EXPLOITING A LARGE DATA BASE BY LONGMAN

A. MICHIELS (English Dept), J. MULLENDERS (Computer Centre), J. NOËL (English Dept)

University of Liège, Belgium

INTRODUCTION/ABSTRACT.

We wish to explore some of the aspects of the exploitation of two dictionary files by LONGMAN Ltd, one for 'core' English and one for English idioms.

We'll try to show the feasibility of an approach to language processing based on a lexicon, conceived of as the repository of grammatical, semantic and knowledge-of-the-world information.

After giving a brief description of the computer files (Section I) we'll focus on the following points :

a) a lexical approach to grammar allows a considerable simplification of the PSG component of a parsing system (Section II, Part One);

b) the syntactic potential of many lexemes (at surface structure level) can serve as a guide to their deep structure configurations (Section II, Part Two);

c) provided that a dictionary makes use of a limited defining vocabulary, the texts of the dictionary definitions can be processed on the basis of correlations between syntactic structures (filled with individual lexemes or lexemes belonging to specifiable classes) and semantic relationships such as that between a process verb and an instrument (Section III).

SECTION I. DESCRIPTION OF THE COMPUTER FILES

A contract with LONGMAN Ltd has made it possible for us to have access to the computer files of two dictionaries, LDOCE (LONGMAN DICTIONARY OF CONTEMPORARY ENGLISH) and LDOEI (LONGMAN DICTIONARY OF ENGLISH IDIOMS). We have had the LDOCE file for some time but have only just received the LDOEI one.

LDOCE The features of LDOCE which make it specially useful for language processing are the following :

a) it reflects the surface structure environment of its entries by means of a sophisticated system of grammatical codes, most of which can be thought of as strict subcategorization features. For instance, LDOCE specifies

1.- that nouns like FACT or CLAIM can be followed by a THAT-clause,

2.- that a verb such as WATCH can occur followed by an NP followed by an ING-form (we watched the soldiers bleeding).

Though it is mainly concerned with SURFACE structure, LDOCE nevertheless distinguishes between an NP pair following GIVE (He gave his brother (NP_1) a new bicycle (N_2)) [D_1] code and one following CONSIDER (He considered his brother (NP_1) a fool (NP_2)) [X_1] code;

b) through a system of semantic codes of the Katz-and-Fodor type (these codes do not appear in the printed version of the dictionary), LDOCE places semantic restrictions on the subjects and objects of verbs (or on the type of noun that an adjective can modify), specifying for instance that PERSUADE requires a [+ HUMAN] object, and EXTEMPORIZE a [+ HUMAN] subject.

c) LDOCE makes use of a defining vocabulary of some 2,000 items - all the definitions and all the examples associated with the 60,000 entries are couched in that restricted vocabulary.

Concerning points a and b it should be emphasized that the grammatical and semantic codes can appear at two different levels :

1.- ENTRY level : the code is appropriate to all the definitions of the entry in question,

2.- DEFINITION level : the code is not appropriate to the whole entry (i.e. in all its senses) but only to those readings that correspond to the definitions that the code is tagged to.

For instance, READ cannot be assigned the same grammatical and semantic codes in sentences 1 and 2 :

1.- He manages to read at least one book every day

2.- Your paper doesn't read too well.

This second level makes it possible to avoid a proliferation of indiscriminate disjunctions in the specification of the codes to be associated with a given lexeme. It seems to us that by restricting the occurrence of code specifications at only one level (namely, the

ENTRY level), one reduces the predictive power of both grammatical and semantic codes to practically nil in the case of complex entries. On the other hand, the codes that are appropriate at DEFINITION level provide an interesting type of correlation between strict subcategorization and selection rules on the one hand and choice of appropriate reading on the other : such a type of correlation is bound to prove very useful for machine translation purposes.

