Progress in natural language understanding—An application to lunar geology

by W. A. WOODS

Bolt Beranek and Newman Inc.
Cambridge, Mass.

INTRODUCTION

The advent of computer networks such as the ARPA net (see e.g., Ornstein et al.[3]) has significantly increased the opportunity for access by a single researcher to a variety of different computer facilities and data bases, thus raising expectations of a day when it will be a common occurrence rather than an exception that a scientist will casually undertake to use a computer facility located 3000 miles away and whose languages, formats, and conventions are unknown to him. In this foreseeable future, learning and remembering the number of different languages and conventions that such a scientist would have to know will require significant effort—much greater than that now required to learn the conventions of his local computing center (where other users and knowledgeable assistance is readily available). The Lunar Sciences Natural Language Information System (which we will hereafter refer to as LUNAR) is a research prototype of a system to deal with this and other man-machine communication problems by adapting the machine to the conventions of ordinary natural English rather than requiring the man to adapt to the machine.

English as a query language

There are two important reasons why one might want to use English as a mode of communication between a man and a machine. First, the man already knows his natural language and if he is to use a particular computer system or data base only occasionally or as a minor part of his work, then he may not have the time or inclination to learn and remember a formal machine language. Second, the human thinks in his native language, and if the mode of communication involves the free and immediate communication of ideas to the machine which the user is conceiving in the course of the interaction, then the additional effort required for him to translate his ideas into another language more suitable to the machine may slow down or otherwise interfere with the interaction. English is therefore an attractive medium because the human can express his ideas in the form in which they occur to him.

State of the art

Although the state of the art in "understanding" natural language by machine is still very limited, significant advances in this area have been made in recent years. Since Simmons' first survey of question answering systems,[4] our understanding of the mysterious "semantic interpretation" component has been made more clear by work such as Woods,[7,8] and the techniques for mechanically parsing natural language sentences have been advanced by the advent of transition network grammars and their parsing algorithms.[9,10,11] Recently, Terry Winograd's blocks world system[6] has demonstrated the potential of some of these techniques—especially those of procedural semantics. The field is now at the point where prototype applications to real problems can make significant contributions to our understanding of the problems of natural language communication with machines. It must be realized, that such applications are still essentially research vehicles, since the problems of mechanical understanding of natural language remain far from solution. However, by using real problems, rather than imaginary toy problems, as the vehicles for such research, one cannot only focus the effort on problems in need of solution, but may also reap the additional benefit of producing a system which will perform a task which someone really wants done.

Natural language understanding

I want to distinguish here between the objectives of this research, which I will call "natural language understanding" research, and the development of so called "English-like" languages and querying systems. There have been a number of computer question answering systems developed for special applications or for general purpose data management tasks which use English words and English syntactic constructions and call their languages "English-like." By this they mean that, although the statements in the language may look more or less like ordinary English sentences, the language makes no attempt to encompass the totality of English constructions that a user might want to use. Also, the interpretations of those sentences by the machine may differ from those which a user would assume without instruction. Essentially, the designer of such a system is trying to go as far as he can toward English with some set of techniques which is either easily implementable or computationally efficient, and when he encounters an English phenomenon which does not fit conveniently within those techniques or which is troublesome, he legislates it out of his language or provides for some restricted or modified form of the original phenomenon. In the kind of research I am describing, one seeks techniques for handling these phenomena.

The design of a *natural* English understanding system as a product is one that is slightly beyond the current state of the art and requires considerable linguistic research and further development of computational techniques before it will become a possibility. The construction of research prototypes such as LUNAR as testbeds and foci for the necessary research is an essential prerequisite for the eventual development of such products. As spinoffs from this research we will achieve better and better "English-like" systems as we go along.

