PANEL ON NATURAL LANGUAGE PROCESSING

Roger Schank (chairman); Eugene Charniak; Yorick Wilks; Terry Winograd; William Woods

Four questions were posed to the panelists. Some panelists chose not to answer each question. The questions were:

1) To what extent is natural language processing a separate field? Would it be wrong to say that AI and NLP are identical fields? Can problem solving be separated from NLP, for example?

2) Has frame theory been a significant contribution to NLP? Is the organization of world knowledge the fundamental problem in NLP?

3) Is syntax a dead issue? What low level language problems remain to be solved?

4) Why are new programming or representation languages necessary for NLP?

QI: To what extent is natural language processing a separate field? Would it be wrong to say that AI and NLP are identical fields? Can problem solving be separated from NLP, for example?

EC: Natural language processing has been a sub-branch of Artificial Intelligence since the early days of the field. For most of this time however, it has been slightly on the periphery, partially due to its close relation to linguistics, a relation not shared by other sub-branches, and partially due to the fact that artificial Intelligence was seen as nearly synonymous with problem solving and heuristic search. This has changed as of late and It now seems clear that language processing is a central topic within AI. That is to say, many NLP researchers have been finding that AI problems occur in NLP as well.

Minsky (1976) tacitly assumes a close connection between the problem of visual recognition (of, say, a room) and language "recognition" (as in recognizing that certain story circumstances are instances of a birthday party). This assumption has been extended by Bobrow and Winograd (1976) who see practically all of AI in terms of the recognition problem. Yet other researchers have commented on the role of problem solving (Rieger 1976) (Schank and Abelson 1975) or search (Charniak 1977) in NLP. There are still aspects of NLP which are unique to it within AI (grammars) and problems in other domains which do not enter NLP (line finding), but it seems safe to say that NLP is well situated in the heart of AI.

Whether NLP is deservingly situated in the heart of AI is a more difficult question, as It depends on whether one sees the extension of NLP into questions of knowledge representation and problem solving as well motivated. That it <u>is</u> well motivated can be supported by one or more of the following premises.

a) These problems are important whether or not they are NLP in a strict sense. Furthermore, since many of these issues are most easily approached from the viewpoint of NLP it makes sense to "extend" NLP to these issues.

b) If one is concerned with not merely "analysing" a sentence, but really "understanding" it, I do not see how these problems can be avoided. Strictly speaking this would depend on one's definition of "understand", but it is surely not coincidence that all of the standard tests for natural language comprehension assume that one has a large body of common sense knowledge, and that one can do problem solving with it.

c) Some more obviously "linguistic" processes like ambiguity and reference resolution depend on these processes.

These premises are not mutually exclusive, and I would support all of them, but any one would do to support the current view of NLP held in AI.

RS: Natural Language Processing has been only a peripheral part of AI until the last few years. In the 1969 IJCAI only 10% of the papers were on NLP. In 1975 that percentage was 23%. When I was looking for a job in 1968, I talked to the Stanford AI lab about writing a program that would talk to their hand-eye system in English. There was no interest in the project by the lab.

Recently things have changed. Part of the change was due to the success of Winograd's program that did what the Stanford AI lab said was uninteresting. The change has come I think from something larger than any one program, however. Lately, researchers in AI have started talking about human process and programs that simulate those processes, much more frequently than they ever did. Researchers are starting to understand that tour-de-forces in programming are interesting but non-extendable.

To go back to the example I cited earlier, the AI lab at Stanford was interested in Winograd's tour-de-force. On the other hand, it doesn't seem Winograd was interested in it as much as Stanford was. In any case he didn't pursue it. He (as well as the majority of members of this panel) has become more interested in what the general issues are. How people do cognitive processing is much more a part of AI than it ever was before.

For that reason AI and NLP are more and more becoming nearly identical fields, as the AI people recognize that how people use and represent knowledge is the key issue in the field and as the NLP people realize that that same issue is at the root of the problems of language processing.

