COMPUTER SCIENCE RESEARCH REVIEW

1986 — 1987

Department of Computer Science
State University of New York at Buffalo
Buffalo, NY 14260

A Computational Theory of Natural-Language Understanding

William J. Rapaport

1. INTRODUCTION

We are undertaking the design and implementation of a computer system that can parse English sentences containing terms that the system does not "know" (i.e., that are not in the system's lexicon), build a semantic-network representation of these sentences, and express its understanding of the newly acquired terms by generating English sentences from the resulting semantic-network database. The system will be a modification of the natural-language processing capabilities of the SNePS Semantic Network Processing System (Shapiro 1979, 1982; Shapiro and Rapaport 1985, 1986). It is intended to test the thesis that symbol-manipulating systems (computers) can "understand" natural language.

The research is concerned with testing a theory about the nature of semantics: that the semantic interpretation of lexical (and other linguistic) items is fundamentally a *syntactic* process. This is admittedly a paradoxical-sounding theory, but one that I believe can shed much light on issues in natural-language understanding, artificial intelligence, and cognitive science. Moreover, formal semantic interpretation of a language L is *normally* a syntactic process, proceeding as follows: An interpretation function is established between syntactical items from L and ontological items from the "world" W that the language describes. But this, in turn, is usually accomplished by describing the world in *another language*, L_W, and showing that these two *languages*, L and L_W, are notational variants by showing that they are isomorphic. (Cf. Rapaport 1985a.)

2. CAN COMPUTERS UNDERSTAND LANGUAGE?

The principal motivation for the present investigation derives from an examination I have undertaken over the past couple of years into an argument purporting to show that computers cannot understand language (or, hence, be intelligent, artificially or otherwise). The argument, due to the philosopher John R. Searle, is a thought-experiment known as the Chinese-Room Argument (Searle 1980). In this experiment, Searle, who knows neither written nor spoken Chinese, is imagined to be locked in a room and supplied with instructions in English that provide an algorithm for processing written Chinese. Native Chinese speakers are stationed outside the room and pass pieces of paper with questions written in Chinese characters into the room. Searle uses these symbols, otherwise meaningless to him, as input and—following only the algorithm—produces, as output, responses written in Chinese characters. He passes these back outside to the native speakers, who find his "answers ... absolutely indistinguishable from those of native Chinese speakers" (Searle 1980: 418). The argument that this experiment is supposed to support has been expressed by Searle as follows:

> [I] still don't understand a word of Chinese and neither does any other digital computer because all the computer has is what I have: a formal program that attaches no meaning, interpretation, or content to any of the symbols. [Therefore,] ... no formal program by itself is sufficient for understanding .... (Searle 1982: 5.)

The Chinese-Room Argument is a variation on the Turing test (Turing 1950), according to which we should be willing to say that a machine can think if it can fool us into believing that it is a human. If the Chinese-language-understanding program passes the Turing test, then it *does* understand Chinese. And indeed it does pass the test, according to the very criteria Searle sets up. So how can Searle conclude that it doesn't understand Chinese? One reason that he offers is that the program doesn't understand because it doesn't "know" what the words and sentences *mean*:

> The reason that no computer program can ever be a mind is simply that a computer program is only syntactical, and minds are more than syntactical. Minds are semantical, in the sense that they have more than a formal structure, they have a content. (Searle 1984: 31.)

That is, meaning—"semantics"—is something over and above mere symbol manipulation—"syntax." Meaning is a relation between symbols and the things in the world that the symbols are supposed to represent or be about. This "aboutness," or *intentionality*, is often cited by philosophers as a feature that only minds possess. So, if AI programs cannot exhibit intentionality, they cannot be said to think or understand in any way.

But there are different ways to provide the links between a program's symbols and things in the world. One way is by means of sensor and effector organs. Stuart C. Shapiro (personal communication; cf. Shapiro and Rapaport 1985) has suggested that all that is needed is a camera and a pointing finger. If the computer running the Chinese-language program (plus image-processing and robotic-manipulation programs) can "see" and "point" to what it is talking about, then surely it has all it needs to "attach meaning" to its symbols.

Searle calls this sort of response to his argument "the Robot Reply." He objects to it on the grounds that if he, Searle, were to be processing all of this new information along with the Chinese-language program, he still would not "know what is going on," because now he would just have more symbols to manipulate: he would still have no direct access to the external world.

