[From: A.Zampolli, N.Calzolari (eds.) *Computational and mathematical linguistics*. Proceedings of the International Conference on Computational Linguistics, Pisa 1973, vol.I (Firenze: Olschki, 1977)]

YORICK WILKS - ANNETTE HERSKOVTTS

AN INTELLIGENT ANALYSER AND GENERATOR FOR NATURAL LANGUAGE

1. INTRODUCTION

After the unhappy conclusion of most early attempts at machine translation [MT], some justification is required for presenting it again as a reasonable computational task. M. MINSKY (1968), among others, argued that there could be no MT without a system that, in an adequate sense, understood what it was trying to translate. The meaning structures and inference forms that constitute the present system are intended as an understanding system in the required sense, and so as justifying a new attack on an old but important problem.

MT is not only an important practical task; it also has a certain theoretical significance for a model of language understanding. For it provides a clear test of the rightness or wrongness of a proposed system for representing meaning, since the output in a second language can be assessed by people unfamiliar with the internal formalism and methods employed. Few other settings for a theory of language analysis admit of such objective test: dialogue systems are notoriously difficult to assess, and command systems are restricted to worlds to which commands are relevant: such as those of physical objects and picking them up, which exclude the world of real non-imperative discourse about such subjects as friendship, the United Nations, and the problems of juvenile delinquency. On the other hand, conventional systems of linguistics produce only complex representations that can be disputed only on internal grounds. They are never used to produce objective discussible output, like a sentence in another language, that would test the adequacy of the whole representation.

Since the early MT work there has been a considerable development in formal linguistics, and in particular the creation of the school

Acknowledgement: This work has been supported under Advanced Research Projects Agency Contract N. 457.

of transformational grammatical analysis [TG]. This form of analysis of natural language has little relation to the work described here, and for three reasons.

Firstly, TG was set up to be quite independent of all considerations of meaning, context and inference, which constitutes something of a disqualification for the present task, namely understanding language. Consider even such an apparently structural-grammatical matter as the ambiguity of prepositions: *out of*, for example, is highly ambiguous, which can be seen from any reflection on such sentences as:

I live out of town.
I hit her out of anger.
I threw the ball out of the window.
The statue is made out of marble.

An objective measure of the ambiguity is that the occurrences of out of in those sentences would be translated into French in three quite different ways. Yet, even in such a basic structural area, TG makes no suggestions whatever as to how the choice should be made. Whereas, in the preference semantics [PS] system described below, the choice is made in a simple and natural manner. Such defects as this have been to some extent remedied in a recent development of the TG system, generative semantics [GS]. However, for our purposes GS, like TG, suffers from the other two defects below.

Secondly, it is a matter of practical experience, that TG systems have been extremely resistant to computational application. This practical difficulty is in part due to theoretical difficulties concerning the definition and computability of TG systems.

Thirdly, TG and GS systems suffer from one overwhelming defect, from the point of view of understanding natural language. Both have a “derivational paradigm”; which is to say, both envisage a system which constructs a derivation by running from an initial symbol to a language sentence. Such derivations have the function of either accepting a sentence, or of rejecting it because no such derivation can “reach” the sentence from the starting symbol. Thus all sentences are sorted into two groups by such systems: the acceptable and the unacceptable, and by doing this they claim to define the notion of an “acceptable”, “meaningful” or “grammatical” sentence.

One can see how far such a task is from the one of understanding language: for sorting in this way is exactly what human beings do not do when they hear a sentence. They endeavor to interpret it, changing their rules if necessary as they do so. Yet, within the TG & GS derivational paradigm, it makes no sense to talk of changing the rides and trying another set, even though that is just what any "intelligent" understanding system must do. For example, most conventional grammatical systems are armed with some rule equivalent to “only animate things perform tasks of a certain class”, which compels them to reject such perfectly comprehensible utterances as those which speak of the wind as opening doors and cars drinking gas. (It is unimportant here whether any particular system employs such a particular rule. The point here is a general one about behavior in the face of rule failure). Only an “intelligent” system, outside the derivational paradigm, and able to reconsider its own steps, can overcome this defect. The limitations of TG & GS systems, from the point of view of this project, have been discussed in detail in Y. WILKS (1971, 1972).

The proper comparisons for the present work are with systems of analysis originating from within either artificial intelligence [AI] or computational linguistics [CL], none of which owe any strong debt to the TG tradition, and all of which, in differing degrees, make the concept of meaning representation central: such as the work of R. SIMMONS (1971), T. WINOGRAD (1972), R. SCHANK (1972), and E. SANDEWALL (1971).