LDOEI Owing to the use of the same defining vocabulary, LDOEI is a natural extension of LDOCE. Whereas the latter merely lists the idiomatic phrases under the relevant headwords, LDOEI gives the information necessary for recognizing and generating all the syntactic and morphological variants of each idiom. To give only one example, in the entry "TELL° I WHERE TO GET OFF [V : Pass 2]" the sign ° indicates that TELL admits of morphological variation in this phrase, I specifies the place of the indirect object (which does not belong to the idiomatic phrase as such) and the grammatical note [V : Pass 2] informs the user that the syntactic value of the whole phrase is verbal (i. e. that it functions as a VP) and that the passive is to be formed by selecting the indirect object as subject ("He was told where to get off").

LOLEX (LONGMAN LEXICON)
This forthcoming thesaurus is also designed to tie in with LDOCE, of which it is partly a by-product. As Section III will make clear, our analysis of LDOCE definitions will have to rely on a thesaurus, but we do not know yet whether LOLEX will be available in machine-readable form.

SECTION II. TOWARDS A SEMANTICALLY ENRICHED SURFACE PARSER BASED ON LDOCE

I.- A lexically based syntax.

It stands to reason that automatic parsing programmes have to have access to at least two linguistic components : a grammar and a lexicon.

In most systems that we know something about, the grammar is a good deal more sophisticated than the lexicon. The latter includes only a small sub-part of the total lexicon for the language under study, while the grammar takes care of a large proportion of the basic grammatical structures.

We would like to explore a diametrically opposed approach : our starting-point is a sophisticated lexicon for core English and our aim is to make maximum use of the information it contains to keep our grammar within strict bounds.

An obvious first step in developing a parser based on LDOCE is to write algorithms that translate the various grammatical codes into scanning procedures . Most of these algorithms are fairly straightforward and have already been written. What we would like to focus on here is the simplification of the categorial component that such a lexically based syntax permits. Consider 3 :

3. The claim that he has succeeded is patently false. Since there is a code (namely, 5) that stipulates whether an element (in this case, a countable noun coded C - the whole code is therefore [$\mathfrak{C}_5$]) can be followed by a THAT-clause, we will not attempt to account for THAT-clauses via rewrite rules for the category NP, i. e. we won't have such a rule as :

NP ⟶ NP THAT S

Naturally enough, there is no LDOCE code stipulating that a noun can be followed by a relative clause (such a code would be meaningless since virtually all nouns can have a relative clause - if not a restrictive, then at least an appositive one - tagged on their right). We will therefore have to include relative clauses somewhere in our rewrite rules for the category NP. Here too, however, the lexical approach to syntax can prove useful. To show this, let us first define a CONCATENATION as a string every member of which is tied to some other by means of a LDOCE grammatical code (it requires the other member for the satisfaction of its code or it serves to satisfy the other member's code). The concept of CONCATENATION can be equated with that of CLAUSE if it is extended to cover :

1.- free elements, i. e. elements which are not bound to one particular word or phrase inside the clause (both sentential adjuncts and linking words such as conjunctions would fall into this category).
2.- a subject role, i. e. the creation of a link between a tensed V (the starting-point for the concatenation - see below) and an NP to be found on its right or on its left.

We have already looked into the mechanisms of tensed V searches and subject role assignments and we have found that various properties of English make the task of algorithmizing these mechanisms less formidable than it appears at first sight. The most prominent among these properties are the following :

1.- the conditions of use of the auxiliary DO;
2.- the fact that only tensed Vs require a subject;
3.- the fact that only the first (i. e. leftmost) member of a verbal group can bear tense;
4.- the fact that it must bear tense;
5.- the morphological contrast between verb and noun with respect to number (- S marks singular verbs but plural nouns).

Turning now to relative clauses, we see that we can characterize them with great ease : a relative clause is a concatenation that opens with a relative phrase (one of whose realizations is Ø and another the multi-purpose word THAT, so that a recognition procedure based on

the occurrence of particular morphemes is bound to fail in some cases) and that misses an NP (it is this second property that has to be regarded as essential).

The readers who are familiar with Hudson 1976 will have realized that the approach advocated here is nearer to Hudson's version of systemic grammar than to transformational grammar : we make full use of sister-dependencies, starting with the tensed V, which we believe to provide the best entry-point into the network of relationships woven by the various code-bearing elements in a sentence.