But there is another way to provide the link between symbols and things in the world: Even if the system has sensor and effector organs, it must still have internal representations of the external objects, and it is the relations between *these* and its other symbols that constitute meaning for *it*. Searle seems to think that semantics must link the internal symbols with the outside world and that this is something that cannot be programmed. But if this is what semantics must do, it must do it for human beings, too, so we might as well wonder how the link could possibly be forged for us. Either the link between internal representations and the outside world *can* be made for both humans *and* computers, or else semantics is more usefully treated as linking one set of internal symbolic representations with another. On this view, semantics does indeed turn out to be just more symbol manipulation.

Here is Searle's objection to the Robot Reply:

> I see no reason in principle why we couldn't give a machine the capacity to understand English or Chinese, since in an important sense our bodies with our brains are precisely such machines. But ... we could not give such a thing to a machine ... [whose] operation ... is defined solely in terms of computational processes over formally defined elements. (Searle 1980: 422.)

'Computational processes over formally defined elements' is just a more precise phrase for symbol-manipulation. The reason Searle gives for his claim that a machine that just manipulates symbols cannot understand a natural language is that "only something having the same causal powers as brains can have intentionality" (Searle 1980: 423). What, then, are these "causal powers?" All Searle tells us is that they are due to the (human) brain's "biological (that is, chemical and physical) structure" (Searle 1980: 422). But he does not specify precisely what these causal powers are.

Thus, Searle has two main claims: A computer cannot understand natural language because (1) it is not a biological entity, and (2) it is a purely syntactic entity—it can only manipulate symbols, not meanings. In earlier work (Rapaport 1984b, 1985b, 1986a), I have argued that the biological issue is beside the point—that *any* device that "implements" an algorithm (in the technical sense of the computational theory of abstract data types) for successfully processing natural language can be said to *understand* language, no matter how the device is physically constituted.

3. SEMANTIC NETWORK MODELS OF MEANING

The present project is part of the elaboration of an argument that I have sketched (in the works cited) that this device does not need to do semantics in the sense that Searle wants—that a syntactically-based semantics suffices.

The central issues to be considered are these: We understand what someone else says by interpreting it. An interpretation is a map of the other's syntax into our concepts. Similarly, we understand a purely syntactic formal system by interpreting it—by providing a model-theoretic semantics for it. But that is the *human's* interpretation. The Searle-inspired question is: What would it be for such a *formal system* to understand a *human*? In general, what would it be for any "system" (human or computer) to understand any other? The answer to be examined is this: A system understands another by building an internal model (an interpretation) of the other's linguistic output considered as a formal system. But this model is just more syntax.

The natural-language-processing system that is being modified consists of an Augmented Transition Network (ATN) parser-generator (Shapiro 1982), the SNePS Semantic Network Processing System, an English morphological analyzer-synthesizer, and a lexicon consisting of words and information about their grammatical structure. I have used the system in earlier work on belief representation (Rapaport and Shapiro 1984, Rapaport 1984a, Rapaport 1986b), and it is currently being used and extended by members of my research group. The parser component of the ATN grammar takes as input an English sentence or question and the current semantic network, and—using the morphological analyzer and the lexicon—either outputs an updated knowledge base containing a semantic-network representation of the new sentence or performs inferencing on the knowledge base to find or build a representation of an answer to the question. This representation is then passed along to the generator component of the ATN grammar, which expresses it in English. It is, thus, possible to "converse" with the system in (a fragment of) English. Since this is a computer program that manipulates symbols, this process is purely syntactic.

The system's "understanding" of the user's input utterances is represented by its semantic-network knowledge base. This is not yet a complete model of natural-language understanding: there are no facilities for processing visual input, for instance. Part of my theory, however, is that any such additional information would be transduced into an internal representation. This representation might be part of the semantic network; even if not, it would have to interface with it. In either case, it would have to consist of symbols that the system could manipulate. Thus, it would still be syntactic.

The *external* meaning of the nodes in the semantic network—i.e., the user's interpretation of the semantic network—is provided (in part) by nodes representing the utterances that are the input. These nodes are linked to the rest of the semantic network by "LEX" arcs. Thus, a fragment of network consisting of a node with a LEX arc pointing from it to a node labeled with an English word represents the system's concept of that word (cf. Shapiro and Rapaport 1985). The *internal* meaning of a node—i.e., the system's interpretation of it—is given by the *structure* of the network (cf. Carnap 1928, Sect. 14; Quillian 1968). That is, the system's semantic interpretation of a node N at the tail of a LEX arc (indeed, of *any* node) is the complex of nodes and arcs impinging on N. This is an example of a completely contextual theory of meaning, of the sort previously examined by me in Rapaport 1981.