Habitability

Most English-like systems, although they use English words and English syntactic constructions, have a syntax as limited and as formal as FORTRAN and require comparable amounts of effort to learn, remember, and use. This is not to say that these languages merely permit one to stick English words into some fixed English formats (although systems of that sort have also gone by the name of "English-like")—the language may contain a fairly complex grammar and a parsing algorithm for analyzing sentences with it. However, the "English-like" grammar is at best a restriction of what a full English grammar would be and at worst may bear little resemblance to ordinary English. The ease or comfort with which a user can learn, remember and obey the conventions of such a language has been considered by William Watt[5] and given the name "habitability." A habitable system is one in which the user will not be constantly overgeneralizing the capabilities of the system and venturing beyond its conventions, nor will he painfully adhere to a miniscule subset of the capabilities for fear of misinterpretation. It is very difficult to evaluate the habitability of a natural language or English-like question answering system without actual hands-on experience, and reported data on the subject (with the exception of the extremely inadequate statistics which will be reported here for LUNAR) is essentially non-existent. However, it has been my opinion of every English-like system which I have encountered that its habitability is very low, and although its techniques represent a significant advance in the state of the art, I am far from satisfied with the habitability of LUNAR.

THE LUNAR SYSTEM

LUNAR was originally developed with support from the NASA Manned Spacecraft Center as a research prototype for a system to enable a lunar geologist to conveniently access, compare, and evaluate the chemical analysis data on lunar rock and soil composition that was accumulating as a result of the Apollo moon missions. The objective of the research was to develop a natural language understanding facility sufficiently natural and complete that the task of selecting the wording for a complex request would require negligible effort for the geologist user.

The LUNAR system processes English requests in three successive phases:

(i) syntactic analysis using heuristic (including semantic) information to select the most "likely" parsings,
(ii) semantic interpretation to produce a formal representation of the "meaning" of the query to the system,
(iii) execution of this formal expression in the retrieval component to produce the answer to the request.

The language processor in LUNAR makes use of a general parsing algorithm for transition network grammars and a general rule-driven semantic interpretation procedure which were developed at Harvard University and at BBN over a period of years and which have been reported on in the literature.[7–11] In addition, LUNAR contains a grammar for a large subset of English, a set of semantic interpretation rules for interpreting requests for chemical analyses, ratios, etc., and a dictionary of approximately 3500 words, plus functions for setting up and interrogating a data base, computing averages and ratios, etc. The LUNAR system is described in detail in a technical report.[12]

All of the components of the system are implemented in BBN-LISP on the PDB-10 computer at BBN in Cambridge, Mass., running under the TENEX time sharing system with hardware paging and a virtual core memory for each user of up to 256K. The system is operational in two 256K tasks (called "forks")—one containing the parser, interpreter, grammar and dictionary, and the other containing the data base and retrieval functions. Although there is considerable overhead in running time for programs written in LISP and executed in a paged environment, the flexibility of this system has been a critical factor in the development of the present level of capability.

The data base

LUNAR contains two data base files provided by NASA MSC. One is a 13,000 entry table of chemical, isotope, and age analyses of the Apollo 11 samples extracted from the reports of the First Annual Lunar Science Conference, and the second is a keyphrase index to those reports. In this paper we will consider only the first. This table contains entries specifying the concentration of some constituent in some phase of some sample, together with a reference to the article in which the measurement was reported. (There are generally several entries for each combination of sample, phase, and constituent—measured by different investigators.)

The formal query language

The data base of the LUNAR system is accessed by means of a formal query language into which the input English requests are translated by the language processing component. The query language is essentially a generalization of the predicate calculus which could either be manipulated as a symbolic expression by a formal theorem prover to derive intensional inferences or be executed directly on the data base to derive extensional inferences. Only the latter, extensional inference facility, is used in LUNAR.