Some points of difference between these systems and PS may be mentioned briefly:

(i) PS is very much oriented towards processing realistic text sentences of some complexity and up to 20-30 words long. This difference of emphasis, and the sentence fragmentation and large-scale conceptual linkages its implementation requires, distinguish PS from all the approaches mentioned.

(ii) PS copes with the words of a normal vocabulary, and with many senses of them, rather than with single senses of simple object words and actions like Winograd. It is not wholly clear that his methods could, even in principle, be extended in that way.

(iii) PS contains no conventional grammar for analysis or generation: their task is performed by a strong semantics. This contrasts with Winograd, who took over a linguistic grammar and binary marker “package”, and to some extent with Simmons’ use of case grammar.

(iv) PS does not take theorem proving techniques, of whichever major type, to be the core manipulations for an understanding system; but rather sees them as techniques to be brought in where appropriate.

In this respect, it differs most strongly from Sandewall, whose work assumes some form of theorem prover of a resolution type, into which his predicate calculus representations of natural language sentences can be plugged. PS also differs here from Winograd, whose PLANNER-based system is far more oriented to the proving of truths than the PS system described below. Another major difference between PS and these two other systems is that PS inference rules operate on higher level items, semantic concepts and cases, rather than items at the level of text words and facts (or predicates that replace such items one to one). The latter approach leads to an enormous multiplication **of** axioms/inference rules, with all the subsequent difficulty of searching among them.

Nothing here, of course, denies 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 a 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 beings tend to fall down.

The deductive approaches mentioned claim to tackle just such examples, of course, but in this paper we will argue for a different approach to them which we shall call common sense [CS] inference rules.

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 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 Lakoff and McCawley.

Consider the 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 we think that it would be simply irresponsible 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 above notion of "preferring" is important: 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 & Katz's "selection restrictions". For we can be said do drink gall and wormwood, and cars are said to drink gasoline. It is proper to prefer the normal (quite different from probabilistically expecting it, we shall argue) but it would be absurd, in an intelligent understanding system, not to accept the abnormal 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.

Just as elements are to be explained by seeing how they function 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:

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 (structured lists), 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 sentences in the way that human beings appear 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, or acts, or their written equivalents.

We would not wish to defend, item by item, 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 defence is that the system using them actually works, and if anyone replies that its working depends on mere inductive generalization, we 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 left-right 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, one scan the frame of formulas, containing formula 4a for *crook*, will pick up the sequence of formulas labelled above 1 3 4a, in that order. Again, when a frame containing formula 4b, the shepherd's sense of *crook*, is scanned, since MAN FORCE THING is also a proper bare template, the sequence of formulas 1 3 4b 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, 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 4a for human crooks can satisfy that preference, the formula 4b, 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→→l→→3←←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→→1→→3←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 M. JOOS (1972) calls “semantic axiom number one”, that the right meaning is the least meaning; or what M. SCRIVEN (1972) has called “the trick [in meaning analysis] of creating redundancies in the input”. As we shall see 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.

As an example of linguistic syntax, done by preference, let us take the sentence

John gave Mary the book,

onto which the matching routine will have matched two bare templates as follows, since it has no reason so far to prefer one to the other:

John	*gave*	*Mary*	*the*	*book*
MAN	GIVE			THING
MAN	GIVE	MAN		

The expansion routine now seeks for dependencies between formulas, in addition to those between the three formulas constituting the template itself. In the case of the first bare template, a GIVE action can be expanded by any substantive formula to its immediate right which is not already part of the bare template (which is to say that indirect object formulas can depend on the corresponding action formula). Again book is qualified by an article, which fact is not noticed by the second bare template. So then, by expanding the first bare template we have established in the following dependencies at the surface level, where the dependency arrows “→” now correspond to relations established between formulas for the words linked.

$$\begin{tikzcd}
\text{John} \arrow[r] & \text{gave} & \text{book} \arrow[l] \\
& \text{Mary} \arrow[u] & \text{the} \arrow[u]
\end{tikzcd}$$

But if we try to expand the second bare template by the same method, we find we cannot, because the formula for *Mary* cannot be made dependent on the one for give, since in that template Mary has already been seen, wrongly of course, as a direct object of giving, hence it cannot be an indirect object as well. So then, the template MAN GIVE MAN cannot be expanded to yield any dependency arcs connecting formulas to the template; whereas the template MAN GIVE THING yields two dependency arcs on expansion, and so gives the preferred representation. This general method can yield virtually all the results of a conventional grammar, while using only relations between semantic elements.

