specifically designed for people who preferred to work *with* the natural numbers, rather than the logarithms of numbers.

The first and most famous of these devices is the one that Napier called rabdologia, but almost everyone else called *Napier's bones.* They obtained this nickname because the better quality sets were made from bone or ivory. In essence, they were simply vertical strips cut from a 10 by 10 multiplication table which could then be rearranged into the order required to produce a single-digit multiplication table for a multi-digit number.

The other two devices never gained any popularity, primarily because they were either too difficult to manufacture or involved concepts that were unintelligible to most. His second device, the *multiplicationis promptuarium,* was a more complex version of the bones. It consisted of two sets of strips, to be stacked on top of and at right angles to one another, to create a multiplication table for any two multi-digit numbers. The third was a form of a table abacus that was set on a chessboard, the rows and columns of which represented places within the binary number system.

Napier died in 1617 in Merchiston, the same town in which he was born 67 years earlier.

Bibliography

1914. Horsburgh, E. M. *Handbook of the Napier Tercentenary Celebration on Modern Instruments and Methods of Calculation.* Edinburgh: The Royal Society of Edinburgh (reprinted by The MIT Press and Tomash Publishers, 1982).

1983. Williams, M. R. "From Napier to Lucas, the Use of Napier's Bones in Calculating Machinery 1617–1900," *The Annals of the History of Computing,* **5**, *3,* 279–296.

1988. Hawkins, W. F., Tomash, E., and Williams, M. R. "The Promptuary Papers," *Annals of the History of Computing,* **10**, *1,* 35–67.

1991. Napier, J. *Rabdology.* (Reprint #15 of the CBI for the History of Computing.) Cambridge, MA: MIT Press.

Michael R. Williams

NATURAL LANGUAGE PROCESSING

For articles on related subjects *see* ARTIFICIAL INTELLIGENCE; DATABASE MANAGEMENT SYSTEM; GRAMMARS; HUMANITIES APPLICATIONS; KNOWLEDGE REPRESENTATION; LOGIC PROGRAMMING; MACHINE TRANSLATION; RELATIONAL DATABASE; and SPEECH RECOGNITION AND SYNTHESIS.

Natural language processing (NLP) refers to computer systems that analyze, attempt to understand, or produce one or more human languages, such as English, Japanese, Italian, or Russian. The input might be text, spoken language, or keyboard input. The task might be to translate to another language, to comprehend and represent the content of text, to build a database or generate summaries, or to maintain a dialogue with a user as part of an interface for database/information retrieval (*q.v.*). This article addresses issues in natural language comprehension and generation from text or keyboard input. Similar techniques can be used for spoken language by adding a system for speech recognition (*see* SPEECH RECOGNITION AND SYNTHESIS).

It is extremely difficult to define how we would ever know that a system actually "understands" language. All we can actually test is whether a system appears to understand language by successfully performing its task. The Turing test (*q.v.*), proposed by Turing (1950) (reprinted in *Computers and Thought* (1963)), has been the classical model. In this test, the system must be indistinguishable from a human when both answer arbitrary interrogation by a human over a terminal. This test has the unfortunate property that, while it sets the ultimate goal, it provides for no intermediate evaluation of work along the way. A growing concern in NLP is with developing more sensitive models of evaluation that can measure progress, given current performance levels. The usual approach is to develop evaluation tests within limited domains to test specific capabilities. For example, in the area of natural language interfaces for data query, statistical performance measures can be determined based on test sets of human-generated questions collected in protocols (*q.v.*) that use another human to simulate the system. It remains an area of active concern, however, as to how more complex systems that can handle extended dialogue can be evaluated.

The principal difficulty in processing natural language is the pervasive ambiguity found at all levels of the problem. For example, all natural languages involve:

- Simple lexical ambiguity (e.g. "duck" can be a noun [the animal] or a verb [to avoid something thrown]).
- Structural or syntactic ambiguity (e.g. in "I saw the man with a telescope," the telescope might be used for the viewing or might be held by the man being observed).
- Semantic ambiguity (e.g. "go" as a verb has well over 10 distinct meanings in any dictionary).
- Pragmatic ambiguity (e.g. "Can you lift that rock?" may be a yes/no question or a request to lift the rock).
- Referential ambiguity (e.g. in "Jack met Sam at the station. He was feeling ill . . . ," it is not clear who is ill, although the remainder of the sentence might suggest a preferred interpretation).