II.- Deep structure configurations.

It is obvious that our parser will have to be able to :

1.- recognize the situations in which the basic order of the constituents (i. e. the one stipulated in the scanning procedures associated with the grammatical codes) is disrupted under the effect of transformations such as PASSIVIZATION, TOPICALIZATION, RELATIVE CLAUSE FORMATION, GAPPING, ...)
2.- keep track of the constituents that have been moved.

We do not intend to deal with these points here but we would like to stress that the problems for RECOGNITION are very different from those for GENERATION. RAISING and EQUI, for instance, are rather formidable and problem-ridden rules from the point of view of generation but we shall argue that we do not need their counterparts for recognition purposes. We shall illustrate this point by looking at verb complementation - at the same time we will show that the syntactic potential of a verb can be used as a guide to its deep structure configuration.

In a VP the SYNTACTIC head is always the first, i. e. tensed verb. As we have seen, the way the parser builds up concatenations reflects this property. As for the SEMANTIC head, it is very often another verb than the first one. This, however, does not matter in so far as the auxiliaries and semi-auxiliaries (HAPPEN, SEEM, ...) do not have any semantic code associated with them and can therefore be regarded as semantically transparent : they have no effect whatsoever on the pairs that the semantic component will be called on to examine for compatibility. Consider such a sentence as 4 :

4.- My father seems to have been reading too many strips.

The starting-point for building the concatenation would be the tensed V, i. e. SEEMS : the concatenation would be allowed to grow both to the left (assignment of subject role to the NP 'my father') and to the right : the appropriate syntactic code for SEEMS is [I 3] here (i. e. followed by an infinitive with TO) :

My father seems to have
Subject NP / satisfies [I_3] code of SEEM

SEEM is not coded semantically, so that the semantic component would not be called on at this stage. In the next step, HAVE would be examined and its [I 8] code seen to be applicable ([I 8] specifies that the code-bearing element be followed by an EN-form) so that a new sister dependency would be established :

My father seems to have been
satisfies [I 8] code of HAVE

In similar fashion, BEEN would have an [I_3] code (i. e. + ING-form) satisfied by READING :

My father seems to have been reading

Neither HAVE nor BE are semantically coded with respect to the definitions that have been chosen on basis of the grammatical codes that are satisfied in the sentence - READING on the other hand, will be coded syntactically (it requires one NP as object-code [T_1]) and semantically (it requires a [+ HUMAN] subject). Since SEEM, HAVE and BEEN are semantically transparent, the semantic component will examine the pair My father and reading and find them to be compatible as a subject-verb configuration. But how does the parser know that my father is the subject of reading ? A very simple-minded rule states that there is no change in subject in a verbal complex as long as there is no interrupting NP; if there is one, it is to be regarded as the subject of the following verb(s) :

I want to read
I started to read
I happened to be reading
→ I subject of READ

I want you to read
I saw you reading
I made you read
→ you subject of READ

This rule admits of at least one exception, namely PROMISE :

I promised you to read (I subject of READ in spite of interrupting YOU).

Another problem relating to deep structure configurations is that of determining, in

V + NP + { (TO) + INFINITIVE / + ING-FORM }

structures, whether the NP is to be regarded as the object of the V or not (contrast 'I want him to go' with 'I persuaded him to go').

Instead of going into each deep structure distinction that can be drawn within the field of verb complementation, we will show that the verb classes which Akmajian and Heny 1975 (p. 364 and foll.) find it necessary to set up in their introduction to transformational grammar to account for deep structure distinctions (Figure 1) can be held apart on the basis of their surface structure potential as captured in their LDOCE grammatical codes.

Figure 1
Akmajian and Heny's verb classes
See appendix I .