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. We will now sketch in, for two sorts of case, how the system deals with non-sentential text fragments with a general template format.

In the actual implementation of the system, an input text is initially fragmented, and templates are matched with each fragment of the text. The input program partitions paragraphs at the occurrence of any of an extensive list of *KEY* words. The list contains all punctuation marks, subjunctions, conjunctions and prepositions. In difficult cases, described in detail in Y. WILKS (1973), 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*.

Let us consider the sentence

John is / in the country

fragmented as shown. It should be clear that the standard agent-act-object form of template cannot be matched onto the fragment John is. In such a 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 country, 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 straightforward action formulas with heads DO and BE. POINT (indicates a spatial location that is not a movable physical object) is the head of the formula for country, 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 country, the DTHIS in the MAN BE DTHIS template for the first fragment of the sentence, is tied to the whole template for the second fragment, expressing where John is.

It is very important to note that a preference is between alternatives. If the only structure derivable does not satisfy a declared preference, then it is accepted anyway. Only thus can we deal naturally with metaphor.

So, in examples like

I heard an earthquake / singing / in the shower

(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 to relations between structures for different fragments by considering, in this case, alternative connectivities for dummy parts of templates.

Thus, there will be a dummy subject and object template for /*singing*/, namely DTHIS CAUSE DTHIS, based on the formula:

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.

In order to give a rough outline of the system, our description has centered upon the stages of analysis within the individual fragment. After what has been described so far, TIE routines are applied to the expanded templates in a context of templates for other fragments of the same sentence or paragraph. The same techniques of dependency and preference are applied between full templates for different fragments of a sentence or paragraph. At that stage, 1) case ties are established between templates (using the same cases as occur within formulas at a lower level); 2) dummies are attached to what they “stand for” as we indicated with the earthquake example; 3) remaining ambiguities are resolved and 4) anaphoric ties are settled.

The first of these tasks is done by applying *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> <case> <stereotype>

A stereotype is a context sensitive generation pattern which will be described in the next section. The paraplates are attached, as ordered lists, to key words in English. Consider the following three schematic paraplates for in: written out in order as follows, without generation stereotypes for ease of explanation, but with a description in lower case of which sense of in is in question in each case.

((2OBCAS INST GOAL) (PRMARK (MOVE CAUSE)) (PROBJE CONT THING) TO (into))
((PRMARK (MOVE CAUSE)) (PROBJE CONT THING) TO (into))
((2OBHEAD) (PRMARK *DO) LOCA (make part))

TO and LOCA are case markers, 2OBCAS and 2OBHEAD are simply predicates that look at both the object (third) formulas of the template in hand, and of the preceding template, i.e. at two objects. 2OBHEAD is true iff the two have the same head, and 2OBCAS is true iff they contain the same GOAL or INSTRUMENT subformula. PRMARK is a predicate on the semantic form of the mark, or word governing the fragment that the key begins. In all the following examples the mark is the action in the first fragment, and the predicate is satisfied iff it is a (MOVE CAUSE) action: an action that causes something to move. Similarly, PROBJE is a predicate on the semantic form of the object (third formula) of the current template, and is satisfied if the predicate's argument is found in the formula.

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, and so to satisfy (PROBJE CONT). 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 2OBCAS 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 simple preference basis. In cases such as

I bought the wine, / sat 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, that require access to more information than is contained in formulas, templates or paraplates, the system brings down what we referred to earlier as CS inference rules. Examples that require them will be ones like:

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 subformulas, 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*, unpicking the formula for *water* given earlier would extract that some liquid was now 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.

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 subformulas in the statement of the rules, rather than words, “fitting” means an “adequate match of subformulas”.

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[*fired at*] → X[*hurt*] etc. But such a chain would be longer than the one already constructed and would not be preferred.