The query language contains three kinds of constructions: *designators*, which name objects or classes of objects in the data base (including functionally determined objects), *propositions*, which are formed from predicates with designators for arguments, and *commands*, which take arguments and initiate actions. For example, S10046 is a designator for a particular sample, OLIV is a designator for a certain mineral (Olivine), and (CONTAIN S10046 OLIV) is a proposition formed by substituting designators as arguments to the predicate CONTAIN. TEST is a command for testing the truth value of a proposition. Thus, (TEST (CONTAIN S10046 OLIV)) will answer yes or no depending on whether sample S10046 contains Olivine. Similarly, PRINTOUT is a command which prints out a representation for a designator given as its argument.

The major power and usefulness of the query language comes from the use of a quantifier function FOR and special enumeration functions for classes of data base objects to carry out extensional quantification on the data base. The format for a quantified expression is:

(FOR QUANT X / CLASS : PX ; QX)

where QUANT is a type of quantifier (EACH, EVERY, SOME, THE, numerical quantifiers, comparative quantifiers, etc.), X is a variable of quantification, CLASS determines the class of objects over which quantification is to range, PX specifies a restriction on the range, and QX is the proposition or command being quantified. (Both PX and QX may themselves be quantified expressions.)

The specification of the CLASS over which quantification is to range is performed in the system by special enumeration functions which (in addition to whatever other parameters they might have) take a running index argument which is used as a restart pointer to keep track of the state of the enumeration. Whenever FOR calls an enumeration function for a member of the class, it gives it a restart pointer (initially NIL), and each time the enumeration function returns a value, it also returns a new restart pointer to be used to get the next member.

The use of enumeration functions for quantification frees the FOR function from explicit dependence on the structure of the data base—the values returned by the enumeration function may be searched for in tables, computed dynamically, or merely successively accessed from a precomputed list. A general purpose enumeration function SEQ can be used to enumerate any precomputed list. For example:

(FOR EVERY X1 / (SEQ TYPECS) : T ;
(PRINTOUT X1))

is an expression which will printout the sample numbers for all of the samples which are type C rocks (i.e. breccias).

The principal enumeration function for the chemical analysis data base is the function DATALINE which takes arguments designating a data file, a sample, a phase name, and a constituent. DATALINE enumerates the lines of the data file which deal with the indicated sample/phase/constituent triple. Other complex enumeration functions are NUMBER and AVERAGE which take an argument format similar to the FOR function and perform counting and averaging functions.

CAPABILITIES OF THE CURRENT SYSTEM

The current LUNAR prototype permits a scientist to easily compare the measurements of different researchers, compare the concentrations of elements or isotopes in different types of samples or in different phases of a sample, compute averages over various classes of samples, compute ratios of two constituents of a sample, etc.—all in straightforward natural English. The system removes from the user the burden of learning the detailed formats and codes of the data base tables, or learning a special programming language. For example, the system knows the various ways that a user may refer to a particular class of samples, it knows whether a given element is stored in the data base in terms of its elemental concentration or the concentration of its oxide, it knows what abbreviations of mineral names have been used in the tables, etc., and it converts the user's request into the appropriate form to agree with the data base tables. Thus, the system has made significant strides toward the goal of habitability. The following examples will illustrate some of the kinds of operations LUNAR performs.

The most typical example of a request which a geologist might make to the LUNAR system is illustrated by the protocol:

```
38**(WHAT IS THE AVERAGE CONCENTRATION OF ALUMINUM IN HIGH ALKALI ROCKS)
***
PARSING
1331 CONSES
4.987 SECONDS
INTERPRETING
2427 CONSES
11.025 SECONDS
INTERPRETATIONS:
(FOR THE X13 / (SEQL (AVERAGE X14 / (SSUNION X15 / (SEQ TYPEAS) : T ; (DATALINE (WHQFILE X15)
X15 (NPR* X16 / (QUOTE OVERALL)) (NPR* X17 / (QUOTE AL203)))) : T)) : T ; (PRINTOUT X13))

BBN LISP-10 03-09-72 . . .
EXECUTING
(8.134996 . PCT)
```