The raised numbers on the features in the matrix below refer to the following list of test sentences :

1. I want to go
2. I want him to go
3. a) ? I want that he should go
   b) * I want that he goes
4. * I persuaded to go
5. I persuaded him to go
6. * I persuaded that he went
7. * I believe to have gone
8. I believe him to have gone
9. I believe that he has gone
10. I failed to go
11. * I failed him to go
12. * I failed that he went

CLASS NUMBER + ONE TYPICAL EXPONENT	CODES		
	T_3/I_3	V_3/X (to be)...	T_5/T_{5a}
I : WANT	$+^1$	$+^2$	$-^3$
II : PERSUADE	$-^4$	$+^5$	$-^6$
III : BELIEVE	$-^7$	$+^8$	$+^9$
IV : FAIL	$+^{10}$	$-^{11}$	$-^{12}$

The NP following the verb is its deep object only in the case of Class II verbs (I persuaded him to go ⇒ I persuaded him); there is no NP in Class IV (* I failed him to go) and the NP is not the object in Class I or in Class III (I want him to go ⇏ I want him; I believe him to have gone ⇏ I believe him).

As for PROMISE (not discussed in Akmajian and Heny 1975) it could be defined by means of the following feature row : + T_3, + T_5, + V_3 :

I promised to go (T_3)

I promised him to go (V_3)

I promised that I would go (T_5)

The NP between PROMISE and the TO-INFINITIVE is the object (as in the PERSUADE class) but it is not the subject of the infinitive.

SECTION THREE : LDOCE DEFINITIONS : AN IR APPROACH TO SEMANTIC AND KNOWLEDGE-OF-THE-WORLD INFORMATION.

LDOCE definitions convey semantic information in a fairly explicit, but non-formatted, form. Even though all definitions are written in a DEFINING VOCABULARY (not to be confused with a BASIC VOCABULARY - see below), no attempt has been made to stick to a limited number of DEFINING FORMULAE. To give an example of what we mean by DEFINING FORMULA, and to anticipate on what will be the main concern of this section, we wish to look at the class of INSTRUMENTS. In theory, it could be agreed by the dictionary-makers that all instruments have to include the phrase "instrument used for Ving" in their definitions. In such a defining formula the word INSTRUMENT would be a DEFINING PRIMITIVE and the predicate USED FOR would be a DEFINING RELATION (in this case, between an instrument and a predicate). Such a kind of formatted definition would be less precise and less exact, but infinitely more usable, than a common type definition. Smith and Maxwell 1973 (p2) point out that in a typical dictionary approximately 50 % of the vocabulary appears in the definitions. LDOCE is a major improvement on such a typical dictionary in that its defining

vocabulary is restricted to some 2,000 items (used to define some 60,000 entries). My purpose in this section is to reflect on the possibility of turning a significant number of LDOCE definitions into fully formatted ones (i.e. making use of defining formulae).

Consider the sentence :

I saw the man in the park with a telescope

[Woods in Rustin 1973, p. 172]

The PREFERRED reading is the one that associates 'with a telescope' with the predicate 'saw' rather than with either of the NP heads 'man' or 'park' : 'saw with a telescope' rather than 'man with a telescope' or 'park with a telescope'. If we had available a formatted definition of TELESCOPE ("instrument used for seeing ..."), there would be no problem in a system of preferential semantics : the link between 'saw' and 'telescope' (embodied in the definition of the latter) would lead to the selection of the preferred reading on the basis of the DENSEST MATCH FIRST principle. As a matter of fact, the LDOCE definition for 'telescope' is very nearly what we need :

> "a tubelike scientific instrument used for seeing distant objects by making them appear nearer and larger"

A simple matching procedure between our suggested defining formula for instruments and the LDOCE definition for 'telescope' would have been sufficient in this case. The problem, of course, is that there is absolutely no guarantee that the defining formula will be part of the definition of all instruments. HAMMER, for instance, is defined as :

> "a tool with a heavy head for driving nails into wood or for striking things to break them or move them" (Definition 1)

No simple procedure will associate INSTRUMENT with HAMMER. The fact that LDOCE makes use of a defining vocabulary, however, ensures that the defining noun (TOOL in this case) is a member of a finite list, namely the LDOCE defining vocabulary itself. One can go a step further and make the hypothesis that the defining noun will belong to a definite subset within the defining vocabulary. One can go through that vocabulary and select the words that could stand for INSTRUMENTS. The subset that this procedure yields can fairly easily be divided into two further groups : on the one hand one finds such general words as TOOL and APPARATUS (note that the latter would not be included in a BASIC VOCABULARY) which could also be used in defining formulae; on the other hand one has to include such specific items as BOAT, BICYCLE and GUN, which are instances of instruments. The second group is of course much more problematic than the first : one has to be concerned with TYPICAL instruments, otherwise all PHYSICAL OBJECTS would have to be included :

He hit her with the tail of a dead snake.