Of course, all these forms of ambiguity may interact, producing an extremely complex interpretation process. It is the prevalence of ambiguity that distinguishes natural languages from precisely defined artificial languages, such as logic and programming languages. It also makes most of the techniques developed in programming language grammars, parsing, and semantics ineffective for NLP unless significantly modified.

Natural Language Database Query Systems: Syntax and Semantics

The most successful NLP systems to date have been front ends to databases. These systems can understand isolated questions dealing with the content of the database; several systems that do so are commercially available. While they have not been precisely evaluated, these systems provide a wide coverage of English questions, including quite complex quantified database queries. The LUNAR system was the first system to develop this technology and serves as the prototype for many current-day commercial systems. The core of the LUNAR system was a syntactic grammar in a formalism called an Augmented Transition Network (ATN) Grammar (Woods, 1970). An ATN is a graphical notation that can be shown to be equivalent to context-free grammars. The exception is the augmentation: each arc in the grammar may have a procedurally defined augmentation that can enforce non-context-free restrictions that provide a representation of the sentence convenient for semantic interpretation. The principal contribution of the LUNAR system was to demonstrate that such augmented systems could retain the efficiency of pure context-free parsing algorithms, yet handle the context-sensitive aspects found in natural language. The architecture of a typical NLP database query system is shown in Fig. 1. Examples of natural language interfaces are TEAM (Grosz *et al.*, 1987) and IRUS (Bates *et al.*, 1986). An example of a natural language query to TEAM is *Which employees earn more than their manager's salary?*, which, after syntactic and semantic processing would result in a query, such as the following, to a relational database:

```
(IN $E EMPLOYEE) (IN $D DEPT) (EQ ($E DEPT)
($D NAME)) (IN $M EMPLOYEE) (EQ ($M NAME)
($D MANAGER)) (GT ($E SALARY) ($M SALARY))
(($E NAME)).
```

The other major development in the area of grammars and parsing for natural language was the development of definite clause grammars (DCGs) based on Prolog (*see* LOGIC PROGRAMMING: LANGUAGES). Prolog offers an efficient mechanism for parsing context-free grammars simply by writing each context-free rule as a Prolog clause. The additional power required to handle natural language is obtained by using variable bindings and unification to add additional restrictions and to build a convenient representation of the sentence for semantic interpretations. Prolog-based systems have the additional advantage that the semantic processing and the database itself, for that matter, can be represented within the same notation. Pereira and Warren (1980) describe this approach in some detail.

New grammatical formalisms that slightly extend context-free grammars are an active area of research in both computational and theoretical linguistics. These theories require finer distinctions than found in the traditional Chomsky Hierarchy (*q.v.*) to characterize their generative power. A good survey of such formalisms has been given by Perrault (1984).

As for semantic processing, the technology is at a considerably less developed stage, and most work is still being done within research prototypes of limited scope; there are very few commercial applications. Within limited-scope domains, such as database query applications, the semantic component is not much more than a translation program from the output of the parser into a database query language. In the more general research systems, the semantic interpretation phase produces a mapping from the parser output to a knowledge representation that supports inference and the later stages of pragmatic interpretation.

Semantic interpreters can be placed into two major classes: the *compositional* and the *noncompositional.* The noncompositional allows arbitrary transformations from the parser output to the final form. The com-positional requires that interpretation rules are applied in accordance with the structure of the parser output. In its strongest form, compositional semantics requires a single semantic interpretation rule for each syntactic rule, and can support simultaneous syntactic and semantic processing while parsing. While

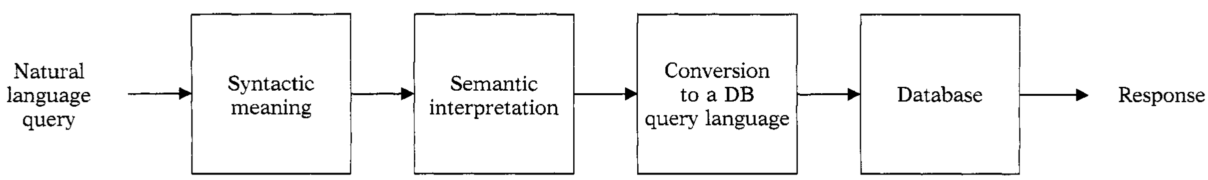

Figure 1. The architecture of an NL database query system.