4. MEINONGIAN SEMANTICS

In that paper, I investigated the use of "Meinongian objects" in natural-language semantics. These are the objects of thought—the things we think and talk about, even if they do not exist. They are typically said to be "constituted" by properties; e.g., the meaning of 'bachelor' might be the object: <being male, being unmarried>. This theory can be extended to provide a theoretical foundation for the present project, in the following way.

We can distinguish two notions of the "context" in which a linguistic expression appears: A broad notion includes intonation, information about the dialect being spoken, information about the speaker's and hearer's beliefs, etc. A narrower notion is that part of the broader one consisting of the sentence- or utterance-fragments "surrounding" the expression whose meaning we are interested in.

Let us consider those narrower contexts that are open sentences (i.e., propositional functions). If we extend the notion of a Meinongian object of thought so that it is constituted, not by properties, but by these contexts, we can then replace <being male, being unmarried> by: < ... is male,

We could then say that the meaning of an expression for speaker S in context C is a certain "extended Meinongian object" constituted by those open sentences that are fragments of utterances heard by S in the past. (In the present application, S can be taken as the computer system.) There will have to be some limitations on the constituents. E.g., after a period of time, it seems reasonable that not all new contexts will add to or change a meaning for S; so those that do not change the meaning need not be constituents. Not all heard contexts will be remembered or recognized by S, though some "forgotten" or "unrecognized" ones might be stored in or recognized by S's "unconscious"; so those contexts that are completely eradicated need not be constituents. As an example, the meaning of 'bachelor' for S in C might be:

< ... s are unmarried, John is a ... , that guy is a ... , no women are ... s, ... s are men, [etc.]>.

Two words that appeared to mean the same thing on this theory could be distinguished via intensional or even quotational contexts.

An advantage of the extended Meinongian theory for this project is that it can be used in accounts of language acquisition: for, as we learn our language (which we do continually), we add to or change the contexts constituting the meanings of words for us. The broader notion of context could also be used: the meaning of an expression *e* for S in C would then be constituted by (almost all) the contexts in the broader sense in which *e* was previously used in S's presence. (The 'previously' helps avoid circularity.) In later work (Rapaport 1985a; Shapiro and Rapaport 1985, 1986), I showed how the nodes of SNePS networks could be interpreted as Meinongian objects. Thus, the meanings of words will be the nodes of the semantic network *together with their links to other nodes.*

The present project is to test this theory by experimenting with a parser that "learns" new words and refines its "knowledge" of these words by updating its semantic-network knowledge base. 'Learning' should be understood in a restricted sense. The system will *acquire* new words. Whether this constitutes *learning* is an open question, and one that will be addressed at the end of the project. E.g., I am *not* concerned with *formal* theories of language acquisition or inductive inference. Rather, I am concerned with the ability of the system to *understand* language. Nevertheless, some work done under the rubric of "learning" may be relevant to the project. In particular, there have been investigations on learning new words and idioms (Haas and Hendrix 1983, Zernik and Dyer 1985).

5. IMPLEMENTATION

One member of our research group has written a preliminary version of an interactive lexicon editor that provides the parser with the ability to ask the user for grammatical information about new words (Weekes 1985). Thus, if a sentence containing a term that is not in the lexicon is to be parsed, the parser *first* requests the needed grammatical information from the user and *then* parses the sentence. The user is also asked whether the new information should be permanently added to the lexicon. The present research consists of the following modifications:

5.1. The lexicon editor is being made more robust. This requires (1) a complete and formal specification of the grammatical categories that our parser needs to handle; (2) an interface between the editor and the morphological analyzer, to enable the system to make full use of the latter's capabilities; and (3) the ability for the parser to call the editor in the *middle* of parsing—i.e., at the time the new word is encountered—rather than at the beginning. The editor will then need to be fully tested.

5.2. The next step is to modify the parser so that it can provide its own answers to the editor's questions, bypassing the user. When an ATN grammar processes a word, it must first determine the grammatical category of the word, in order to decide which of several paths it must take in the parsing process. If a word is of more than one category, the ATN chooses one (in a pre-determined manner) and continues parsing; if the parse fails, the system backtracks and tries an alternative category.