The INSTRUMENT reading of the 'with' - phrase is not due to any intrinsic property of either 'tail' or 'snake', but rather to four factors :

a) WITH often introduces an instrumental adjunct;
b) the 'with' -phrase in this sentence cannot be read as postmodifying 'her';
c) it cannot be read as an accompaniment adjunct for 'he' either;
d) the predicate 'hit' can take an instrumental adjunct.

The reader will have noticed that factors a, c and d also apply - mutatis mutandis - to the example involving the predicate SEE. This, however, does not imply that the link between TELESCOPE and SEE was of no use in preferring the instrument reading for the 'with' -phrase - note that 'with a telescope' COULD postmodify the NP heads 'man' and 'park'; besides, even if it could not, we would still have to find a way of telling the system and this task may well prove considerably more formidable than that of associating instruments and predicates.

The following items in the LDOCE defining vocabulary could be regarded as making up the subset for the concept INSTRUMENT :

GROUP I

apparatus
instrument
machine
machinery
means
organ
tool

GROUP II

arm [R]
arms [R]
army
arrow
axe
beak
belt
bicycle
boat
boot
brain
brick
bridge
brush
bullet
bus
button
camera
candle
car
card
cart
chain
coin
comb
control [R]
cover
curtain
drum
engine [R]
factory
fence
fork
gate
gift
glass
gun
hammer
hand [R]
handle [R]
hook
key
knife
knot
ladder
lamp
law
letter
map
mat
medicine
message
microscope
mirror
motor [R]
nail
needle
network
pan
pen
pin
plane
poison
pole
post
pot
prayer
proof
pump
radio
railway
road
rod
roof
rope
sail
scales
scissors
screw
servant [R]
shield
shoe
sign
signal
slave [R]
spade
spring
stairs
stone
string
support
sword
system
taxi
telephone
telegram
telegraph
television
thread
thumb
ticket
tooth
train
trap
vehicle
weapon [R]
wheel
whip
whistle

NOTES

1. For all items in both groups, POS (Part of Speech) = n
2. All items in Group I - except MEANS, which is itself a head - appear under the head TOOL in Roget's Thesaurus.
3. In Group II the items followed by [R] occur in Roget's Thesaurus under the head TOOL.

4. The underlined items in Group II are more general and could perhaps be singled out in a third group, intermediate between I and II.

Obviously, the lists as such are not sufficient for our purpose : words such as SPRING and MEDICINE are not relevant to the INSTRUMENT concept in some of their most frequent uses - for our purposes the defining vocabulary should not have been limited to a list of LEXICAL ITEMS; in case of polysemic words, numbers should have been added to make clear which definitions were to be associated with the defining word : SPRING 1 (= a source), 2 (= a season), 4 (= elasticity), 5 (= an active healthy quality) and 6 (= an act of springing), are not relevant to the INSTRUMENT concept. Since - in theory - the noun SPRING can be used with all six meanings in LDOCE definitions, its inclusion in our list is liable to prove detrimental : it can lead the system to associate the INSTRUMENT concept with a defining word that has nothing to do with instrumentality.

