UNDERSTANDING WITHOUT PROOFS

by

Yorick Wilks
Artificial Intelligence Project
Stanford University
Stanford, California

ABSTRACT

The paper describes the analysis part of a running analysis and generation program for natural language. The system is entirely oriented to matching meaningful patterns onto fragmented paragraph length input. Its core Is a choice system based on what I call "semantic density". The system is contrasted with (1) syntax oriented linguistic approaches and (2) theorem proving approaches to the understanding problem. It is argued by means of examples that the present system is not only more workable, but more intuitively acceptable, at least as an understander for the purpose of translation, than deduction-based systems.

1. Introduction

In this paper I describe a working system for understanding natural language. The assumptions underlying it are somewhat different from those of current A.I. natural language systems, and the chief points of difference are these: the present system is not based on a theory of natural language derived from contemporary linguistics, nor is it based on theorem proving [TP] techniques and the essential use of deductive processes. There are excellent reasons for doing automatic theorem proving, but not for doing it and calling the product natural language analysis.

It seems a priori implausible that the operations of the understanding are essentially deductive; the average man finds the exercises at the end of the first chapter of a logic book quite hard; yet he understands adequately most of what he reads and hears. There is, I believe, an unexamined assumption of contemporary A.I. here that strikes at the root of the whole empiricist tradition in science, namely that the principles of logic play an essential role in our description of the world. It is perfectly possible to deny empiricism at that point and still be scientific, but doing so should give cause for more thought and discussion. The two attitudes to this question, of the relation of deduction and ordinary reasoning, were distinguished by Hume as follows:

> "And if [ideas about facts] are apt, without extreme care, to fall into obscruity and confusion, the inference are always much shorter in these disquisitions, and the Intermediate steps..... much fewer than in the [deductive] sciences".

Understanding systems intended to mimic commonsense reasoning should perhaps aim for these "shorter", non-deductive, inferences, even if TP techniques were much more advanced than they now are. So then, for language understanding by machine we should start from an appropriate place, rather than from what we happened to be doing already. Also, one should work with a system of analysis capable of handling language in the form in which it actually comes, not in elementary fact-form sentences of about eight words, very handy for direct translation in PLANNER or the predicate calculus, but in 20-30 word sentences, full of ambiguous words, complex constructions, and metaphorical uses. This lacuna is particularly clear in Charniak's[1] natural understanding system, which actually starts with children's stories as formalized codings.

One argument for starting with an artificially simplified format is that *a* language containing only, say, simple object words is just part of our language and could, in principle, be expanded to the whole. But Wittgenstein carefully constructed an experimental language of words like "block" and demonstrated, to the satisfaction of many, that even though it WAS a language, it was one "quite different from our own".

Nor do the systems constructed by contemporary linguists provide any real starting point for this task. Their syntax-oriented approaches have failed to provide adequate computational basis for the analysis of quite ordinary sentences; and the semantic analyses that come with them, such as the binary marker system of Foder and Katz[2] are quite inadequate for any attempt to make meaning the central Issue. It is not possible, for example, to express the meaning of complex actions like "provoke", "irritate" or "amuse" by means of any binary system of markers.

The sytem I describe is intended to be different. It is a natural language analyzer and generator centered on notions of meaning and context. It accepts input in English and outputs it in French. This process is indeed machine translation [MI], and, as Minsky[3] pointed out, any successful attack on the problem will indeed require understanding. The advantage of KT as a context in which to set an understander is its built-in empiricism; the answer is clearly right or wrong. There is a danger of circularity with task-defined notions of "understanding" of course: the system might be said to understand if and only if it translates adequately.

Yet the notion need not be circular here, for one can suggest desiderata for a meaning centered system. To understand, for MT, a system must understand the conceptual meaning of complex but everyday notions like "fascism" or "knowing": it must understand conceptually what is involved in such complex actions as "forgive" or "interrogate". It must resolve the anaphora of pronouns without recourse to long and implausible deductions. It must also be able to understand the difference *of* meaning, say, between "word" *and* "story" so that when It comes to translate "I told an X" it will use a different verb according to whether X is a word or a story——and this must be done in a general way, not just by lists. It must also resolve word sense ambiguity from context, not only of agents and actions, but ,and above all ,have some theory adequate to disambiguate prepositions. Anyone who doesn't realize how ambiguous they are should consider that "out of" has at least six translations into French, depending on its conceptual role.

Nothing here, of course, is denying the need for knowledge of the physical world, and inferences based upon it, for understanding and translation. What is being argued for here is non-deductive, common sense, inference expressed in formalism that is a natural extension of the meaning representation itself.