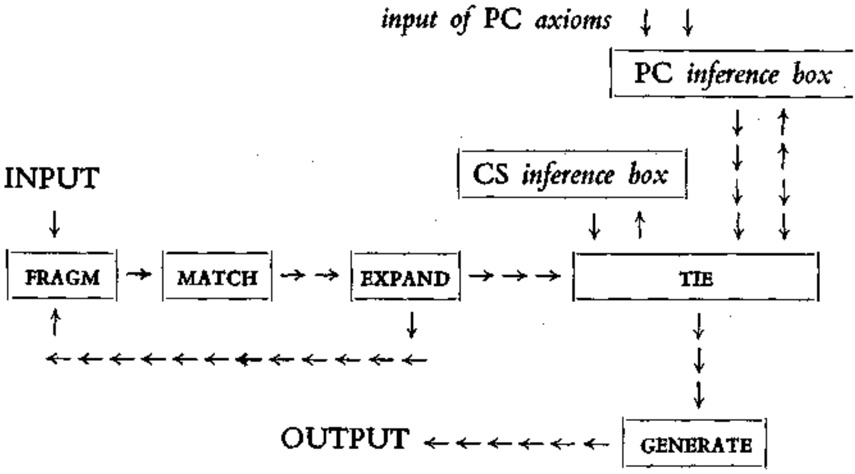
The most important aspect of this procedure is that it gives a rationale for selecting a preferred interpretation, rather than simply rejecting one in favor of another, as other systems do. It can never be right to reject another interpretation irrevocably in cases of this sort, since it may turn out 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 we are not contradicting ourselves in this paper by describing the CS approach while arguing against deductive and theorem proving 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[*fall*] → 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 MLISP 2 and runs on-line at the Stanford Artificial Intelligence Project. It is at present running over a small vocabulary of about 500 words, but expanding rapidly and already accepting information of up to small paragraph length.

The general structure of the system is indicated by the diagram below:



The sections of the analysis program up to and including EXPAND were programmed in LISP 1. 6; those beyond and the GENERATE program were programmed in MLISP 2.

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 later 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, though this cannot be called using the semantics while doing the syntax, since that distinction does not really exist in the system.

Frames of formulas for English 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 very few, and 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. THE GENERATION SYSTEM FOR FRENCH

Translating into French requires the addition to the system of generation patterns called *STEREOTYPES*. Those patterns are attached to English words in the dictionary, both to keys and content words, and carried into the interlingual representation [IR] by the analysis.

A content word has a list of stereotypes attached to each of its formulas. When a word-sense is selected during analysis, this list is carried along with the formula inside the interlingual representation. Thus, for translation purposes, the IR is not constructed simply with formulas but with *SENSE-PAIRS*. A sense-pair is:

<formula for a content word> <list of stereotypes>

As for key words, we have seen in the last section that each key paraplate contains a stereotype, which gets built into the IR for a fragment if the corresponding paraplate has been selected by the TIE routines. This stereotype is the generation rule to be used for the current fragment, and possibly for some of the fragments that follow it.

The simplest form of a stereotype is a French word or phrase standing for the translation of an English word in context, plus a gender marker for nouns. For example:

private (a soldier)	:	(MASC *simple soldat*)
odd (for a number)	:	(*impair*)
build	:	(*construire*)
brandy	:	(FEMI *eau de vie*)

Note that, after processing by the analysis routines, all words are already disambiguated. Several stereotypes attached to a formula do not correspond to different senses of the source word, but to the different French constructions it can yield.

Complex stereotypes are strings of French words and functions. The functions are of the interlingual context of the sense-pair and evaluate to either a string of French words, a blank, or (for content words only) to NIL. i.e. such stereotypes are context-sensitive rules, which check upon and generate from the sense-pair and its context, possibly including fragments other than the current one. When a function in a content word stereotype evaluates to NIL, then the whole stereotype fails and the next one in the list is tried. For example, here are the two stereotypes adjoined to the ordinary sense of advise:

(*conseiller* (PREOB a MAN))
(*conseiller*)

The first stereotype would be for translating *I advised my children to leave*. The analysis routines would have matched the bare template MAN TELL MAN on the words *I-advised-children*. The function PREOB looks whether the object formula of the template, i.e. the one for *children* in our example, refers to a human being; if it does, the stereotype generates a prepositional group with the French preposition a, using the object sense-pair and its qualifier list. Here this process yields *à mes enfants*, and the value of the whole stereotype is *conseiller à mes enfants*.

For the sentence *I advise patience* however, whose translation might be *je conseille la patience,* this stereotype would fail, as the object head in the template, brought in by the concept of patience, is STATE. The second is simply (*conseiller*}, because no prescription on how to translate the object needs to be attached to *conseiller* when the semantic object goes into a French direct object. This is done automatically by the higher level function which constructs French clauses.