Going back to the LDOCE definition for HAMMER, we realize that the algorithm that will associate instruments and predicates will have to take into account, not only the Ving form (in the formula 'for Ving'), but also its object; otherwise a hammer is going to be thought of as a kind of vehicle :

Compare

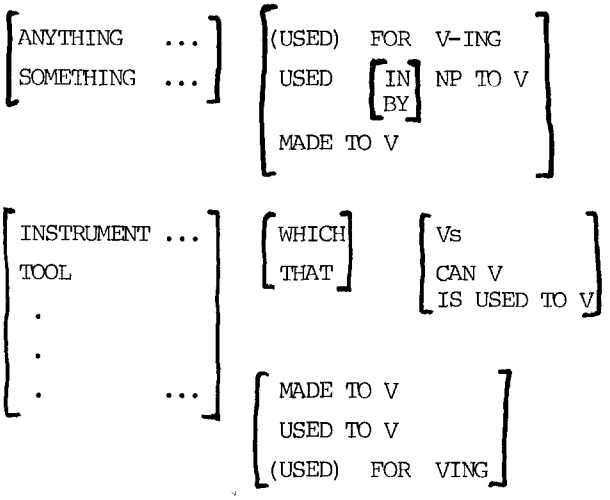
a tool ... for driving DRIVE[1] 2/3 in LDOCE

with

a tool ... for driving nails DRIVE[1] 5/6 in LDOCE

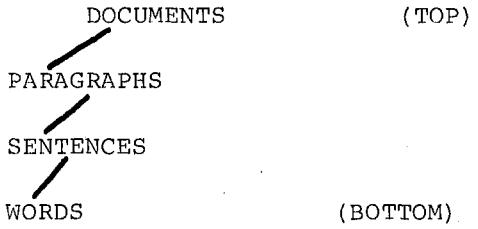
A second difficulty that we must face up to is that there may be no defining NOUN, but an all-purpose indefinite such as SOMETHING or ANYTHING. In that case, however, the INSTRUMENT concept is likely to be expressed somewhere else in the definitions, by means of (USED) FOR Ving, for instance. This last point leads us to an examination of the various ways in which the link between instrument and predicate can be conveyed; the existence of a defining vocabulary is a help but the range of SYNTACTIC possibles remains enormous; however, there is something that could be called the LEXICOGRAPHICAL TRADITION and familiarity with that tradition can help cut down on the number of possible formulae - the following stand a good chance of being rather heavily used :

FIG. 2



Obviously, processing LDOCE definitions is a lot of work in terms of the necessary algorithms and in terms of the sheer volume of language data to be scrutinized. We suggest that a useful approach is provided by IR (Information Retrieval) techniques as embodied in

the IBM system known as STAIRS. STAIRS processes various objects, which can be worked into the following hierarchy :



The various paragraphs of a given document can be assigned labels, so that the search within a single document can be oriented.
STAIRS provides a number of SEARCH OPERATORS, which will be briefly characterized below. A STAIRS search operator can be used to link any of the following three categories :

1. word tokens, e.g. DISEASES, APPLIES, COMPUTERIZED, IF;
2. a) stems, e.g. RUN-, ANTAGONIZ-, MOTHER- (the use of a character mask enables the system to assign RUNNING, RUNS, RUNNER, RUNNERS, etc. to the stem RUN-) and b) lexemes for which STAIRS generates the morphological variants;
3. any expression consisting of elements of type 1 or 2 linked by STAIRS SEARCH OPERATORS (the definition is therefore recursive, and allows any degree of embedding).
   Let A and B stand for elements belonging to any of the above three types. The operators that STAIRS works with are the following :

   A ADJ B : A and B occur next to each other and in that order in the document to be retrieved.

   A SYN B : A and B are to be regarded as synonyms for a given search operation

   A WITH B : A and B occur in the same sentence

   A SAME B : A and B occur in the same paragraph

   NOT B : B doesn't occur in the document to be retrieved

   A AND B : both A & B

   A OR B : inclusive OR

   A XOR B : exclusive OR

In our system the STAIRS hierarchy would correspond to the following :

A. A DOCUMENT -A HOMOGRAPH (e.g. DOUBT[2]) or AN ENTRY WITHOUT HOMOGRAPHS (e.g. PONDEROUS) in the LDOCE file

B. A PARAGRAPH -a specified FIELD within A, e.g. POS (part of speech), GRAMMATICAL CODE, SEMANTIC CODE, TEXT OF THE DEFINITION,...