A simple case will establish the need for such inference: consider the sentence "The soldiers fired at the women, and I saw several of them fall". Anyone who writes that sentence will be taken to mean that the women fell, so that when, in analyzing the sentence, the question arises of whether "them" refers to "soldiers" or "women" (a choice which will result in a differently gendered pronoun in French) we will have to be able to infer that things fired at often fall, or at least are much more likely to fall than things' doing the firing. Hence there must be access to inferential information here, above and beyond the meanings of the constituent words, from which we could infer that hurt things tend to fall down.

The deductive approaches mentioned claim to tackle just such examples, of course, but later in this paper I will argue for a different approach to them which 1 shall call common sense [CS] inferences rules. I shall also distinguish clearly between CS inference and what I have called the deductive approach.

CS Inference rules are put in "from the bottom": that is to flay they would be typed in at the console in English, in the same form as the one above that hurt things fall. They would be treated as for translation into French, except that they would remain within the system in the interlingual format, but marked as CS inference rules. They could also be put out in French of course; but the importance here is that the CS rules remain in the same form of representation as the material to be translated subsequently.

2. A System of Semantics Based Language Analysis

A fragmented text is to be represented by an interlingual structure consisting of TEMPLATES bound together by PARAPLATES and CS INFERENCES. These three items consist of FORMULAS (and predicates and functions ranging over them and sub-formulas), which in turn consist of ELEMENTS.

ELEMENTS are sixty primitive semantic units used to express the semantic entities, states, qualities and actions about which humans speak and write. The elements fall into five classes, which can be illustrated, by example, as follows. (elements in upper case). (a) entities: MAN (human being), STUFF (substances), THING (physical object), PART (parts of things), FOLK (human groups), ACT (acts), STATE (states of existence), BEAST (animals), etc., (b) actions; FORCE (compels), CAUSE (causes to happen), FLOW (moving as liquids do), PICK (choosing), BE (exists) etc., (c) type indicators: KIND (being a quality), HOW (being a type of action) etc., (d) sorts: CONT (being a container), GOOD (being morally acceptable), THRU (being an aperture), etc., (e) cases: TO (direction), SOUR (source), GOAL (goal or end), LOCA (location), SUBJ (actor or agent), OBJE (patient of action), IN (containment), POSS (possessed by), etc.

FORMULAS are constructed from elements and right and left brackets. They express the senses of English words; one formula to each sense. The formulas are binarily bracketed lists of whatever depth is necessary to express the word sense. They are written and interpreted with, in each pair at whatever level it comes, a dependence of left side on corresponding right. Formulas can be thought of, and written out, as binary trees of semantic primitives. In that form they are not unlike the lexical decomposition trees of Lakoff5, together with a dependency rule for interpreting the trees. The present system of semantic coding is a refinement of one developed in Cambridge, England, in the early Sixties.

Consider Che action "drink" and its relation to the formula:

((*ANI SUBJ)(((FLOW STUFF)OBJE)((*ANI IN)(((THIS (*ANI (THRU PART)))TO)(BE CAUSE)))))

*ANI here is simply the name of a class of elements, those expressing animate entities namely, MAN, BEAST and FOLK (human groups). In order to keep a small usable list of semantic elements, and to avoid arbitrary extensions of the list, many notions are coded by conventional sub-formulas; so, for example, (FLOW STUFF) is used to indicate liquids, and (THRU PART) is used to indicate apertures.

Let us now decompose the formula for "drink". It is to be read as an action, preferably done by animate things (*ANI SUBJ) to liquids ((FLOW STUFF)OBJE), of causing the liquid to be in the animate thing (*ANI IN) and via (TO indicating the direction case) a particular aperture of the animate thing; the mouth of course. It is hard to indicate a notion as specific as "mouth" with such general concepts. But is would be simply irresponsible, I think, to suggest adding MOUTH as a semantic primitive, as do semantic systems that simply add an awkward lexeme as a new "primitive". Lastly, the THIS indicates that the part is a specific part of the subject.

The notion of preference is important here: SUBJ case displays the preferred agents of actions, and OBJE case the preferred objects, or patients. We cannot enter such preferences as stipulations, as many linguistic systems do, such as Fodor and Katz's[2] "selection restrictions", where, if a restriction is not satisfied, then a sentence simply has "no reading". For we can be said to drink gall and wormwood, and cars are said to drink gasoline. It is proper to prefer the normal (quite different from probabilistically expecting it, I shall argue) but it would be absurd, in an intelligent understanding system, not to accept the normal if it is described. Not only everyday metaphor, but the description of the simplest fictions, require it.

A formula expresses the meaning of the word senses to which it is attached. This claim assumes a common sense distinction between explaining the meaning of a word and knowing facts about the thing the word indicates. The formulas are intended only to express the former, and to express what we might find in a reasonable dictionary though in a formal manner. This common-sense distinction cannot be pushed too far, but it will serve provided we have (as we do have) other ways of accessing facts about the world than through formulas.