(Here, the system has typed the two asterisks, the user typed the question, beginning and ending with parentheses, and the system typed the rest. The comments 1331 CONSES and 4.987 SECONDS give a record of the memory resources and the time used during the parsing phase. A similar record is generated for the interpretation phase: The expression following the comment INTERPRETATIONS: is the formal retrieval program which is executed in the data base to produce the answer). This request illustrates a number of features of the system:

1. The user types the question exactly as he would say it in English (terminal punctuation is optional and was omitted in the example).
2. The system has translated the phrase "high alkali rocks" into the internal table form TYPEAS.
3. The system has filled in an assumed OVERALL phase for the concentration since the request does not mention any specific phase of the sample in which the concentration is to be measured.
4. The system is capable of computing answers from the data base as well as simply retrieving them (the average was not stored information).

Perhaps the simplest operation which the system will perform for the user is to collect and list selected portions (not necessarily contiguous) of the data base. For example, for a request "Give me all analyses of S10046," the system protocol would be:

```
***
37**(GIVE ME ALL ANALYOES OF S10046)
***
PARSING
1456 CONSES
9.445 SECONDS
INTERPRETING
2112 CONSES
8.502 SECONDS
INTERPRETATIONS:
(DO (FOR EVERY X9 / (SSUNION X12 / (SEQ MAJORELTS) : T ; (DATALINE (WHQFILE (NPR* X19 / (QUOTE
S10046))) (NPR* X10 / (QUOTE S10046)) (NPR* X11 / (QUOTE OVERALL)) X12)) : T ; (PRINTOUT X9)))

BBN LISP-10 03-09-72 . . .
EXECUTING

I HAVE 15 HITS
DO YOU WANT TO SEE THEM? YES
```

3956	S10046	OVERALL	S102	44.06752	PCT	D70-235
3967			T102	8.3405		
3968				6.50559		D70-254
3865			AL203	11.7149		D70-235
3900			FEO	16.9818		
3901				15.438		D70-254
3928			MNO	.20659		D70-235
3929				.22725		D70-254
3927			MGO	9.11845		D70-235
3875			CAO	13.71216		
3917			K20	.20478		
3918				.19515		D70-242
3919				.14455		D70-254
3933			NA20	.4718		D70-235
3934				.50146		D70-254

This example illustrates some additional features of the system. Again, since no phase was mentioned, the system assumed the OVERALL phase (i.e. the rock as a whole). If the user had wanted to see all the phases, he could have said explicitly "for all phases." Similarly, since no specific elements or isotopes were mentioned, the system assumed a standard list of major elements was intended (Our geologist informant assures us that this is what a geologist would mean by such a question). Again, if the user really wanted to see all chemical element analyses, he could say so explicitly.

The comment I HAVE 15 HITS DO YOU WANT TO SEE THEM? illustrates another feature of the system. If the result of a request is more than 5 "hits" (effectively 5 lines of output), the system types this comment and gives the user the option of listing them offline.