C. A SENTENCE -any sentence included in the text of a given definition

D. A WORD -the various words within a definition or the various codes and POS within a code field or a POS field.

It will be apparent that in order to rewrite FIG. 2 as a set of search operations in a STAIRS - like system we need to be able to refer to specified morphological analyses. Moreover, V and NP are neither word tokens nor stems : they refer to categories (respectively lexical and phrase structure) and we will have to extend the possibilities of the system so that such categories can be included in the expressions guiding the search and retrieval operations. Phrase structure categories are a hard nut to crack, and we will probably have to do without them in a first stage, but lexical categories such as V can be housed in a STAIRS - like system : a V is the name of any document that includes V among its POS - paragraph.

Leaving aside the reference to NPs, FIG. 2 can be rewritten as a complex STAIRS - like search operation with six levels. The embedding of STAIRS expressions within STAIRS expressions gives rise to the use of labels such as

A1, B1, etc; the colon is to be read as "can be defined as" :

A1 OR A2

```
A1 : B1 WITH B2
     B1 : 'ANYTHING' OR 'SOMETHING'
     B2 : C1 OR C2 OR C3
          C1 : 'USED' WITH 'FOR'
               ADJ V-ING
          C2 : 'FOR' ADJ V-ING
          C3 : 'MADE' ADJ 'TO' ADJ V

A2 : B3 WITH B4
     B3 : 'INSTRUMENT' OR SYN-
          INSTRUMENT
     B4 : C4 OR C5 OR C6 OR C7
          C4 : D1 ADJ D2
               D1 : 'WHICH' OR 'THAT'
               D2 : Vs OR E1 OR E2
                    E1 : 'CAN' ADJ V
                    E2 : 'IS' ADJ
                         'USED'
                         WITH 'TO'
                         ADJ V
          C5 : 'MADE' ADJ 'TO' ADJ V
          C6 : 'USED' WITH 'TO' ADJ V
          C7 : D3 OR D4
               D3 : 'USED' WITH
                    'FOR' ADJ V-ING
               D4 : 'FOR' ADJ V-ING
```

Note that to be really useful, the algorithms that associate predicates and instruments should have access to a thesaurus-like classification of predicates. Take for instance the definition of MICROSCOPE :

"an instrument that makes very small near objects seem larger, and so can be used for examining them"

The preferential link is between MICROSCOPE and EXAMINE and a sentence such as :

"He examined the new virus with an extremely powerful microscope"

will be interpreted the right way. But what about

"He studied the new virus with an extremely powerful microscope" ?

We could get around this problem if we had access to a thesaurus like Roget's: since STUDY and EXAMINE share a SUBHEAD in Roget's, viz. SCAN in 438 VISION, a link between STUDY and MICROSCOPE could be established.

BIBLIOGRAPHY

-LONGMAN DICTIONARY OF CONTEMPORARY ENGLISH, 1978.

-LONGMAN DICTIONARY OF ENGLISH IDIOMS, 1979.

-AKMAJIAN AND HENY, 1975 : Akmajian, A. and Heny, F., An Introduction to the Principles of Transformational Syntax, MIT Press, Cambridge and London, 1975.

-HUDSON 1976 : Hudson, R.A., Arguments for a Non-transformational Grammar, The University of Chicago Press, Chicago and London, 1976.

-Roget's Thesaurus of English Words and Phrases, Penguin Books, 1962 Longman Edition.

-RUSTIN, 1973 : Rustin, R. (ed.), Natural Language Processing, Algorithmics Press, New York, 1973.

-SMITH & MAXWELL 1973 : Smith, R.N., and Maxwell, E., An English Dictionary for Computerized Syntactic and Semantic Processing Systems, mimeo, Pisa 1973.

Appendix I

Figure 1 Akmajian and Heny's verb classes

CLASS I	: prefer, want, hate, like, hope, desire, love.
CLASS II	: force, persuade, allow, coax, help, order, permit, make, cause.
CLASS III	: believe, assume, know, perceive, find, prove, understand, imagine.
CLASS IV	: condescend, dare, endeavour, fail, manage, proceed, refuse.
CLASS V	: seem, appear, happen, turn out.