2): 155–172.

Miikkulainen, R. 1996. Subsymbolic Case-Role Analysis of Sentences with Embedded Clauses. *Cognitive Science* 20(1): 47–73.

Miikkulainen, R. 1993. *Subsymbolic Natural Language Processing: An Integrated Model of Scripts, Lexicon, and Memory*. Cambridge, Mass.: MIT Press.

Miller, G. 1991. WORDNET: An Online Lexical Database. *International Journal of Lexicography* 3(4).

Miller, G., and Chomsky, N. 1963. *Finitary Models of Language Users*. New York: Wiley.

Miller, S.; Stallard, D.; Bobrow, R.; and Schwartz, R. 1996. A Fully Statistical Approach to Natural Language Interfaces. In Proceedings of the Thirty-Fourth Annual Meeting of the Association for Computational Linguistics, 55–61. Somerset, N.J.: Association for Computational Linguistics.

Mooney, R. J., and DeJong, G. F. 1985. Learning Schemata for Natural Language Processing. In Proceedings of the Ninth International Joint Conference on Artificial Intelligence, 681–687. Menlo Park, Calif.: International Joint Conferences on Artificial Intelligence.

Ng, H. T., and Lee, H. B. 1996. Integrating Multiple Knowledge Sources to Disambiguate Word Sense: An Exemplar-Based Approach. In Proceedings of the Thirty-Fourth Annual Meeting of the Association for Computational Linguistics, 40–47. Somerset, N.J.: Association for Computational Linguistics.

Pereira, F., and Shabes, Y. 1992. Inside-Outside Reestimation from Partially Bracketed Corpora. In Proceedings of the Thirtieth Annual Meeting of the Association for Computational Linguistics, 128–135.

Somerset, N.J.: Association for Computational Linguistics.

Plunkett, K., and Marchman, V. 1993. From Rote Learning to System Building: Acquiring Verb Morphology in Children and Connectionist Nets. *Cognition* 48(1): 21–69.

Rabiner, L. R. 1989. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE* 77(2): 257–286.

Reddy, D. R. 1976. Speech Recognition by Machine: A Review. *Proceedings of the IEEE* 64(4): 502–531.

Reeker, L. H. 1976. The Computational Study of Language Acquisition. In *Advances in Computers, Volume 15*, eds. M. Yovits and M. Rubinoff, 181–237. New York: Academic.

Reilly, R. G., and Sharkey, N. E., eds. 1992. *Connectionist Approaches to Natural Language Processing*. Hillsdale, N.J.: Lawrence Erlbaum.

Riloff, E. 1993. Automatically Constructing a Dictionary for Information-Extraction Tasks. In Proceedings of the Eleventh National Conference on Artificial Intelligence, 811–816. Menlo Park, Calif.: American Association for Artificial Intelligence.

Rumelhart, D. E., and McClelland, J. 1986a. On Learning the Past Tense of English Verbs. In *Parallel Distributed Processing, Volume 2*, eds. D. E. Rumelhart and J. L. McClelland, 216–271. Cambridge, Mass.: MIT Press.

Rumelhart, D. E., and McClelland, J. L., eds. 1986b. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volumes 1 and 2*. Cambridge, Mass.: MIT Press.

Samuelsson, C., and Rayner, M. 1991. Quantitative Evaluation of Explanation-Based Learning as an Optimization Tool for a Large-Scale Natural Language System. In Proceedings of the Twelfth International Joint Conference on Artificial Intelligence, 609–615. Menlo Park, Calif.: International Joint Conferences on Artificial Intelligence.

Sejnowski, T. J., and Rosenberg, C. 1987. Parallel Networks That Learn to Pronounce English Text. *Complex Systems* 1:145–168.

Shannon, C. 1951. Prediction and Entropy of Printed English. *Bell Systems Technical Journal* 30:50–64.

Simmons, R. F., and Yu, Y. 1992. The Acquisition and Use of Context-Dependent Grammars for English. *Computational Linguistics* 18(4): 391– 418.

Skinner, B. F. 1957. *Verbal Behavior.* New York: Appleton-Century-Crofts.

Soderland, S., and Lehnert, W. 1994. Wrap-Up: A Trainable Discourse Module for Information Extraction. *Journal of Artificial Intelligence Research* 2:131–158.

St. John, M. F., and McClelland, J. L. 1990. Learning and Applying Contextual Constraints in Sentence Comprehension. *Artificial Intelligence* 46:217–257.

Stolz, W. 1965. A Probabilistic Procedure for Grouping Words into Phrases. *Language and Speech* 8.

Waibel, A., and Lee, K. F., eds. 1990. *Readings in Speech Recognition*. San Francisco, Calif.: Morgan Kaufmann.

Waltz, D. L. 1978. An English Language Question-Answering System for a Large Relational Database. *Communications of the Association for Computing Machinery* 21(7): 526–539.

Wells, R. 1947. Immediate Constituents. *Language* 23.

Winograd, T. 1972. *Understanding Natural Language*. San Diego, Calif.: Academic.

Woods, W. A. 1977. Lunar Rocks in Natural English: Explorations in Natural Language Question Answering. In *Linguistic Structures Processing*, ed. A. Zampoli. New York: Elsevier.

Zelle, J. M., and Mooney, R. J. 1996. Learning to Parse Database Queries Using Inductive Logic Programming. In Proceedings of the Thirteenth National Conference on Artificial Intelligence, 1050–1055. Menlo Park, Calif.: American Association for Artificial Intelligence.

Zelle, J. M., and Mooney, R. J. 1994. Inducing Deterministic Prolog Parsers from Tree Banks: A Machine-Learning Approach. In Proceedings of the Twelfth National Conference on Artificial Intelligence, 748–753. Menlo Park, Calif.: American Association for Artificial Intelligence.

Eric Brill is a computer science faculty member and a member of the Center for Language and Speech Processing at Johns Hopkins University. His research interests include natural language processing, machine learning, and speech recognition. His main research goal is to make using computing devices and accessing information a natural and painless task. His e-mail address is brill@cs.jhu.edu.

Raymond J. Mooney is an associate professor of computer sciences at the University of Texas at Austin. He received a B.S. in computer engineering and an M.S. and a Ph.D. in computer science, all from the University of Illinois at Urbana-Champaign. He is an editor of *Machine Learning* and an author of more than 60 technical papers on various aspects of machine learning. His current primary research interest involves using relational learning methods to construct semantic parsers and information-extraction systems from annotated examples. His e-mail address is mooney@cs.utexas.edu.