So, for example, to know the meaning of "water" we need to know it is a liquid substance, among other things. But we do not need to know the fact of physics that it freezes into ice. Many of the world's inhabitants have never seen ice and do not know of its existence even, but they cannot therefore be said to be ignorant of the meaning of whatever the word for water is in their language. And anyone who, at this point, wants to say that those people simply do not know part of the meaning of water should ask himself if he really wants to say that showing them ice is teaching them about MEANINGS. I follow from a denial of the common sense distinction above.

This flexible method of formula encoding and decomposition, down to any degree of depth necessary to express the meaning of a word, is designed in part to avoid a number of pitfalls, well known in other systems of meaning analysis, such as trying to specify in advance all the ways in which an action or agent can be qualified. In a number of A.I. approaches there is often no attempt at lexical decomposition or the establishment of semantic primitives. New words "encountered" are Simply added as primtives in new "axioms". This leads to an endless proliferation of "primtive" vocabulary, as well as inefficiency of representation, and the inability to generalize and connect clearly connected things (such as two facts differing only by a synonym, for example).

Just as elements are to be explained by seeing how they functioned within formulas, so formulas, one level higher, are to be explained by describing how they function within TEMPLATES, the third kind of semantic item in the system. The notion of a template is intended *to* correspond *to* an intuitive one of message: one not reducible merely to unstructured associations of word-senses as some have suggested.

A template consists of a network of formulas grounded on a basic actor-action-object triple of formulas. This basic formula triple is found in frames of formulas, one formula for each fragment word in each frame, by means of a device called a bare template. A bare template is simply a triple of elements which are the heads of three formulas in actor-action-object form.

For example: "Small men sometimes father big sons", when represented by a string of formulas, will give the two sequences of heads (or main,right-most,elements):

KIND MAN HOW MAN KIND MAN

and

KIND MAN HOW CAUSE KIND MAN.

(CAUSE is the head of the verbal sense of "father"; "to father" is analyzed as "to cause to have life".)

The first sequence has no underlying template; however, in the second we find MAN CAUSE MAN which is a legitimate bare template. Thus we have disambiguated "father", at the same time as picking up a sequence of three formulas which is the core of the template for the sentence. It must be emphasized here that the template is the sequence of formulas, and not *to be* confused with the triple of elements (heads) used to locate it.

It is a hypothesis of this work that we can build up a finite but useful inventory of bare templates adequate for the analysis of ordinary language: a list of the messages that people want to convey at some fairly high level of generality (for template matching is not in any sense phrase-matching at the surface level), The bare templates are an attempt to explicate a notion of a non-atomistic linguistic pattern, to be located whole in texts in the way that human beings seem to when they read or listen.

The present working list of bare templates is stored in the program in Backus Normal Form for convenience of reading. The list consists of items like

|< *ANI > < FEEL > < *MAR >|

which says that, for bare templates whose middle, action, element is FEEL, the first, agent, element must be from the class of elements *ANI. Similarly, the object element must come from the element class *MAR, and therefore be one of the mark elements STATE, SIGN or ACT. All of which is to say that only animate things can feel, and that what they *feel (since* the notion of tactile feeling is covered by SENSE, not FEEL) are internal states, and acts, or their written equivalents. I would not wish to defend the particular template list In use at any given moment. Such lists are always subject to modification by experience, as are the formulas and even the inventory of basic elements. The only possible defense is that the system using them actually works (which can only be verified by a visit in the case of a computer program), and If anyone replies that its working depends on mere inductive generalization I can only remind them of Garvin's obvious but invaluable remark that all linguistic generalizations are, and must be, inductive.

Let us now illustrate the central processes of expansion and preference by considering the sentence "The big policeman interrogated the crook", let us take the following formulas for the four main word senses:

(1) "policeman": ((FOLK SOUR)((((NOTGOOD MAN) OBJE) PICK)(SUBJ MAN)))

i.e. a person who selects bad persons out of the body of people (FOLK). The case marker SUBJ is the dependent in the last element pair, indicating that the normal "top first" order for subject-entities in formulas has been violated, and necessarily so if the head is also to be the last element in linear order.

(2) "big": ((*PHYSOB POSS)(MUCH KIND))

i.e. a property preferably possessed by physical objects (substances are not big).

(3) "interrogates": ((MAN SUBJ)((MAN OBJE)(TELL FORCE)))

i.e. forcing to tell something, done preferably by humans, to humans.

(4a) "crook": ((((NOTGOOD ACT)OBJE)DO)((SUBJ MAN))

i.e. a man who does bad acts. And we have to remember here that we are ignoring other senses of "crook" at the moment, such as the shepherd's.

(4b) "crook": ((((((THIS BEAST)OBJE)FORCE)(SUBJ MAN)) POSS)(LINE THING))

i.e. a long straight object possessed by a man who controls a particular kind of animal.