In addition to averaging and listing, the system can also compute ratios, count, and interpret some anaphoric references and comparatives as indicated in the following examples:

```
31**(HOW MANY BRECCIAS CONTAIN OLIVINE)
***
PARSING
815 CONSES
4.633 SECONDS
INTERPRETING
1514 CONSES
7.29 SECONDS
INTERPRETATIONS:
(FOR THE X12 / (SEQL (NUMBER X12 / (SEQ TYPECS) : (CONTAIN X12 (NPR* X14 / (QUOTE OLIV)) (QUOTE
NIL)))) : T ; (PRINOUT X12))

BBN LISP-10 03-09-72 . . .
EXECUTING
(5)

-----
***
32**(WHAT ARE THEY)
***
PARSING
487 CONSES
2.755 SECONDS
INTERPRETING
1158 CONSES
4.053 SECONDS
INTERPRETATIONS:
(FOR EVERY X12 / (SEQ TYPECS) : (CONTAIN X12 (NPR* X14 / (QUOTE OLIV)) (QUOTE NIL)) ; (PRINTOUT
X12))

BBN LISP-10 03-09-72 . . .
EXECUTING
S10019
S10059
S10065
S10067
S10073
-----
***
34**(DO ANY SAMPLES HAVE GREATER THAN 13 PERCENT ALUMINUM)
***
PARSING
981 CONSES
4.614 SECONDS
INTERPRETING
902 CONSES
3.566 SECONDS
INTERPRETATIONS:
(TEST (FOR SOME X16 / (SEQ SAMPLES) : T ; (CONTAIN' X16 (NPR* X17 / (QUOTE AL203)) (GREATERTHAN
13 PCT))))

BBN LISP-10 03-09-72
EXECUTING
YES.
T
-----
```

```
***
35**(LIST K / RB RATIOS FOR BRECCIAS)
***
PARSING
662 CONSES
3.366 SECONDS
INTERPRETING
1642 CONSES
6.537 SECONDS
INTERPRETATIONS:
(DO (FOR GEN X9 / (SSUNION X10 / (SEQ TYPECS) : T ; (RATIO (QUOTE KO2) (QUOTE RB) X10 (NPR* X11 /
(QUOTE OVERALL)))) : T ; (PRINTOUT X9)))

BBN LISP-10 03-09-72
EXECUTING
I HAVE 17 HITS
DO YOU WANT TO SEE THEM? YES
(472.2222  S10018  D70-205)
(473.5884  S10018  D70-242)
(518.2477  S10019  D70-218)
(345.4411  S10019  D70-256)
(463.3003  S10021  D70-242)
(568.8333  S30046  D70-235)
(462.4408  S10046  D70-242)
(408-2933  S10048  D70-220)
(566.1499  S10056  D70-235)
(480.1913  S10059  D70-253)
(481.85    S10060  D70-235)
(457.9177  S10060  D70-242)
(487.5714  S10060  D70-248)
(489.1304  S10061  D70-205)
(458-9973  S10065  D70-236)
(473.1551  S10065  D70-258)
(500.173   S10073  D70-215)
---
```

The system also understands restrictive relative clauses and certain adjective modifiers (some of which cause restrictions on the range of quantification of the noun phrase and some of which change the interpretation of the noun they modify). Some other modifiers (such as "lunar" modifying "samples") are known to be redundant and are deliberately ignored. The following example contains all three:

```
46**(LIST MODAL PLAG ANALYSES FOR LUNAR SAMPLES THAT CONTAIN OLIV)
***
PARSING
1099 CONSES
4.346 SECONDS
INTERPRETING
2774 CONSES
12.33 SECONDS
INTERPRETATIONS:
(DO (FOR GEN X20 / (SSUNION X1 / (SEQ SAMPLES) : (CONTAIN X1 (NPR* X3 / (QUOTE OLIV)) (QUOTE
NIL)) ; (DATALINE (WHQFILE X1) X1 OVERALL (NPR* X4 / (QUOTE PLAG)))) : T ; (PRINTOUT X20)))

BBN LISP-10 03-09-72
EXECUTING
I HAVE 13 HITS
DO YOU WANT TO SEE THEM? YES
1679   S10020   OVERALL   PLAG   30.7   ***   D70-159   0
1680                             21.4         D70-173   31
1681                             28.5                   40
```

1682		24.6	D70-305	0
2141	S10022	15.6	D70-179	
3109	S10044	33.1	D70-154	41
3110		34.1		42
4440	S10047	37.8	D70-159	0
5796	S10058	37.1	D70-155	
8582	S10072	20.4	D70-173	
8583		18.5	D70-179	
9311	S10084	22.0	D70-186	
9312		15.0	D70-304	

The structure of the formal query language for accessing the data base and the techniques for semantic interpretation enable the user to make very explicit requests with a wide range of diversity within a natural framework. As a natural consequence of the arrangement, it is possible for the user to combine the basic predicates and functions of the retrieval component in ways that were not specifically anticipated, to ask questions about the system itself. For example, one can make requests such as "List the phases.", "What are the major elements?", "How many minerals are there?", etc. Although these questions are not likely to be sufficiently useful to merit special effort to handle them, they fall out of the mechanism for semantic interpretation in a natural way with no additional effort. If the system knows how to enumerate the possible phases for one purpose, it can do so for other purposes as well. Furthermore, anything that the system can enumerate, it can count. Thus, the fragmentation of the retrieval operations into basic units of quantifications, predicates, and functions provides a very flexible and powerful facility for expressing requests.