Thus we see that content words have complex stereotypes prescribing the translation of their context, when they govern an "irregular" construction, that is irregular by comparison to a set of rules matching the French syntax onto the IR.

The general form of the generation program is a recursive evaluation of the functions contained in stereotypes. Thus, depending on its context of occurrence, a particular word of the French output sentence may have its origin in stereotypes of different levels: content word stereotype, key word stereotype or stereotypes that are part of a set of top level basic functions.

The stereotype for a content word can prescribe the translation of fragments other than the one in which it occurs. Before describing such stereotypes, we need to introduce a new item called a "link". A link has been attached to each fragment by the TIE routines, namely a list of the following three items of information: the key, mark and case.

The mark is a list of one or several words outside the current fragment, each of which relates to the current fragment through the same dependency. The catalogue of dependencies considered includes linguistic relationships such as:

subject on predicate
governor on prepositional phrase
verb on object
verb of main clause on dependent clause
etc.

The case and mark have been selected by the application of paraplates during TIE, in the way we described above.

Here is an example of an English sentence, fragmented, with the matching bare template under each fragment (filled out with dummies where necessary) and with the key, mark, case and stereotype attached by TIE:

fragment / */ bare template*	*key*	*mark*	*case*	*stereotype*
The delegate urged the women MAN TELL MAN	NIL	NIL	NIL	((INDCL))
who were striking MAN NOTDO DTHIS	who	(workers)	SPEC	((WHCL))
to be patient DTHIS BE KIND	to	(urged)	GOAL	(de (INFVP))

In general a generation rule for a fragment comes from a key paraplate. A list of key paraplates reflects the fact that rules of syntax are normally based on some semantic classification; i.e. for given semantic categories and relationships in the context of the key, the output syntax is represented by the adjoined stereotype. However, in a natural language there will be exceptions to any classification scheme. Exceptions are dealt with here by entering the replacement generation rule via the word governing the construction (in general the mark of the fragment).

For example, the paraplates for to as in

John told him / to leave,

state that if the mark is an act of verbal communication (formula headed TELL), then the to phrase should be translated by de followed by an infinitive: *John lui a dit de partir*. This is generally the case; however *to urge* when going into *exhorter*, has also been coded with a TELL head, yet gives the different construction *à partir*. Thus one of the stereotypes for this sense of *urge* is:

(exhorter (DIROB MAN) (FIND-LINK GOAL IR.-VP) a (INFVP))

which would apply in the above example: *the delegate urged* etc., and to *John urged him/to leave* to yield *Jean l'a exhorté à partir*. The point here is that the construction replacing and following to is in general found from the stereotype for to, yet in the present example of urge to, the construction is found from the stereotype for urge which takes precedence over the key stereotype for generation purposes as follows:

DIROB is a function that constructs a direct object with the template object if it is a human being.

FIND-LINK takes as arguments a case, and a descriptor of template types, here IR-VP, which indicates the set of templates with a dummy subject. It searches the IR down from where *urged* occurs, for a fragment with case and template type specified by the arguments, and with this occurrence of urged itself as a mark. The third fragment in our example fulfils these conditions. The control function supervising the evaluation of the stereotype starts then generating from it, using the part of the stereotype which follows FIND-LINK, i.e. “à (INFVP)”, instead of the stereotype of *to* which had been selected during TIE (namely “de (INFVP)”).

INFVP generates an infinitive verb-phrase, after inferring its implicit subject (here *women*) from the semantics, as follows: acts of verbal communication involving an attempt to influence the interlocutor, such as: persuade, order, advise, ... contain a rightmost subformula (FORCE TELL) and the subject of the dependent to phrase is their object. The knowledge of the implicit subject is necessary to proper agreement in French. Thus the translation of the phrase here is: *à être patientes* where *patientes* agrees with *les femmes*.

Key stereotypes are best described by looking at the functions most frequently found in them.

(PREOB <French preposition>) This function will generate a prepositional group, using for the object the stereotypes attached to the object formula of the template. It calls the basic function *NOUN-GROUP*, which uses a sense-pair and a list of qualifying sense-pairs to generate a French nominal group.

(INDCL) Generates a French clause in the indicative mood, from an agent-action-object triple in the IR. Given the earlier process of fragmenting on key-words, these three elements of a “semantic” clause are sometimes in different fragments and then the mark and case make their relationships explicit (the cases used are PRED (predicate) and OBJE (object)). INDCL calls the basic function CLAUSE-GROUP.