Researchers in NLP have become less and less concerned with language issues per se. We are more interested in inferencing and memory models for example. We seem lately to be coming against the same problems that everyone else in AI has; knowledge representation; processing of goals; planning and so on. I believe however that our solutions to these problems will continue to be different than those already proposed in AI. Whether AI shifts over to our solutions, or whether we continue to go on our merry way, it seems that a radical change from the initial conception in AI of NLP as being some peripheral (albeit difficult) field has occurred. AI and NLP are becoming fields that aim to model human cognition. The more this becomes the case, the more the particular input and output devices being used to gather and transmit input to a system that understands will become side issues.

YW: The superficial answer is that NLP must remain a separate field, if only because requirements of text processing, like spelling correction, will have no analogue in, say, visual processing.

However, under the question are, I think, two deeper questions:

a) Will different AI activities (language, vision, physical manipulation) ultimately require access to the same knowledge base?

b) If knowledge bases for different activities are not independent, is one more fundamental than another?

On (a) I remain unconvinced by the "they must have the same base" arguments—it is not at all clear to me that my physical ability to drive a car and my linguistic knowledge (or knowledge accessed in talk about cars, rather) are, or needs be, the same (Viz: one's difficulty in describing easy physical tasks). Evolutionary arguments (Gregory: language came later than vision—therefore visual structures were used for language) and those from dogs, say, that see but do not talk, seem to me very weak, and to lead to no clear, undisputable, consequences for constructing AI systmes.

On (b) I believe that knowledge is dependent on language, rather than vice-versa (as seems the norm In AI beliefs, see McCarthy). This is hard to justify without bringing in the whole of linguistic philosphy, but as a very high-level psychological assumption it does, I believe, have empitical consequences for how we should construct AI systems.

TW: The fact that these questions are posed for a panel like this is a good indication of how immature our science still is. Imagine a time centuries ago when a similar panel gathered at the conference on celestial mechanics to discuss questions like: To what extent is astronomy a separate field? Would it be wrong to say that physics and astronomy are identical fields? Can angular momentum be separated from astronomy, for example?

The answers to the questions (in both forms) are "To a large extent," "Yes," and "It depends on what you mean by 'separate'." AI is the general study of those aspects of cognition which are common to all physical symbol systems, including humans and computers. As such, it covers a wide range of cognitive processes, each of which embodies the general principles, and each of which has special features of its own. AI and NLP are not identical any more than AI and vision, or AI and game playing, or AI and medical diagnosis. Human language poses a special set of problems, having to do with creating and interpreting structured symbolic objects which can be conveyed over a limited channel (sequence of speech sounds or marks), and which are intended to communicate from one thinking being to another. In this aspect, it is quite different from any of the other AI topics listed above. The structure of language is a result of its function, and by reducing it to what it has in common with other AI areas, we lose sight of its unique features.

The fact that the questions are posed as they are reflects a historical situation In the development of AI. In fact, most of the difficult Issues with which researchers in NLP must grapple are not language issues at all, but more fundamental issues of representation and cognitive processing. These include Issues such as: the role of primitives in representation; the use of frame-like constructs in reasoning; the nature of plans and their relation to actions; the structuring of knowledge for problem solving and deduction; the amount of inferencing done with new knowledge; etc. etc. A quick scan through the literature in AI natural language work will show a high proportion of the verbiage devoted to these rather than the issues of language.

This is a necessary step. It would be as impossible to develop a satisfactory theory of language without having an understanding of general cognitive processing as it would be to have a satisfactory theory of celestial motions without basing it on theories of mechanics. We who do natural language research are in a good position to examine these questions. By having a specific set of problems to deal with, we are put into contact with the broader issues in a way which would be impossible through sheer top-down musing. It will almost certainly remain true for many years to come that research in natural language will be intertwined with research in broader issues of cognition and representation. But in doing this kind of mixed research, we should not lose sight of the fact that we are really wearing two hats, and that language is only a partial reflection of the range of issues in AI.

Q2: Has frame theory been a significant contribution to NLP? Is the organization of world knowledge the fundamental problem in NLP?

RS: Frame theory may not be the particular solution to the problems of NLP, but its significance seems clear. Natural Language Processing will be successful precisely when an adequate theory of the organization and representation of knowledge has been fully worked out. Frames are one suggestion for how to organize world knowledge. They will probably, when fully worked out, only help to solve one aspect of the organization of knowledge problem, but that is a significant aspect indeed.