The template matching algorithm will see the sentence under examination as a frame of formulas, one for each of its words, and will look only at the heads of the formulas. Given that MAN FORCE MAN is in the inventory of bare templates, then one scan of a frame of formulas (containing formula *(ka)* for "crook), will have picked up the sequence of formulas labelled above 1, 3, 4*a,* in that order. Again when a frame containing formula *(kb),* the shepherds' sense of "crook", is scanned, since MAN FORCE THING is also a proper bare template, the sequence of formulas 1, 3, *kb* will also be selected as a possible initial structure for the sentence,

We now have two possible template representations for the sentence after the initial match; both a triple of formulas in actor-action-object form. Next, the templates are expanded, if possible. This process consists of extending the simple networks we have so far: both by attaching other formulas into the network, and strengthening the bonds between those already in the template, if possible. Qualifier formulas can be attached where appropriate and so the formula numbered 2 (for "big") is tied to that for "policeman" in both templates. But now comes a crucial difference between the two representations, one which will resolve the sense of "crook".

The expansion algorithm looks into the formulas expressing preferences and sees if any of the preferences are satisfied: as we saw formula 2 for "big" prefers to qualify physical objects. A policeman is such an object and that additional dependency is marked in both templates: similarly for the preference of "interrogate" for human actors, in both representations. The difference comes with preferred objects: only the formula 4*a* for human crooks can satisfy that preference, the formula 4*b* for shepherds' crooks, cannot. Hence the former template network is denser by one dependency, and is preferred over the latter in all subsequent processing: its connectivity is (using numbers for the corresponding formulas, and ignoring the "the"s):

$$2 \rightarrow \rightarrow 1 \rightarrow \rightarrow 3 \leftarrow \leftarrow 4a$$

and so that becomes the template for this sentence. The other possible template (one arrow for each dependency established) was connected as follows:

$$2 \rightarrow \rightarrow 1 \rightarrow \rightarrow 3 \leftarrow 4b$$

and it is now discarded.

Thus the parts of the formulas that express preferences of various sorts not only express the meaning of the corresponding word sense, but can also be interpreted as implicit procedures for the construction of correct templates. This preference for the greatest semantic density works well, and can be seen as an expression of what Joos calls "semantic axiom aumber one"[4], that the right meaning is the least meaning, or what Scriven' has called "the trick (in meaning analysis) of creating redundancies in the input". This uniform principle works over both the areas that are conventionally distinguished in linguistics as Syntax and semantics. There is no Such distinction in this system, since all manipulations are of formulas and templates, and these are all constructed out of elements of a single type.

The limitation of the illustrative examples, so far, has been that they are the usual short example sentences of linguists, whereas what we actually have here is a general system for application to paragraph length texts. I will now sketch in, for two sorts of case, how the system deals with non-sentential fragments with a general agent-action-object template format.

In the actual implementation of the system, an input text, of up to small paragraph length, is initially fragmented, and templaesare matched with each fragment of the text. The input routine partitions paragraphs at the occurrence of any of an extensive list of KEY words. The list contains almost all punctuation marks, subjunctions, conjunctions and prepositions. In difficult cases, described in detail in [9] fragmentations are made even though a key word is not present, as at the stroke in "John knows/Mary loves him", while in other cases a fragmentation is not made in the presence of a key word, such as "that" in "John loves that woman". All that I am calling fragmentation here would be called primitive syntax by many linguists. There is no conflict of views there, for the distinctions that conventional syntax marks must be made within any system. What is hypothesized here is that such discriminations require no special emphasis within a system whose only form of coding is what would normally be called semantic. That is to say the fragmentation routine FRAGM has access only to the semantic formulas for the words of a text.

Let us consider the sentence "John is/in the house", fragmented into two parts at the point marked by the stroke. It should be clear that the three part template, of standard agent-act-action form, cannot be matched onto the fragment "John is". In such case, a degenerate template MAN BE DTHIS is matched onto the two items of this sentence; the last item DTHIS being a dummy object, indicated by the D.

With the second fragment "in the house" a dummy subject DTHIS fills out the form to give a degenerate template DTHIS PBE POINT. The PBE is the same as the head of the formula for "in", since formulas for prepositions are assimilated to those for actions and have the head PDO or PBE. The fact that they originate in a preposition is indicated by the P, so distinguishing them from the straightforward action formulas with heads DO and BE. POINT is the head of the formula for "house", so this bare template triple for the fragment only tells us that "something is at a point in space". At a later stage, after the preliminary assignment of template structures to individual fragments, TIE routines attach the structures for separated fragments back together. In that process the dummies are tied back to their antecedents. So, in "John is in the house", the DTHIS in the MAN BE DTHIS template for the first fragment of the sentence, ties to the whole template for the second fragment, expressing where John is.