To describe the operation of CLAUSE-GROUP and NOUN-GROUP, it is necessary to introduce the two functions which handle stereotypes.

$MAP takes a stereotype as argument. It goes down the stereotype string, building a French string in the process, by concatenating the French words and the result of evaluating the functions in the stereotype. It stops and returns NIL whenever a function returns NIL; otherwise it returns the French string constructed.

$SELECT takes as argument a list of stereotypes and applies $MAP to each of its members in turn, until $MAP returns a non-NIL value.

The bodies of the two main syntactical functions CLAUSE-GROUP and NOUN-GROUP consist of the application of $SELECT to a list of stereotypes which reads somewhat like the phrase structure rules of the corresponding French syntactical constituent. The bottom level functions recursively call $SELECT to work on the list of stereotypes of a given content word and operate upon its output for proper concord, agreement, etc.... To that effect, special variables carry along information of gender, number, person etc....

In fact each function in a stereotype calls $SELECT to work upon a list of stereotypes so that the sequence of $SELECT calls during execution follows the underlying tree structure of the constituent. French words found in stereotypes correspond to the terminal nodes. Generation proceeds from left to right. Concatenation to the right is done by $MAP.

However, some complexity arises from the fragmented structure of the IR and with the problem of integrating complex (i.e. context-sensitive) stereotypes.

As for fragmentation, the program maintains a cursor which points to the fragment which is being generated from; the purpose of certain functions in stereotype (such as FIND-LINK above) is to move the cursor up and down the IR.

Integration of complex stereotypes in some contexts requires the reordering of the stereotype string. Thus, for *I often advised him to leave*, going into *Je lui ai souvent conseillé de partir*, the stereotype:

(*conseiller* (PREOB a MAN))

needs to be rearranged. This is done by a feature in $MAP which permits the values of designated functions in a stereotype to be lifted and stored in registers. The values of these registers can be used at a higher level of recursive evaluation to construct a new correct French string. Finally, the integration of complex stereotypes requires the implementation of a system of priorities, for regulating the choice of generation rules. Since any word or key can dictate the output syntax for a given piece of IR, there may arise conflicts, which are resolved by having carefully settled priorities. The general principle, as in the analysis program, is that a more specific rule has priority over a more general one. Thus, when a content word stereotype prescribes the translation of fragments other than its immediate context, it has priority over any key stereotype. In the above example, generation will proceed from the stereotype of urge and ignore the stereotype (de (INFVP)) attached to the third fragment by the TIE routines.

CLAUSE-GROUP has a general rule for the object of an action, namely, concatenate the value of NOUN-GROUP applied to it. However this is overruled whenever the action stereotype dictates a different handling of the object.

A function REPHRASE allows us complex rephrasings, such as the following: *John nearly killed himself*, which translates properly into *John a failli se tuer*, i.e. the adverb *nearly* goes into the verb *faillir*. *Nearly* has the following stereotype:

```
((REPHRASE VERB-GROUP ((VERB-GROUP FAILLIR) (INFVGO)))
```

The function REPHRASE indicates that the execution of the function VERB-GROUP (a constituent in CLAUSE-GROUP) should be replaced by the evaluation of the stereotype which is its second argument. This will generate a verb-group constructed from *faillir*, followed by an infinitive verb-group with the “current” subject (that of *faillir*) as its own subject. Any stereotype from a REPHRASE call takes precedence over whatever stereotypes the substituted function contained.

Implementation of these priorities requires some functions in the stereotypes to test other stereotypes in advance in order to decide what to generate next. And the overall control function does some bookkeeping; i.e. it keeps track of which sense-pair and fragments have already been generated from, and which stereotype was used.

The overall control function sets the cursor to the first fragment and picks up its stereotype; $MAP is run through it, and the cursor moves up or down the IR as the recursive structure calls for. When $MAP pops up, after exhaustion of the first stereotype, the French phrase that is its value is concatenated to the text already generated. The program then moves down into the IR until it finds a fragment which has not been translated yet; the process is then reiterated as with the first fragment.

Below is an example of on-line input and output from the system:

a pair of English sentences, followed by their translation into good French, and by the semantic block, or interlingual representation, created by the analysis and from which the generation was done. It is the last which is of interest, of course, because the French output, though correct, remains intellectually inscrutable.