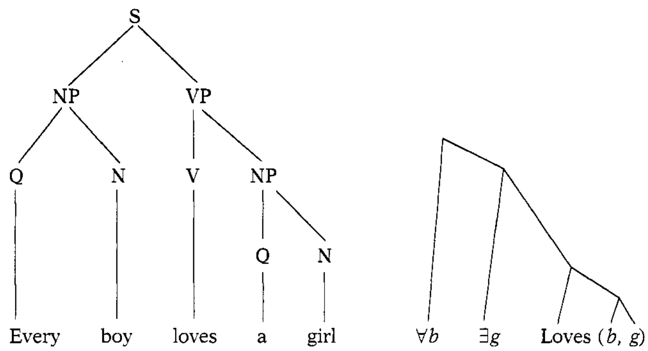

Figure 2. The structure of natural language compared with FOPC quantification. [S: Sentence; NP: Noun Phrase; VP: Verb Phrase; N: Noun; V: Verb; Q: Quantifier].

noncompositional schemes were common in early systems because of their greater power and flexibility, the compositional approaches are now more frequently used because they are significantly more modular and extendable. However, by working within the stricter constraints that a compositional approach imposes, difficult problems arise that require solutions when the syntactic structure of the sentence differs significantly from the structure of the final meaning representation. For example, consider the form of quantification in language versus quantification in a logic in Fig. 2. The structure of the English sentence *Every boy loves a girl* is quite different from the structure of the logic formula $\forall b\ \exists g\ Loves(b, g)$. English puts the quantifiers within the noun phrases, whereas in logic all the quantifiers are outside the scope of the proposition representing the sentence. In addition, there is no natural subpart of the logical formula that corresponds to the interpretation of the noun phrase *a girl.* Rather, the interpretation is spread between the quantifier outside the scope of the predicate and the variable within the scope of the predicate. Proposals for handling this problem within the compositional framework involve introducing an intermediate form of representation that can be built compositionally from the syntactic structure. This representation is then used as input for a second interpretation phase called *quantifier and operator scope determination* that produces the final meaning representation. This identifies yet another source of ambiguity in language: the scope of quantifiers and sentential operators.

Most current NPL systems use a knowledge representation expressively equivalent to or weaker than the first-order predicate calculus (FOPC). But significant aspects of language appear to remain outside the range of first-order logic, and considerable basic research into more expressive formalisms is required before systems will be able to represent the meaning of a significant subset of natural language.

Text Understanding: Pragmatics and World Knowledge

Understanding extended text, such as newspaper articles, paper abstracts, or books, requires significant additions to the capabilities required for question-answering systems discussed above. In particular, there is a strong pragmatic component as well—namely, the use of common everyday knowledge about the world in order to determine the relationships between the sentences in the text. There is a need for significant world knowledge even within single sentences. For example, the sentence *Jack couldn't drive to work because he lost his keys* requires knowledge about cars and keys (e.g. you need a key to start a car, driving to work requires starting the car, etc.). Without this basic knowledge, a system will not be able to determine why Jack couldn't drive to work. The need to use large amounts of common knowledge for natural language understanding was a major focus of the work by Roger Schank and his students. This work focused primarily on representing general knowledge about everyday actions and using this knowledge in interpreting language. These systems could understand simple stories about everyday activities, such as eating in a restaurant or taking public transit. To demonstrate this, they answered questions that required information necessary to understand the story, but not explicitly given in the story. A good description of the techniques is Schank and Riesbeck (1981).

The same motivations were used in the development of the GUS system (Bobrow *et al.*, 1977), which used a representation based on encapsulated knowledge about a specific task called *frames* (*see* KNOWLEDGE REPRESENTATION) to capture knowledge about planning airplane trips. Using the predefined knowledge that captured the structure of the information involved in planning trips, the system "understood" requests in this domain by instantiating the general knowledge to the specific knowledge described in the sentences. Such frame-based approaches still play an important role in current text-understanding systems. Typical application areas for current research systems include understanding messages regarding equipment failures, and extracting the key facts (i.e. those for which slots are defined in the frame) from newspaper articles about takeover attempts in the financial market. The following is an example of the analysis of an article by the SCISOR system, developed by GE Labs (Jacobs and Rau, 1990).