EVALUATION

As I said in my introduction, the construction and evaluation of research prototypes such as LUNAR is an essential prerequisite to the development of natural English understanding systems. It is natural at this point to ask, "How close to that goal are we?" "How natural is the communication with LUNAR?" In this section, we will consider some of the data both statistical and anecdotal which we have available. It is admittedly grossly inadequate to the task but may help to give some impression of what can be achieved with the techniques used in LUNAR and also some of the problems that still remain with this and other natural language understanding systems.

Demonstration of the prototype

At the Second Annual Lunar ~~Science~~ Conference~~, in~~ Houston, Texas, January 11-13, 1971, a version of LUNAR system was run as a demonstration twice a day for three days. During this time the lunar geologists attending the conference were invited to ask questions of the system. Approximately 110 requests were processed, many of which were questions whose answers would contribute to the work of the requestor and not merely "toy" questions to see what the system would do. These requests were limited to questions which in fact dealt with the data base of the system (many people asked their questions before they could be told what the data base contained) and were restricted to not contain comparatives (which we did not handle at the time). The requests were freely expressed, however, without any prior instructions as to phrasing and were typed into the system exactly as they were asked.

Of 111 requests entered into the system during the three days, 10 percent of them failed to perform satisfactorily because of parsing or semantic interpretation problems. Another 12 percent failed due to trivial clerical errors such as dictionary coding errors which were easily corrected during or immediately after the demonstration. The remaining 78 percent of the requests were handled to our complete satisfaction, and with the correction of trivial errors, 90 percent of the questions expressed fell within the range of English understood by the system. This performance indicates that our grammar and semantic interpretation rules, which were based on the information of a single geologist informant, did indeed capture the essential details of the way that geologists would refer to the objects and concepts contained in our data base. Examples of the requests which were processed are:

```
(GIVE ME THE AVERAGE SM ANALYSIS OF TYPE A ROCKS)
(WHAT IS THE AVERAGE MODAL CONCENTRATION OF ILMENITE IN TYPE A ROCKS?)
(GIVE ME EU DETERMINATIONS IN SAMPLES WHICH CONTAIN ILM)
(GIVE ME ALL K / RB RATIOS FOR BRECCIAS).
(WHAT BE ANALYSES ARE THERE?)
(GIVE ME OXYGEN ANALYSES IN S10084)
(WHAT SAMPLES CONTAIN CHROMITE?)
(WHAT SAMPLES CONTAIN P205?)
(GIVE ME THE MODAL ANALYSES OF P205 IN THOSE SAMPLES)
(GIVE ME THE MODAL ANALYSES OF THOSE SAMPLES FOR ALL PHASES)
```

(DOES S10046 CONTAIN SPINEL?)
(WHAT PHASES DOES S10046 HAVE?)
(WHAT IS THE AVERAGE CONCENTRATION OF IRON IN ILMENITE)
(GIVE ME REFERENCES ON SECTOR ZONING)
(GIVE ME REFERENCES ON ABYSSAL BASALTS)
(GIVE ME ALL IRON / MAGNESIUM RATIOS IN BRECCIAS)
(GIVE ME ALL SC46 ANALYSES)
(WHAT SOILS CONTAIN OLIV)
(GIVE ME ALL OLIV ANALYSES OF S10085)
(WHAT ARE ALL TUNGSTEN ANALYSES?)
(GIVE ME IRON ANALYSES FOR PLAGIOCLASE IN S10022)
(GIVE ME ALL ZIRCONIUM CONCENTRATIONS IN ILMENITES)