Scripts, our particular instantiation of frame theory, will in no sense solve all of the problems in NLP. But it seems clear that a good deal of human functioning is script-based, and we have gone fairly far using scripts to understand newspaper stories.

YW: I don't think it has yet, but it well may do so. There has been a rush of ingenious suggestions about, and implementations of, frame systems; but not yet enough thought about what claims are being made, and whether or not they are true. My instinct is to answer "no" to the second question, simply because I can imagine someone having organized world knowledge perfectly, but having got no farther with NLP. Conversely, we are all pretty good at NLP, but have pretty bad organization of world knowledge in many respects. I cannot see that "organization of world knowledge" is a concept that makes much sense, <u>as such</u>, and wholly independent of particular tasks and purposes.

However, it remains true that world knowledge organization is a growth point in NLP at the moment, and will probably remain so for some time. My own hunch is that the use of world knowledge in NLP (at high levels, that is) is going to pay off in connexion with robust systems that run on texts where they <u>don't</u> already know all the word senses etc. And the question here that I am most unsure of the answer to is, what is the role of high-level knowledge structures, like frames, in actually <u>parsing</u> input text?

TW: The second half of this question is another form of the questions above. It is both true and patently false, depending on how "problem in NLP" is interpreted. If we are interested in building systems or models which reflect the processes of language production or comprehension, then the answer is "Yes." When a person uses language, he or she is not simply using a "language faculty", but is making use of a full range of mental operations, of which only some are directly linguistic. If we wish to model this behavior, we must model all of these operations, and in doing so the organization of world knowledge provides the largest stumbling block.

On the other hand, if we view "NLP" as describing a scientific enterprise, then the organization of world knowledge is not a problem within Its domain at all. It is a problem in the more general domain of AI, and one whose solutions will form a basis for attacking problems in NLP, which deal with the structure of language and communication.

The first half of the question raises two issues. The phrase "contribution to NLP" is subject to all of the problems discussion above, and I won't go into them again. The phrase "frame theory" raises the question "What is a frame theory?" As far as I can see, none of the work which goes by this name is really a "theory" at all. There are two levels at which it has been couched: a general intuition, and a set of specific mechanisms. At the intuitive level, it has bee been of great importance. The attention of researchers has been focussed on a different set of issues from those which were predominant in the older, more atomistic ways of thinking about knowledge and meaning. Issues of memory structure, chunking, accessibility, and pattern-related control structure are beginning to make inroads into a literature which previously focussed on predicates, primitives, parsers, and uniformly applicable algorithms. At the detailed level, I think we are far from having satisfactory notions of how to apply the intuitions. There are a number of simple ideas, such as the use of explicit cross linkages to switch between frames (Minsky), the notion of scripts as linear sequences of stereotypical events

(Schank), the association of specific procedures with parts of a declarative frame structure (Winograd), and the need to view a single object as an instantiation of multiple frames (Bobrow and Winograd), but all of these are over-simplified and lack the kind of depth which will make them the center of a "theory". They are all useful, but do not constitute a theory any more than a set of specific techniques for differentiation is a theory of calculus.

Is syntax a dead issue? What low level language problems remain to be solved?

Syntax is by no means a dead issue, but it is by no means clear what the issue is. Nobody claims that one can do without syntax, it is too easy to give sentences where it comes in handy, and nobody claims that one can rely entirely on syntax to solve one's parsing problems, that we easily understand ungrammatical, or a-grammatical, sentences shows the contrary. ("Parsing" here is simply the process of going from surface structure to "semantic representation". I am not restricting the term to grammatical parsing.) The real substantive issues as I see it are two: "how much" syntax is needed (both for analysis and generation), and how should this information be interfaced with the rest of the system. Further discussion will show, I think, that we have no good answers to either of these questions, and so we should add a third, more pragmatic question: What is the best way to proceed until answers are ±n on the first two?