PILLSBURY SURGED 3 3-4 TO 62 IN BIG BOARD COMPOSITE TRADING OF 3.1 MILLION SHARES AFTER BRITAIN'S GRAND METROPOLITAN RAISED ITS HOSTILE TENDER OFFER BY $3 A SHARE TO $63. THE COMPANY PROMPTLY REJECTED THE SWEETENED BID, WHICH CAME AFTER THE TWO SIDES COULDN'T AGREE TO A HIGHER OFFER ON FRIENDLY TERMS OVER THE WEEKEND (Dow Jones News Service, 12 December 1988)

The system extracts the following information from the story:

Corporate-takeover-event:

Target:	Name: Pillsbury-Corporation
Suitor:	Name: Grand Metropolitan
	Country: United Kingdom
Type:	Hostile
State:	Rejected-offer
Price:	$63/share
Stock-exchange:	NYSE
Volume:	3.1 million
Subevent:	Increased-offer:
	Effect-on-stock
	Increment: 3 3-4
	Direction: Up
	Type: Surge
	Final-Value: $62/share
Previous-state:	Negotiated-offer:
	Type: Friendly

It is realistic to expect 90% (combined recall and precision) accuracy for certain useful, carefully constructed tasks, but unrealistic to expect much higher. Many difficulties in reading texts appear when trying to achieve better results, but the most common limitation seems to be the degree of real inference required for understanding. In spite of its fairly sophisticated methods for combining linguistic and world knowledge, SCISOR has very little of the latter.

Dialogue Systems: Discourse and Communication

Systems that can engage in extended natural dialogue present particular challenges in addition to the issues described above. In order to account for dialogue phenomena, a model of conversational interaction needs to be developed. In addition, significant reasoning is required, both to recognize the other speaker's intentions and to produce reasonable responses. The most promising model so far has been based on the notion of speech acts—actions that are performed by speaking, such as requesting, informing, warning, etc. Computational speech act models have been developed by using models of actions and planning developed in work in knowledge representation (Cohen and Perrault, 1979). Plan recognition becomes an important technique for understanding the underlying motivations behind questions. These models can be used to generalize question-answering systems so that the answers generated are more useful and appropriate. These models, however, are not yet capable of explaining dialogues longer than a few sentences. In addition, structural models of discourse have been developed (Grosz and Sidner, 1986) that appear promising. There are currently no systems that come close to having human dialogue capabilities, which involves considerable clarification and correction subdialogues, topic change, and other complexities. It is reasonable to expect that prototype systems will be developed in the next few years that can handle dialogues within limited-topic application domains.

Machine Translation

Machine translation (language translation) was one of the first applications that led to AI work on natural language processing. Machine translation is a very active area of research, especially in Europe and Japan, and is now undergoing a resurgence in the USA. There are two primary approaches. The first is based on defining corresponding lexical, syntactic, and semantic correspondences between a pair of languages, and defining a transducer based on these rules. The second is based on a notion of a language-independent representation or *interlingua* (*cf.* INTERMEDIATE LANGUAGES). To translate, one would parse one language into the interlingua and then from that generate text in the second language. While the second is the more general approach, the most successful systems to date are based on the former techniques. It seems commonly accepted that, except in limited technical domains, high-quality machine translation of general text is either impossible or a very long way off in the future. What is feasible currently, however, is the development of machine translation tools to aid human translators, and the development of translation systems that then require post-editing by a human. While this might seem a failure, using such techniques can in practice significantly increase the productivity of each human translator. There are commercial systems available that offer these abilities, and we can expect considerable growth in the use and development of machine-translation "workstations" in the next decade.

A rudimentary system currently in place is part of the AltaVista search engine on the World Wide Web (*q.v.*). A user who obtains a "hit"—a short possibly relevant text passage received in a "foreign" language—can request automatic translation into the user's native language.

Generation

An issue that arises in dialogue systems, in text summarization applications, and in many machine translation systems is natural language generation (i.e. the production of sentences to describe a given body of knowledge). There are two primary problems in generation: deciding what content needs to be communicated and then deciding how to realize that content in language. The former problem is related to the reasoning abilities of the system, say those required for participating in a dialogue, whereas the latter is related to inverting the parsing and semantic interpretation processes. Typically, generation systems have been developed independently of the understanding component because each component faces a different set of issues. Present-day generation systems can generate paragraph-length text to describe a prespecified body of information in some knowledge representation. The issue of intelligently choosing what knowledge needs to be realized is just beginning to be addressed.