It is very important to note that a preference is always between alternatives: if the only structure derivable does NOT satisfy a declared preference, then it is accepted anyway. Only in that way can we deal naturally with metaphor.

So, in examples like "I heard an earthquake/singing/in the shower" (with fragmentation as indicated by slashes), as contrasted with "I heard/an earthquake sing/in the shower", we shall expect, in the first case, to derive the correct representation because of the preference of notions like singing for animate agents. This is done by a simple extension of the density techniques discussed to relations between structures for different fragments (the TIE routines), in this case, by considering alternative connectivities for dummy parts of templates.

Thus, for the fragment/singing/,there will be a template with a dummy subject and a dummy object. The template will be based on the triple of heads DTHIS CAUSE DTHIS, and will contain only one real formula, namely:

"singing":((*ANI SUBJ)((SIGN OBJE)(((MAN SUBJ)SENSE) CAUSE)))))

which is to say, an act by an animate agent of causing a human to experience some sign (i.e. the song).

Now the overall density will be greater when the agent DTHIS, in the template for "singing", is tied to a formula for "I" in a preceding template, than when it is tied to one for "earthquake", since only the former satisfies the preference for an animate agent, and so the correct interpretation of the whole utterance is made.

But, and here we come to the point of this example, in the second sentence, with "sing", no such exercise of preference Is possible, and the system must accept an Interpretation in which the earthquake sings, since only that can be meant.

So far, I have emphasized the procedures of analysis within the individual fragment. After what I have described, the TIE routines are applied to the expanded templates in a wider context: the same techniques of expansion, dependency and preference are applied between full templates for different fragments of a sentence or paragraph. At that stage, (1) case ties are applied (using the same cases as occur within formulas at a lower level); (2) the equivalence of actives and passive forms is noted; (3) dummies are attached to "what they stand for" as I indicated with the "earthquake example"; and, importantly, (4) anaphoric ties are settled.

The TIE routines apply PARAPLATES to the template codings, using the same density techniques one level further up, as it were. Paraplates have the general form:

<list of predicates> <list of generation itmes and functions> <list of template predicates>

An ordered list of paraplates is attached to English key words. Consider the following three schematic paraplates for "in":

((2OBCAS INST GOAL)(PRMARK *DO)IN(into)(FN1 CONT THING) (PRCASE *DIRE))

((PRMARK *DO) IN (into) (FN1 CONT THING) (PRCASE *DIRE))

((2OBHEAD NIL) (PRMARK *DO) IN (make part) (PRCASE LOCA))

*DIRE is a direction case marker (covering two sub-cases: TO, mentioned above, and FROM), 20BCAS and 20BHEAD are simply predicates that look at both the object (third) formulas of the template in hand, and of the preceding templates, i.e. at two objects. 20BHEAD is true iff the two have the same head, and 20BCAS is true iff they contain the same GOAL or INSTRUMENT subformula. The lower case words simply explain which sense of "in" is the one appropriate to the paraplate in which it occurs. When the system is functioning as a translator these generation items will in this case be different French prepositions.

Now consider the sentence "I put the key/in the lock", fragmented at the stroke as shown. Let us consider that two templates have been set up for the second fragment: one for "lock" as a fastener, and one for the raising lock on a canal. Both formulas may be expected to refer to the containment case. We apply the first paraplate and find that it fits only for the template with the correct (fastener) sense of "lock", since only there will 20BCAS be satisfied, i.e. where the formulas for "lock" and "key" both have a subformula under GOAL indicating that their purpose is to close something. The second paraplate will fit with the template for the canal sense of "lock", but the first is a more extensive fit (indicated by the order of the paraplates, since the higher up the paraplate list, the more non-trivial template functions a paraplate contains) and is preferred. This preference has simultaneously selected both the right template for the second fragment and the correct paraplate linking the two templates for further generation tasks.

If we now take the sentence "He put the number/ in the table", with two different templates for the second fragment (corresponding to the list and flat object senses of "table" respectively) we shall find that the intuitively correct template (the list sense) fails both the first paraplate and the second, but fits the third, thus giving us the 'make part of" sense of "in", and the right (list) sense of "table", since formulas for "number" and (list) "table" have the same head SIGN, though the formula for (flat, wooden) "table" does not.

Conversely, in the case of "He put the list/in the table", fitting the correct template with the second paraplate will yield "into" sense of "in" (case DIRECTION) and the physical object sense of "table"; and this will be the preferred reading, since the fit (of the incorrect template) with the third paraplate yields the "make part of a list" reading in this case.