In asking "how much" syntax is needed one usually takes as a basis for comparison the yet hypothetical set of rules which would allow one to generate all and only the grammatical sentences of English. Restricting ourselves to problems of analysis (since synthesis has played a comparatively unimportant role in AI), we can then ask at least two new questions. Are there any rules in the hypothetical basic set which are <u>never</u> needed to parse English, and secondly, are there rules which are so seldom needed that we should not apply them "unless we have evidence that they are needed" (these last are scare quotes only). As for the first of these, since nobody has offered any candidates for such a rule, the current answer must be "don't know". My personal guess however is that there are few if any such rules. Take, for example, the question of verb subject agreement. As everybody knows, and I have inadvertantly shown all too often (in French), one can be understood in spite of the fact that one's verbs have the wrong ending. Perhaps then various of the verb ending rules are not needed. There are, however, occasional sentences where they are crucial (Hunting dogs is/are forbidden), hence they do not qualify as being completely superfluous. Nevertheless, perhaps, they qualify as rules which are so seldom needed that we should only use them if we know they are necessary.

This is plausible, but there are technical reasons why this is impossible with current parsing systems. Virtually all AI parsers are depth first in that they try to produce one complete parsing, hopefully the correct one, first, and only try another if the first is rejected. In the example given above, by the time the parser gets to the verb It will have made a choice about the structure of the noun phrase. If the parser were to ignore subject - verb agreement, it would reach the end of the sentence with a successful, but possibly wrong, parse. That is to say, it would have no way to "know" that in this example the rule of subject agreement is needed. One could, naturally, have one's parser discover both readings of the noun phrase, and hence have some indication that subject agreement is necessary, but whether in fact this is worth doing is a complicated trade-off question - more time handling noun phrases against less time for verb phrases. My intuition tells me it Is a bad trade-off, but only because subject verb agreement is such a simple question. In fact, nobody really knows the answers to questions like this.

One example may be suggestive, but it is hardly an argument. And the situation with respect to the second question I asked at the outset, "how should syntactic information be interfaced with the rest of the system" is just as bad. So let me move on to the pragmatic question of what we should do next. My personal belief is that one should simply take one of the existing parsers "off the shelf" and use it as a "front end". This view is rather old fashioned, but my reasons are quite simple. Few, if any of us, are really worrying about ungrammatical texts, hence a grammatical parse will not be unduly destructive. Furthermore it can be quite useful in establishing the functional structure of the sentence. (That is, answering questions like, which is the direct object of the verb.) This is not say of course, that syntax is necessarily the only way, but given a free syntactic parser, why not use it. Except from arguments stemming from our ability to understand ungrammatical sentences, the only arguments one sees against syntactic parsers are of the form, "why do a syntactic parse if it Is not necessary". They are, that is, efficiency arguments.

Even if these statements are correct (and I for one have seen no hard figures, or arguments, to back them up) we know so little about language comprehension at the present time that questions of efficiency (aside from problems of exponential growth) are completely beside the point.

This is not to say that syntax is no longer a problem. If it is true that we can syntactically parse a much broader range of sentences than our programs can "understand" it is because of our ignorance of the latter process, and not because we understand the former.

RS: Although I doubt that many people agree with me, syntax has always been a dead issue. Our operating view was always semantics first, syntax later. Later it became Knowledge Representation first, syntax later. Syntax is the last thing that children get right when they learn how to talk. If that strategy works for them it should work for us. Note that I am not saying that syntax doesn't exist, only that more progress on understanding issues will be made by ignoring it than by studying it. A great deal of effort has already been put into syntax. Most of these efforts do little to solve the problem because they beg the issue of what will be done with the syntactic trees they produce. By concentrating on understanding issues, syntax will be reduced to the secondary role It so richly deserves.

YW: No, it's not a dead issue, but a number of people have agreed to put it on ice, and turn to something more interesting for a while.

Although there are now good off-the-peg syntax parsers like Woods', I do not think syntax parsing is <u>settled</u>. Those ATN's are clearly much better than the Harvard Analyzer, say, but still far too fragile with regard to semantic problems.

Conversely, I do not think those who have argued the superiority of semantics-driven parsers have proved their case (T include myself.) I think they will only do so when they produce a semantics-driven parser as robust and portable as Woods' syntax one, and in as perspicuous a formalism (i.e. ATN's or production systems, or something like that). This will require a great deal more detail than has been forthcoming so far about how the semantic structures and rules determine relations as low-level as concord, for example, and where exactly the conventional syntactic generalizations are expressed in the system (if they are <u>not</u>, that too will be highly interesting). I think the whole business is unproven at the moment, and settling it will have interesting consequences for linguistics.