Thus, the first thing that the modified system should do is "guess" the unfamiliar word's likely grammatical category—with the morphological analyzer helping in the analysis—as determined by the choice of paths open to it. E.g., if there are two paths open, one for a verb, one for an adjective, then the system might determine (perhaps on the basis of morphological information) that the current word is a verb, add that information to the lexicon, and continue processing; if the parse fails, it would backtrack, change the lexicon to indicate that the word is an adjective, and continue.

5.3. Next, the system must be modified to be able to *refine* its guesses by processing further uses of the word in other contexts. Since each successful parse adds information to the semantic-network knowledge base, such information should also be able to be used at this stage of processing to aid the system in understanding the input.

5.4. The heart of the project will be the ability of the system to determine the *semantic* interpretation of the new word in terms of the semantic-network knowledge base. Thus, we will want the system to be able to describe—in terms of the node's location in the semantic network—the node whose LEX arc points to the word. This will be done by having the system *generate*, in response to questions asked by the user, one or more English expressions describing the node.

Currently, the system is capable of doing a small amount of generation from certain nodes, and it is being modified to be able to generate from *any* node. We need, however, to be able to generate more information about each node. For instance, if the system has been told that John is rich and speaks French, then the node representing John can be expressed in English in a variety of ways: John; rich John; rich, French-speaking John; etc. Some of these are more appropriate for some purposes than others; we shall investigate heuristics for choosing among potential generations.

Ultimately, we would like the system to be able to give a semantic *definition* of each node representing a lexical entry, in terms of its location in the network. This can be done using the SNeBR belief-revision system (Martins 1983) and the SNePS Inference Package, according to the following general algorithm presented in Rapaport 1981:

Consider the word 'bachelor' and all "synthetic relations" involving it (i.e., those propositions in the network that have the word as a component but that are not "analytically true"; e.g., that John is a bachelor, that Barbara isn't a bachelor, that bachelors are unmarried, that no women are bachelors, that bachelors are men, etc.; but *not* that bachelors are bachelors). Find among them those that have the largest number of others among them as deductive consequents. Call these "definitional." Thus, e.g., while 'Bachelors are unmarried males' expresses a synthetic proposition, since all the other meanings (or "semantic relations") of 'bachelor' can be deduced from it (along with relevant other premises, e.g., about John, 'John,' etc.), it holds a central place in the logical network of these relations and so may be termed definitional.

6. SIGNIFICANCE

6.1. The modified system that will be built will provide a computational "laboratory" for testing the theory of natural-language understanding that I have outlined. In this way, it will be a contribution to contemporary issues in Philosophy of Mind and Cognitive Science, and, perhaps, to Machine Learning.

6.2. The modified system will also provide a useful and robust tool for investigations into natural-language processing *per se*, and can be used as a natural-language front-end for SNePS. It is, thus, also a contribution to Computational Linguistics.

6.3. Finally, SNePS and its natural-language-processing components are currently being used in two ongoing research programs at the University at Buffalo:

(a) I am the Principal Investigator for an NSF research project on "Logical Foundations for Belief Representation" (IST 8504713). This project involves the construction of a system for taking

belief reports expressed in natural language and representing them in a logically and philosophically correct way. The modifications to the natural-language-processing components of SNePS that are described here will be of great value to the NSF project. Conversely, the NSF project can provide a source of examples for this one.

(b) A subgroup (of which I am a participant) of the SUNY Buffalo Graduate Group in Cognitive Science is developing a psychologically and linguistically valid computational model of a reader of narrative text. The underlying claim of the group's project is that the system we build will understand spatial, temporal, and focal-character deictic information in the text. The research described here will be an important *tool* for the Cognitive Science project as well as provide theoretical and experimental *support* for our claim that such a system does in fact "understand."

REFERENCES

(1) Carnap, Rudolf (1928), *The Logical Structure of the World*, R. A. George (trans.) (Berkeley: Univ. of California Press, 1967).

(2) Haas, N., and Hendrix, G. (1983), "Learning by Being Told: Acquiring Knowledge for Information Management," in R. Michalski, J. Carbonell, T. Mitchell, *Machine Learning* (Palo Alto, CA: Tioga): 405-27.

(3) Martins, João (1983), "Reasoning in Multiple Belief Spaces," Technical Report No. 203, SUNY Buffalo Dept. of Computer Science.