The format of the block is a list, each item of which, at the top level, is a text fragment tied to a template, the latter being a list of pairs (of formulas and generation stereotypes), and of sublists of such pairs that are dependents on the main nodes of the template in the manner described above. In the lists at the same level as the text fragments are the key generation stereotypes for fragments, as well as paraplate and inference nodes that declare satisfactory preferred ties.

For example at the line of the block flagged →→→→ $1 on its left hand side, is the node that shows that it has been tied to the correct wine and not the closer but incorrect *table*, which results in *le* rather than *la* in the final output. Also at the line of the block flagged →→→→ $1 is the *dans* in the stereotype for drink that will give the correct translation of *outof* in this context, by taking precedence over all stereotypes directly attached to *outof* such as the one containing *hors de* flagged at line →→→→ $2. At the lines flagged →→→→ $3 and →→→→ $4 can be seen quite different stereotypes for *outof*, both containing *par*, which constitute quite different representations for *outof* for its last two occurrences in the English sentence, both exemplifying the SOURCE case.

```
I PUT THE WINE ON THE TABLE AND JOHN DRINKS IT OUT OF A GLASS.
HE OFTEN DRINKS OUT OF DESPAIR AND THROWS THE GLASSES OUT
OF THE WINDOW.
```

```
JE METS LE VIN SUR LA TABLE ET JEAN LE BOIT DANS UN VERRE. IL BOIT
SOUVENT PAR DESESPOIR ET JETTE LES VERRES PAR LA FENETRE.
```

```
(((I PUT THE WINE) ((NIL NIL NIL ((INDCL)))) 1 (((THIS MAN) I ((PRON
1 MASC SING))) ((PRES ((*ANI SUBJ) ((*PHYSOB OBJE) (((WHERE POINT)
AT) (BE CAUSE))))) PUT ((*IRREG_VB METTRE MIS METTANT ((METS
METTONS METTENT) METTAIS METTRAI) (METTE))) (METTRE)) ((THE
(((MAN OBJE) PLEASE) (FLOW STUFF))) WINE (MASC VIN)) NIL NIL NIL))
((ON THE TABLE) ((ON (PUT) LOCA ((&PREOB SUR)))) 6 (((THIS DTHIS)
DUMTHING) ((THIS PBE) ON NIL) ((THE (NOTFLOW THING)) TABLE(FEMI
TABLE)) NIL NIL NIL)) ((AND JOHN DRINKS IT) ((AND NIL NIL (ET)) (NIL
NIL NIL ((INDCL)))) 1 (((MAL (INDIV MAN)) JOHN ((*PROP_N MASC JEAN)))
((PRES ((*ANI SUBJ) (((FLOW STUFF) OBJE) ((*ANI IN) (((WRAP THING)
OUTOF) (MOVE CAUSE)))))) DRINKS ((*IRREG_VB BOIRE BU BUVANT
((BOIS BUVONS BOIVENT) BUVAIS BOIRAI) (BOIVE))) (BOIRE (&DIROB)
(&LINK DIRE PR X)  (PREOB →→→→ $1 DANS))  (BOIRE))  ((THE (((MAN
OBJE) PLEASE) (FLOW STUFF))) (IT WINE) ((IT_PRON))) NIL NIL NIL))
((OUTOF A GLASS.) ((OUTOF (DRINKS) →→→→* $2 DIRE ((&PREOB (HORS
DE))))) 6 (((THIS DTHIS) DUMTHING) ((THIS PDO) OUTOF NIL) ((A ((((FLOW
STUFF) OBJE) WRAP) THING)) GLASS (MASC VERRE)) NIL NIL NIL)) ((THE
OFTEN DRINKS) ((NIL NIL NIL ((INDCL)))) 2 (((MAL (THIS MAN)) HE ((PRON
3 MASC SING))) ((PRES ((*ANI SUBJ) (((FLOW STUFF) OBJE) ((*ANI IN)
(((WRAP THING) OUTOF) (MOVE CAUSE)))))) DRINKS ((*IRREG_VB BOIRE
BU BUVANT ((BOIS BUVONS BOIVENT) BUVAIS BOIRAI) (BOIVE))) (BOIRE
(&DIROB) (&LINK DIRE PR X) (PREOB DANS)) (BOIRE)) ((THIS DTHIS)
DUMTHING) NIL (((((MANY (WHEN POINT)) HAPN) HOW) OFTEN ((*LEFT
SOUVENT)))) NIL)) ((OUTOF DESPAIR) ((OUTOF →→→→ $3 (DRINKS)
SOUR ((&PREOB PAR)))) 6 (((THIS DTHIS) DUMTHING) ((THIS PDO) OUTOF
NIL) (((MAN NOTPLEASE) (FEEL STATE)) DESPAIR (MASC DESESPOIR))
NIL NIL NIL)) ((AND THROWS THE GLASSES) ((AND NIL NIL (ET)) (NIL
(HE) PRED)) 4 (((THIS DTHIS) DUMTHING) ((PRES ((MAN SUBJ) ((THING
OBJE) (MOVE CAUSE)))) THROWS ((*IRREG_VB JETER JETE JETANT((JETTE
JETONS JETTENT) JETAIS JETTERAI) (JETTE))) (JETER)) ((THE (MUCH
((((FLOW STUFF) OBJE) WRAP) THING))) GLASSES (MASC VERRE)) NIL
→→→→ $4 NIL NIL)) ((OUTOF THE WINDOW.) ((OUTOF (THROWS) DIRE
((&PREOB PAR)))) 6 (((THIS DTHIS) DUMTHING) ((THIS PDO) OUTOF
NIL) ((THE ((THRU PART) THING)) WINDOW (FEMI FENETRE)) NIL NIL
NIL)))
```