TW: Syntax is far from a dead issue unless we take a narrow (though popular) view that it covers the parsing of simple "grammatical" sentences into trees. Live questions include:

What is the relationship between standard syntactic forms and the "phrasal lexicon"? Many idioms are syntactic structures (e.g. "the X'er the Y'er") which cannot be treated either through normal grammars or as lexical items.

How are syntactic cues used to communicate information about the message structure — focus of attention: what is important, what the speaker considers new information, etc. There is a good deal of work in this area among linguistics, and we are far from having satisfactory answers from the point of view of building successful systems for communicating with human users.

How can syntactic analysis be fit into a multi-knowledge-source framework, to handle real language (with its stops and starts, ungrammatlcalities, etc.). This will demand integrating ideas of syntax and grammar with those of frames and prototypes.

WW: During the past decade or so, many advances have been made in natural language processing. Among these have been the development of formal grammar models that provide efficient systematic frameworks for implementing grammars of a complexity and sophistication matching the most advanced work in linguistics (e.g. Woods, 1970).

The most powerful of these grammars contain augments that allow one to associate conditions with the grammar that refer to general semantic information and world knowledge, thus providing a formal interface between the syntactic knowledge embedded in the grammar itself and world knowledge that may be stored elsewhere and in some other form. This has permitted the beginnings of a systematic investigation of the relationship between syntactic knowledge and general semantic and world knowledge. However, our present state of progress in this area is quite immature — much of it is based on understanding single sentences in isolation, and the role of world knowledge is largely limited to rejecting interpretations that do not satisfy certain semantic selectional restrictions. The ultimate challenge in this area is to be able to choose between alternative readings of a sentence based on sophisticated evaluation of the plausibility of the alternatives in context.

From the other direction, many people have tried to attack the language understanding problem directly from the world knowledge, with minor if any interest in syntax. These systems to date, while they frequently illustrate suggestive approximations to various aspects of human performance, have not begun to develop a formal mechanism capable of handling a general range of language understanding phenomena. In many cases, the devices underlying such approaches, when viewed as formal automata, lack the power necessary for a general treatment of the phenomena they purport to solve. In many other cases, the actual mechanisms underlying their performance are not even sufficiently clear to make such judgments. Much more care in formally defining representational conventions and the algorithms that operate on them is required in the work in this area. Woods (1975) discusses a number of representational issues that are frequently left vague in knowledge representation schemes, and the work of Cercone and Schubert(1975) and Brachman (1977) are beginning to make some progress in this area.

One of the major difficulties in investigating the less syntactic characteristics of natural language understanding is that the fundamental problems of factual inference and plausible reasoning become a critical factor at this level. Since these are active research areas in their own right and are far from understood investigations into knowledge-based language understanding that are to make progress at this point must be making contributions to the study of factual inferencesand plausible reasoning as well. Unfortunately, many people who adopt the so-called knowledge-based approach do so without consideration of much relevant work that has gone on in the area of formal inference, dismissing the relevance of such work on the grounds that human reasoning is not necessarily logical or complete. In doing so, they lose the benefit of a great deal of understanding of fundamental problems and frequently propose inadequate treatments of problems that are well-known and well understood in the formal reasoning camps.

While many of the goals of researchers in formal reasoning differ from those of natural language understanding and artificial intelligence, the kinds of discipline involved in formally specifying the details of one's theory and rigorously assessing its capabilities that is characteristic of the formal inferencing work is one that we should emulate. Successful demonstration of a few chosen examples is not a sufficient benchmark — especially the high-level description of the procedure that accomplishes the demonstration seems intuitively satisfying, but the details of how it works are obscure. In this situation, one is too easily led to believe that something has been accomplished when in fact the underlying mechanism may be inadequate (sometimes either trivial or totally unsystematic and ad hoc).