Here we see the fitting of paraplates, and choosing the densest preferential fit, which is always selecting the highest paraplate on the list that fits, thus determining both word sense ambiguity and the case ambiguity of prepositions at once. Paraplate fitting makes use of deeper nested parts (essentially the case relations other than SUBJ and OBJE) of the formulas than does the template matching.

The TIE routines also deal with simple cases of anaphora on a preference basis. In cases such as "I bought the wine,/sate on a rock/ and drank it", it is easy to see that the last word should be tied by TIE to "wine" and not "rock". This matter is settled by density after considering alternative ties for "it", and seeing which yields the denser representation overall. It will be "wine" in this case since "drink" prefers a liquid object.

In more complex cases of anaphora, chat require access to more information than is contained in formulae, templates or paraplates, the system brings down what I referred to earlier as CS inference rules. Cases that require them will be ones like the sentence: "The soldiers fired at the women and I saw several of them fall". Simple semantic density considerations in TIE are inadequate here because both soldiers and women can fall equally easily, yet making the choice correctly is vital for a task like translation because the two alternatives lead to differently gendered pronouns in French, in such cases the PS system applies a CS rule, whose form, using variables and sub-formulas, would be X(((NOTPLEASE (LIFE STATE))OBJE)SENSE) - X(NOTUP MOVE). For rough expository purposes such a rule is probably better expressed as X[hurt]-X[fall], where the words in square parentheses correspond Informally to the subformulas in the rule. The rules are applied to "extractions" from the situations to form chains, and a rule only ultimately applies if it can function in the shortest, most-preferred, chain.

The way the CS inferences work is roughly as follows: they are called in *at* present *only* when TIE is unable to resolve outstanding anaphoras, as in the present example. A process of extraction is then done and it is to these extractions, and the relevant templates, that the CS rules subsequently apply. The extractions are quasi-inferences from the deep case structure of formulas. So for example, if we were extracting from the template for "John drank the water", unp eking the formula for "water" given earlier would extract that some liquid was inside an animate thing (from the containment case), and that it went in through an aperture of the animate thing (from the directional case). Moreover, since the extractions are partially confirmed, as It were, by the information about actor and object in the surrounding template, we can, by simple tying of variables, extract new quasi-templates equivalent to, in *ordinary* language, "the water is in John" etc. These are (when in coded form) the extractions to which the CS rules apply as it endeavors to build up a chain of extractions and inferences. The preferred chain will, unsurprisingly, be the shortest. This part of the system Is described more fully in [11].

So then, in the "women and soldiers" example we extract a coded form by variable tying in the templates equivalent to [women] [hurt], since we can tell from the formula for "fired at" that it is Intended to hurt the object of the action. We are seeking for partial confirmation of the assertion X? [fall], and such a chain is completed by the rule given, though not by a rule equivalent to, say, X[hurt]~X[die] , since there is nothing in the sentence as given to partially confirm that rule in a chain, and cause it to fit here. Since we are in fact dealing with sub-formulas in the statraent of the rules, rather than words, "fitting" means an "adequate match of sub-formulas".

It is conceivable that there would be an, implausible, chain of rules and extractions giving the other result, namely that the soldiers fall: [soldiers][fire] - X[fire]-X[firedat]-X[hurt] etc. But such a chain would be longer than the one already constructed and would not be preferred.

The most important aspect of this procedure is that it gives a rationale for selecting a preferred interpretation, rather than simply rejecting one in favor of another, as other systems do (see discussion below). It can never be right to reject another Interpretation irrevocably in cases of this sort, since It may turn out later to be correct, as if the "women" sentence above had been followed by "And after ten minutes hardly a soldier was left standing". After inputting that sentence ,the relevant preferences in the example might be expected to change. Nonetheless, the present approach is not In any way probabilistic. In the case of someone who utters the "soldiers and women" example sentence, what he is to be taken as meaning is that the women fell. It is of no Importance In that decision if it later turns out that he intended to say that the soldiers fell. What was meant by that sentence is a clear, and not merely a likelihood matter.

It must be emphasized that, in the course of this application, the CS rules are not being interpreted at any point as rules of inference making truth claims about the physical world. It is for that reason that I am not contradicting myself in this paper by describing CS approach while arguing against deductive and TP approaches. The clearest way to mark the difference is to see that there is no inconsistency involved in retaining the rule expressed informally as "X[fallj-X[hurt]" while, at the same time, retaining a description of some situation in which something animate fell but was not hurt in the least. There is a clear difference here from any kind of deductive system which, by definition, could not retain such an inconsistent pair of assertions.