Speech

Another active area of research is aimed at developing natural language systems that use spoken, rather than written, language (*see* SPEECH RECOGNITION AND SYNTHESIS). But there is more to building a spoken language system than combining a speech recognizer with a natural language system. In particular, new uncertainty and ambiguity is introduced, since the parser does not know precisely what the input words are. On the other hand, other sources of information, such as intonation and prosody, are available to aid the interpretation. It is believed that such information will greatly aid discourse processing, as there appear to be strong intonational clues to discourse structure and communicative intent. To solve these problems and to take advantage of the additional information found in spoken language, new methods of integrating speech recognition and natural language systems must be developed. Systems currently in place include email processors that articulate their messages to the recipient, and systems that "listen" and react to a user's spoken commands.

Prospects

Natural language processing should make considerable strides early in this millennium. Large-scale grammars of natural languages are being written, and there is considerable effort in building large English lexicons using automatic techniques. We can expect to see the emergence of quite sophisticated question-answering systems if there is sufficient economic demand for such technology. In the area of text skimming and summarizing, substantial progress should be made in identifying and capturing a specific set of predetermined topics (e.g. a brief summary of the major financial transactions described in the *Wall Street Journal*). Such a system could automatically read the newspaper and build a database of the transactions described, which could then be searched and used to generate short paragraph summaries of the information extracted. We can also expect considerable progress in the area of dialog systems, although such systems in realistic-sized domains will likely remain as research prototypes.

Bibliography

1950. Turing, A. M. "Computing Machinery and Intelligence," *Mind*, **59** (October), 433–460.

1963. Feigenbaum, E. A., and Feldman, J. (eds.) *Computers and Thought.* New York: McGraw-Hill.

1970. Woods, W. A. "Transition Network Grammars for Natural Language Analysis," *Comm. of the ACM*, **13**, *10*, 591–606.

1977. Bobrow, D., Kaplan, R., Kay, M., Norman, D., Thompson, H., and Winograd, T. "GUS: A Frame Driven Dialog System," *Artificial Intelligence*, **8**, 155–173.

1979. Cohen, P., and Perrault, C. R. "Elements of a Plan-based Model of Speech Acts," *Cognitive Science*, **3**, *3*, 177–212.

1980. Pereira, F. C. N., and Warren, D. H. D. "Definite Clause Grammars for Language Analysis—A Survey of the Formalism and a Comparison with Augmented Transition Network Grammars," *Artificial Intelligence*, **13**, *3*, 231–278.

1981. Schank, R. C., and Riesbeck, C. K. *Inside Computer Understanding.* Hillsdale, NJ: Lawrence Erlbaum.

1984. Perrault, C. R. "On the Mathematical Properties of Linguistic Theories," *Computational Linguistics*, **10**, 165–176.

1986. Bates, M., Moser, M. G., and Stallard, D. "The IRUS Transportable Natural Language Database Interface," in *Expert Database Systems* (ed. L. Kershberg), 617–630. Menlo Park, CA: Benjamin/Cummings.

1986. Grosz, B., and Sidner, C. "Attention, Intentions, and the Structure of Discourse," *Computational Linguistics*, **12**, *3*, 175–204.

1986. Grosz, B., Sparck-Jones, K., and Webber, B. *Readings in Natural Language Processing.* San Francisco: Morgan Kaufmann.

1987. Grosz, B. J., Appelt, D., Martin, P., and Pereira, F. "TEAM: An Experiment in the Design of Transportable Natural-Language Interfaces," *Artificial Intelligence*, **32**, *2*, 173–244.

1990. Jacobs, P. S., and Rau, L. F. "SCISOR: A System for Extracting Information from On-line News," *Comm. of the ACM*, **33**, *11*, 88–97.

1992. Shieber, S. *Constraint-Based Grammar Formalisms.* Cambridge, MA: MIT Press.

1995. Allen, J. *Natural Language Understanding*, 2nd Ed. Reading, MA: Addison-Wesley.

1997. Suereth, R. *Developing Natural Language Interfaces: Processing Human Conversations.* New York: McGraw-Hill.

James F. Allen