In summary, while there remain a number of outstanding and troublesome problems of syntax in natural language understanding, especially in the areas of coordinate and subordinate conjunctions, mass terms, nominal compounds, comparatives and superlatives, and discourse structure, it is clear that many of the most intractable ones involve the interaction of syntactic and semantic information and require access to plausible inference capabilities for the evaluation of alternative hypotheses. Thus, the problems of factual inference and plausible reasoning are becoming a central problem in natural language understanding, as they have in a variety of other artificial intelligence areas. They are not the only problems, however, and their role in natural language understanding is intricately intertwined with the use of syntactic and semantic knowledge.

Q4: why are new programming or representation languages necessary for NLP?

RS: I have noticed that when students are stuck on a problem in AI that is too difficult for them they often suggest writing a new programming language. I have little doubt that such programming languages would be of some value in facilitating program writing, so that is not why I don't let them do it.

I don't let them do it because it has always seemed to me that writing a new programming language is a way of avoiding tackling a problem you do not know how to program in the first place. If you do know what the solution to a problem is like, then it should be possible to program it in any language, though of course some languages will make doing it easier than others. But if you don't know how to solve the problem, new programming languages won't help.

Don't misunderstand. I think that there is a strong possibility we might use such a language if strong theoretical biases that we disagree with are not built into it. Nonetheless I cannot help but wonder if the real problems aren't somehow being avoided by working on new languages.

: I think the best reason is standardization, and the advantage that would come from downgrading superficial differences between notations and idiosyncratic systems. This facility seems to me the main advantage offered by KRL, but it will be psychologically very difficult to get many researchers to adopt <u>any</u> such standard language. One reason is the obvious one that such high-level languages are bound to commit users, both as regards control structures and metaphysical questions of representation, in ways they do not want to be committed. If such a language does <u>not</u> commit one (and KRL makes little or no commitment about inference and control structures) then, of course, it will not be a programming language at all, and many of the problems will not arise. I have a hunch, and this may be over-cynical, that advance on this front may not, in the end, be made by those working at it from our end, as it were. It may come from those working on programming languages proper, and their work may later be seen to be relevant to AI (rather than from those tackling the problem of knowledge representation from AI directly). I feel that we should for now go on working on sloppy systems that actually work, and hope that the language constructors (as well as the "semantics of programs" people) will come along later and make what we do seem perspicuous and sensible as well as effective. The converse view (that we need advance with the languages first) seems to me unappealing because I cannot, here and now, imagine what a new-wonder-programming-language that made NLP possible would be like. (But that may well be only my own lack of imagination.)

TW: New programming and representation languages will always be needed for research not only in NLP, but in all of AI. Research follows a cycle:

A researcher has some new intuitions about the structure of a problem or a mechanism, and decides to embed them in a program.

He or she writes that program in some existing language, which was designed to facilitate working within a previously understood set of intuitions.

Some of the new ideas prove to be of general utility.

Someone (possibly the same researcher) writes a new language which facilitates building systems which make use of these new ideas.

The whole thrust of higher-level languages is to look for things which are done commonly and in a standard way, and to make their detail invisible to the user, so that he or she can concentrate on the next higher level of structure. Without this, research would rapidly bog down, since the complexity of any system is strictly limited by the mental capacities of people who program it. To write more complex systems, we need languages which hide those complexities we can afford not to think about. AI went through a clear case of this twenty years ago, when the complexities of list processing and storage allocation were hidden in the primitives of LISP.

I believe (as do many others today) that there is another level which is ripe for the same treatment. This level has new structure along both the declarative and procedural dimensions. Declaratively, it deals with the way in which descriptions (defined by semantic notions of "description", "prototype", "Instance", etc.) are implemented in list structures. The AI languages (e.g. Micro-Planner, Connlver, QA4) took a small step in this direction, but did not try to deal with the semantics of description in a systematic way. Current representation language research (KRL, MDS, OWL, partitioned semantic nets) is much more ambitious in its attempt to provide the user with a higher level of structure. Procedurally, new languages must provide means to reduce the complexities of building systems which multi-processing, resource allocation, and integrated goal-driven and data-driven processing. We have many different starting points in the current AI systems, (including production systems, standard hierarchical control, coroutines, etc.) but they have not been satisfactorily unified in a workable system.

Although no one can predict the details, it seems clear that some language or languages at this higher level will eventually become a widely accepted standard for work in NLP (and AI in general), just as LISP has dominated the more traditional ALGOL/FORTRAN languages for current AI work.