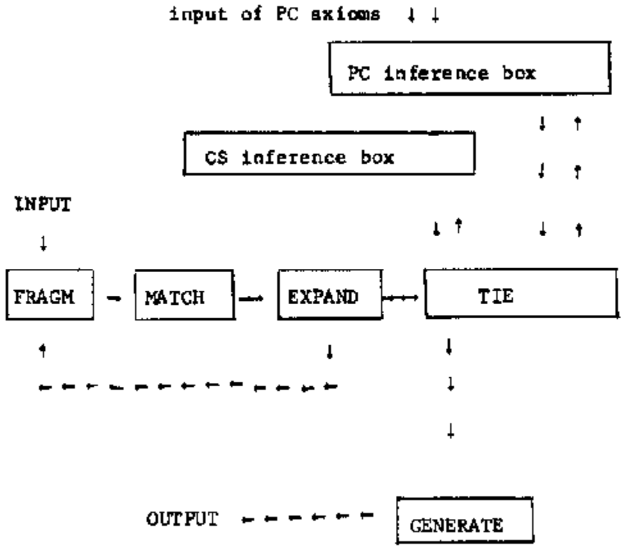
3. Implementation of the System

The system is programmed in LISP 1.6 and MLISP2 and runs on-line at the Stanford Artificial Intelligence Project. It is at present, running over a small vocabulary of about 350 words, but expanding rapidly and already accepting information of up to small paragraph length. Its structural capabilities are already well developed, and by the end of next year or so, we hope to have it translating paragraphs from the AP news wires available every day within the system.

The sections of the analysis program up to and including EXPAND were programmed in LISP 1.6 by the author; those beyond and the GENERATE program were programmed by Annette Herskovits in MLISP2, as was the SAIL program which holds the other programs together.

The emphasis In this paper has been on the concepts in use rather than implementation details, but the generation program is of an independent Interest and is described elsewhere3,9. There is no morphology in the system; every word being a separate LISP atom. This seems justifiable at the present stage, since morphology programs are of no real research interest, but will have to be added as the system grows. The FRAGM routine can call on the results of later and deeper analysis in order to make fragmentations in difficult cases, by considering what matches and subsequent expansions would be possible if certain fragmentations were made and, as usual, preferring the one that would lead to the "semantically densest" overall result. None of this can really be called using the semantics while doing the syntax, since that distinction does not really exist in the system. Everything is done by uniform semantic means.

The general structure of the system *is* indicated by the following diagram:



One shot frames of formulas for fragments are passed to MATCH which sifts them and passes on only the best to EXPAND, where there is no backtracking and the most expanded template is chosen from those available. TIE fits these templates for a text back into a structured representation for the whole by means of the paraplates and common sense inference rules to settle case and anaphora questions. The CS inference rules are brought down and effectively added to the text.

It is not claimed that the present methods will be adequate for tasks like question answering, and the upper box in the diagram envisages an ultimate interface to a deductive system for matters appropriate to it.

4. Discussion

I have argued in this paper for a preference semantics [PS] approach to constructing the core of a language understanding system, and by implication against the thesis that a TP system is necessary for the understanding required for MI. I would also suggest that if it is not necessary then a TP system is not particularly desirable either, unless theorem proving is indubitably what one wants to do. A PS system is more consonant with common sense intuitions, and also avoids the well-known difficulties of searching among the large body of axioms required (unreallstically large for any serious language computation, especially if the axioms contain actual word names as they occur In elementary facts, as they do in the standard approaches), difficulties of proof strategy and so on.

I certainly am not claiming that the inference procedures described in this paper have proved their worth yet. Only that they will be tested with respect to a real and general linguistic base, which seems to me important.

Let me make a final point of comparison with respect to an example of Winograd's[12]. He gives two sentences as follows, though he does not claim to deal with the difficulty they present:

I) I put the heavy book on the table and it broke.

II) I put the butterfly wing on the table and it broke.

There is no problem here (concerning the referent of "it" in each sentence) for a PS analysis if we envisage "broke" as preferring (apparent) agents that are marked FRAGILE or RIGID in their formulas in that order of preference.

In (I) the table will be selected because it is rigid though the book is not, while in (II) the insect-part will be selected because it is FRAGILE while the table is only RIGID.

Note here that it is easy to specify preferences, though I would not know how to begin to specify the appropriate axioms and boundary conditions for a TP approach to the example. There would be too many axioms to search among, in no obviously principled manner, to settle the example.

Note too that the information required for PS here is all available in the appropriate place, in the templates already constructed. And the strategy employed is uniform (i.e. preference for "syntax", semantics, and inference) not ad hoc for each *case in* the *way* that PLANNER specifically encourages. Of course, PLANNER, could be used to program the present approach as well *as any* other, what I am talking about here is the content, the principles programmed, which were entirely different at each stage of Winograd's program (Halliday grammar, Fodor and Katz semantics, first order logic).