The block, or IR, is clearly not wholly target-language independent because it contains the generative rules, however, it is very largely so. Moreover, the semantic representation it expresses could easily be adapted as a data base for some quite different, such as question answering. Indeed, many of the inferences required to set up the IR, like those described in detail above, are equivalent to quite sophisticated question-answering.

5. DISCUSSION

We 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 theorem proving [TP] system is necessary for the understanding required for MT. We 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 (unrealistically large for any serious language computation, if the axioms contain actual word names), difficulties of proof strategy and so on.

This is a difficult and complex question, and not appropriate for detailed discussion here. However, we would point out one unjustified assumption of the TP approach to language analysis, not often brought out, which is that the speaker or writer 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 an animal too. I call it Horace.*

Since the story contains a logical error, any deductive analyser for solving anaphora problems in children's stories, 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.

However, as we said earlier, the sufficiency of the PS system for difficult cases involving reasoning has yet to be tested adequately. One comparison we have not drawn in this paper has been with recent developments in MT by commercial companies (*Natural Language Transl.* etc., 1973). This has been deliberate because, in spite of their descriptions in the press handouts, they almost always turn out to be not translations from natural language, but from some artificial language, like the military's PIMA.

One might say that there always was one other possible way out from MT difficulties, besides the construction of pure linguistic theories or the development of intelligent systems, and that was to restrict the material translated to trivial material, in which case all the well-known problems of natural language analysis disappear. We feel it is this last course that the recently publicized work has chosen.

REFERENCES

A. HERSKOVITS, *The generation of French from a semantic representation*, Stanford Artificial Intelligence Project Memo, forthcoming.

M. JOOS, *Semantic Axiom # 1*, in « Language », I (1972).

J. MCCARTHY, (with P. HAYES), *Some Philosophical Questions from the Standpoint of Artificial Intelligence*, in MELZER, MICHIE (eds.), *Machine Intelligence* 4, Edinburgh, 1969.

M. MINSKY, in M. MINSKY (ed.), *Semantic Information Processing*, Cambridge (Mass.), 1968.

E. SANDEWALL, *Representing natural lan guage information in predicate calculus*, in MELZER, MICHIE (eds.), *Machine Intelligence* 6, Edinburgh, 1971.

R. SCHANK, *Conceptual Dependency*, in « Cognitive Psychology », 1972.

M. SCRTVEN, *The concept of comprehension*, in CARROLL, FREEDLE (eds.), *Language Comprehension*, Washington DC, 1972.

R. SMMONS, (with B. C. BRUCE), *Some relations between predicate calculus and semantic net representations of discourse*, in *Proc. Second Internat. Joint Conf. on* Artificial Intelligence, London, 1971.

Y. WILKS, *Decidability and Natural Language*, in « Mind », 1971.

Y. WILKS, *Grammar, Meaning and the Machine Analysis of Language, London,* 1972.

Y. WILKS, *An Artificial Intelligence Ap*proach to Machine Translation, in SCHANK, COLBY (eds.), *Computer Models of Thought and Language*, San Francisco, 1973.

T. WINOGRAD, *Understanding Natural Language*, Edinburgh, 1972,

Natural Language Translation by Computer Emerges from Government Research Stage, report in « Communications of the ACM», March 1973.