(4) Quillian, M. Ross (1968), "Semantic Memory," in M. Minsky (ed.), *Semantic Information Processing* (Cambridge: MIT Press): 227-66.

(5) Rapaport, William J. (1978), "Meinongian Theories and a Russellian Paradox," *Noûs* 12: 153-80; errata, *Noûs* 13(1979)125.

(6) Rapaport, William J. (1981), "How to Make the World Fit Our Language: An Essay in Meinongian Semantics," *Grazer Philosophische Studien* 14: 1-21.

(7) Rapaport, William J. (1984a), "Belief Representation and Quasi-Indicators," SUNY Buffalo Dept. of Computer Science Technical Report No. 215.

(8) Rapaport, William (1984b), "Searle's Experiments with Thought," SUNY Buffalo Dept. of Computer Science Technical Report No. 216; forthcoming in *Philosophy of Science*, 53 (1986).

(9) Rapaport, William J. (1985a), "Meinongian Semantics for Propositional Semantic Networks," *Proc. 23rd Annual Meeting Assoc. for Computational Linguistics*, (Univ. of Chicago) (Morristown, NJ: Assoc. for Comp. Ling.): 43-48.

(10) Rapaport, William J. (1985b), "Machine Understanding and Data Abstraction in Searle's Chinese Room," *Proc. 7th Annual Meeting Cognitive Science Soc.* (Univ. of California at Irvine) (Hillside, NJ: Lawrence Erlbaum): 341-45.

(11) Rapaport, William J. (1986a), "Philosophy, Artificial Intelligence, and the Chinese-Room Argument," *Abacus* 3(Summer 1986)6-17

(12) Rapaport, William J. (1986b), "Logical Foundations for Belief Representation," *Cognitive Science*, 10: 371-422.

(13) Rapaport, William J., and Shapiro, Stuart C. (1984), "Quasi-Indexical Reference in Propositional Semantic Networks," *Proc. 10th Intl. Conf. Computational Linguistics* (COLING-84): 65-70.

(14) Searle, John R. (1980), "Minds, Brains, and Programs," *Behavioral and Brain Sciences* 32: 417-57.

(15) Searle, John R. (1981), "Analytic Philosophy and Mental Phenomena," *Midwest Studies in Philosophy* 6: 405-23.

(16) Searle, John R. (1982), "The Myth of the Computer," *New York Review of Books* (29 April 1982): 3-6.

(17) Searle, John R. (1983), *Intentionality: An Essay in the Philosophy of Mind* (Cambridge: Cambridge University Press).

(18) Searle, John R. (1984), *Minds, Brains and Science* (Cambridge, MA: Harvard University Press).

(19) Shapiro, Stuart C. (1979), "The SNePS Semantic Network Processing System," in N. V. Findler (ed.), *Associative Networks: The Representation and Use of Knowledge by Computers* (New York: Academic Press): 179-203.

(20) Shapiro, Stuart C. (1982), "Generalized Augmented Transition Network Grammars for Generation from Semantic Networks," *American J. Computational Linguistics* 8.1(January-March 1982)12-25.

(21) Shapiro, Stuart C., and Rapaport, William J. (1985), "SNePS Considered as a Fully Intensional Propositional Semantic Network," SUNY Buffalo Department of Computer Science Technical Report No. 85-15; forthcoming in G. McCalla and N. Cercone (eds.), *The Knowledge Frontier* (Berlin: Springer-Verlag).

(22) Shapiro, Stuart C., and Rapaport, William J. (1986), "SNePS Considered as a Fully Intensional Propositional Semantic Network," *Proc. 5th Nat'l. Conf. on Artificial Intelligence (AAAI-86; Philadelphia)*, (Los Altos, CA: Morgan Kaufmann), Vol. 1: 278-283.

(23) Turing, A. M., "Computing Machinery and Intelligence," *Mind* 59(1950); reprinted in A. R. Anderson (ed.), *Minds and Machines* (Englewood Cliffs, NJ: Prentice-Hall, 1964): 4-30.

(24) Weekes, Ronald (1985), "Parsing and Generating Using an ATN Grammar," preliminary M.S. project report, SUNY Buffalo Dept. of Computer Science.

(25) Zernik, Uri, and Dyer, Michael G. (1985), "Towards a Self-Extending Lexicon," *Proc. Assoc. for Computational Linguistics* 23: 284-92.