It will probably be replied at this point, and rightly so, well that's all very well, but what about the difficult cases where you go wrong? I am sure that all systems will go wrong sometimes, yet I see no reason to think that TP systems will have any better chance of finding they have erred than *a PS* system has, and for four reasons:

1) There is no general test *of* consistency available in any system, and certainly none in the PLANNER type systems. So, even though they have an explicit logic, in which contradiction is of course defined, how could they know they were wrong in any given case, unless the text examined was kind enough to contradict the wrong deduction explicitly and pretty soon after it had been made?

2) Many of the most popular examples in this field have an irresolable vagueness, where one can hardly be said to be "Wong" at all whatever one decides is the antecedent of a particular pronoun. I think Charniak's case is probably one: 'Vhen Penny heard about the costume ball she started thinking about what Mother could wear, Mother had to tell her she had not been invited.". There is simply not enough Information to make a "right" choice about the reference of the "she". Some readers may insist at this point that tne case is not significantly different from *my* "women and soldiers" example where I argued that the intuitively correct answer could not be disputed. Even granted that, my general case is not weakened, for I argued in the earlier case too that the intuitively incorrect answer should not simply disappear from sight, as it were, but be less preferred. What seems to me the important missing piece in the approaches like Charniak's is that one is shown how the "correct" answer *ie* achieved, but never shown why the other answer is NOT achieved. Yet on the sorts of premises usually given for examples, it ought perhaps to be, since it is easy to stress that sentence *so that Penny* becomes the referent of the troublesome pronoun. So even with a deductive analyzer both answers ought perhaps to be "deduced", yet they never are. Nor are we even shown why the desired answer would always be found first. Some analog of preference could perhaps be built into even the deductive approach in terms of relative lengths of proofs.

3) There is an implicit but unjustified assumption in the deductive approach that the utterer will always use correct logic. Should he fail to, things go badly wrong. Consider the following silly children's story: "I have a nice dog and *a* slimy snake. My dog has white furry ears. All animals have ears but my snake has no ears, so it is a mammal too. I call it Horace."

Since the story contains a logical error, any deductive analyzer for solving anaphora problems in children's stories1, must conclude that it is the dog that is called Horace (since only that conclusion is consistent with its information), whereas any reader can see that Horace is a snake. This is only a knock-down argument of course, but it could be amplified from current linguistics,5 and 11 , where a great deal of misguided theoretical effort has been made to establish . the connection between conventional logic and the interpretation of utterances.

4) Cases can be constructed that really do need deductions on facts to resolve such references, and I think Charniak's "top" example is one such. But they are PUZZLES, dear *to* the heart of *all true* A.I. people, and therefore perhaps not examples of natural language understanding at all. No one could possibly deny that there are such puzzles statable in natural language, but ordinary people have difficulty understanding them. They are rarely FOUND in ordinary speech or writing that communicates without causing puzzlement.

It seems to me the onus is on TP people to produce examples, unamenable to PS methods, yet which are not irresolubly vague, nor are they puzzles. I suspect it will not be that easy, since there is ancilliary evidence that people understand in just the sort of conceptual density way I have tried to map.

If I am at all right in this conjecture, then it may be possible that A.I.'s problem-solving and theorem-proving ancestry may have been more a hindrance than a help with the difficult problem of natural language understanding, and *that a* solution may be found by concentrating more on efforts to represent meaning adequately, and to choose, in a principled way, between alternative interpretations.

REFERENCES

1. E. Charniak, in Rustin (ed),"Natural Language Processing", Algorithmics Press, 1975-

2. J. Fodor and J. Rata, "The Structure of a Semantic Theory", Language 1963.

3. A. Herskovits, "On the Generation of French from a Representation", Stanford Artificial Intelligence Laboratory Memo (forthcoming).

4, M. Joos, Semantic Axiom No. 1, Language 1971.

5. G. Lakoff, "Linguistics and Natural Logic", in Davidson and Harman (eds), Semantics of Natural Language. New York, 1972.

6. M. Minsky, in Minsky (ed)(Semantic Information Processing, (MIT 1968).

7. M. Scriven, "The Concept of Comprehension", in Carroll and Freedle (eds), Language Comprehension, Washington, D.C. 1972.

8. Y. Wilks, "Gramnar, Meaning and the Machine Analysis of Language", London, 1972.

9- Y. Wilks, 'the Stanford MI and Understanding Project", in Schank and Colby (eds), Computer Models of Thought and Language, San Francisco, 1973-

10. Y. Wilks, "Lakoff on Linguistics and Natural Logic", Stanford Artificial Intelligence Laboratory Memo No. AIM--161, 1972.

11. Y. Wilks, "Natural Language Inference", Stanford Artificial Intelligence Laboratory Memo, (forthcoming).

1?. T. Winograd, "Understanding Natural Language", in Schank and Colby (eds), Computer Models of Thought and Language, San Francisco, 1975.

This research was supported by the Advanced Research Projects Agency of the Department of Defense under Contract N0.SD-I83.