A linear notation for expressing semantic networks as sets of semantic predicates is outlined. Rule forms are presented for translating from English discourse to semantic predicates. Rules for generating English sentences and for computing causal chains from semantic predicates are described. Each of these rule forms has been tested in an existing question answering system. A brief analysis of rule-based systems is given and a method for designing them is suggested.

Inferential Searches of Knowledge Networks as an Approach to Extensible Language Understanding Systems

David McDonald and Frederick Hayes-Roth
Department of Computer Science
Carnegie-Mellon University Pittsburgh, PA 15213
and (respectively)
The Rand Corporation Santa Monica, CA 90406

A program has been constructed that implements some inference schemes for inferring the meaning of noun-noun, adjective-noun, and agent-verb-object phrases from the constituent words. Dictionary definitions of words are input directly into a semantic network using a context free grammar implemented in a production system. To interpret a novel phrase (e.g., lawn mower, car tire, car wheel) semantically, an intersection search originating from the nodes representing the constituent words is performed in the semantic network. If an intersection is found, a meaning for the phrase is produced; otherwise the phrase is not interpreted. During this interpretation, only knowledge provided by dictionary definitions and parts of speech of words is used.

A Goal-Directed Production System for Story Understanding

Roger C. Schank and Robert Wilensky
Department of Computer Science
Yale University New Haven, CT 06510

Abstract

Using knowledge to make inferences is a crucial aspect of natural language processing. While the notion of a script has been useful in this regard, other forms of knowledge are also needed. In particular, knowledge about goals and knowledge about stories is crucial for story understanding. The understander must know what types of goals characters in a story can have so as to be able to interpret their behavior. Knowledge about the point of a story is needed to direct the understanding process. These forms of knowledge can best be applied by means of a hierarchical production system.

An Expectation-Driven Production System for Natural Language Understanding

Christopher K. Riesbeck
Department of Computer Science
Yale University New Haven, CT 06510

ELI, the English language interpreter for the SAM story understanding system at Yale, is a model of language understanding using productions. Productions are useful because they are flexible, but this flexibility means more work has to be spent controlling and manipulating them. ELI limits the number of productions that have to be manipulated with expectations. Expectations are constraints generated by frame structures that have been built by previously executed productions. Only productions that satisfy existing expectations are used.

On the Use of Programming Knowledge to Understand Informal Process Descriptions[1]

Robert Balzer, Neil Goldman, and David Wile
Information Sciences Institute Marina del Rey, CA 90291

Abstract

The goal of improving and simplifying communication with computers has been pursued largely through the creation and use of better formal languages. This paper investigates an alternative approach by exploring the variety and extent of informal constructs which can be introduced into a formal language without impairing communication. These informal constructs represent the suppression of certain explicit information which must be inferred from the surrounding context.

In general, each informal construct has several possible interpretations, only one of which was intended by the speaker. The system's task is to use the existing context to focus attention on a small ordered subset of the most probable alternatives and to further reduce it by applying any constraints or well-formedness rules. The most probable remaining alternative is selected as the intended one.

Program descriptions were chosen as the example task domain to test this approach because its rules of context and well-formedness are fairly well developed and because we, as computer scientists, are our own domain experts.

Introduction

We believe that current communication between people and computers suffers greatly from an artificial rigidity imposed by formal input languages. While it would be more convenient if natural language could be used, that goal has proved most elusive. Instead, we wish to investigate the use of and justify the need for informal languages for communication with computer systems.

By informal language we mean a language with a formal syntax and semantics that guarantees an unambiguous parse of any input sentence. The semantics are also constructed so that any input sentence composed entirely of complete constructs has a unique interpretation. The informality of the language arises from the use of partial constructs for complete ones; each informal language will have its own set of rules for the kinds of partial constructs allowed.

For each partial construct appearing in the input, the syntax of the language will define the set of possible completions of that construct; the system's task is to select the correct one.

Generally the selection is based on two factors: an ordering of the possibilities based on the context in which the partial construct appears, and a set of well-formedness rules based on the properties of the objects in the construct, the operations being performed on them, and the environment of these operations.

The power of an informal language hence depends upon the use of context and well-formedness rules to select among the possible completions of a partial construct. The stronger the rules for rejecting possibilities and the better the ordering in suggesting acceptable completions, the more information can be suppressed from the input and still remain understandable. This suppression is very important because it focuses attention on the important components of the input, makes it more concise and understandable, and provides an automatic mechanism for maintaining consistency as the environment changes. These features (as well as syntactic variability) are precisely the reasons why natural language is so comfortable to use.

1. This research was supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. DAHC15 72 C 0308.