Anecdotal data

The above statistics have to be evaluated with several grains of salt. However, they do give some impression of the habitability of the system to lunar geologists for which it was tailored. The results must be balanced by the experience of non-geologists who have tried to use the system. In one anecdotal example, which is perhaps typical of the failures of LUNAR and other attempted natural language understanding systems, the first question asked of the system by a graduate student in psychology (given only the instructions "ask it a question about the moon rocks") was "What is the average weight of all your samples?" This sentence failed even to parse due to the fact that the grammar of the system did not know that some determiners in English (such as "all") can be used as a "predeterminer" in addition to another determiner in the same noun phrase. Both "What is the average weight of all samples?" and "What is the average weight of your samples?" would have parsed. Assuming that the request had parsed (the grammar has now been expanded to the point where it would), the system would still not have understood it for several reasons. First, the system contains no semantic rules for interpreting ownership or possession (no one had ever attributed ownership of the samples to the system before). Secondly, LUNAR's data base does not contain information about the weights of samples and consequently there are no semantic rules for interpreting "weight of." This same student had to ask 5 successive questions before he found one that the system could understand and answer. This of course is not surprising for questions that are not even constrained to be about the contents of the data base. On another occasion, a class of graduate students in information retrieval given an appropriate introduction spent an hour and a half asking it questions and found only two that it failed to handle correctly.

The difference in performance for geologist users and non-geologists is not just that the geologists use technical jargon with which the layman is not familiar, but also their familiarity with the material leads to certain habits in language which do not cover the full spectrum of possible English constructions. Thus, in tailoring a system to the geologist, we have concentrated our effort (although not to the exclusion of all else) on those linguistic phenomena and problems which the geologists actually use. The non-geologist strays outside of this habitable region more easily than the geologist and this is the source of most of the problems in the above example.

There are other limitations of the system—even for a geologist user—which do not show up in the statistics of the demonstration. For example, although the parsing system contains experimental versions of some of the most sophisticated conjunction handling of any mechanical grammar to date[11] the handling of conjunctions in the system is still far from adequate. Although none of the questions during the demonstration involved complicated conjunction constructions, it is also true that no single person asked more than a few questions during the conference. I am confident that if a geologist really sat down to use the system to do some research he would very quickly find himself wanting to say things which are beyond the ability of the current system.

Linquistic fluency and completeness

There are two scales which can be used to measure the performance of a system such as LUNAR. We can call them *completeness* and *fluency*. A system is *complete* if there is a way to express any request which it is logically possible to answer from the data base. The scale of *fluency* measures the degree to which virtually any way of expressing a given request is acceptable. The two scales of completeness and fluency are independent in that it is possible to have a fluent system which will accept virtually any variations on the requests which it accepts, but which is nevertheless incomplete. Likewise, a system may be logically complete but very restricted in its syntax. A natural language system which is incomplete cannot answer certain questions, while a system that is not fluent is difficult to use.

Fluency of LUNAR

The LUNAR prototype is quite fluent in a few specific constructions. It will recognize a large number of variations on the request "give me all analyses of constituent x in phase y of sample z." It knows many variations of "give me" and many different variations on "analysis." However, there are other requests which (due to limitations in the current grammar) must be stated in a specific way in order for the grammar to parse them and there are others which are only understood by the semantic interpreter when they are stated in certain ways.

In the area of syntax, the LUNAR system is very competent. Most questions that fail to be understood have nevertheless parsed correctly, and questions having nothing at all to do with lunar geology parse equally well. The grammar knows about such things as embedded complement constructions, tense and modality, and other linguistic phenomena that probably do not occur in the lunar geology application.

Most of the limitations of fluency in the current system are due to the fact that the necessary semantic interpretation rules have not been put into the system. Continued development of the grammar and semantic rules would result in continued improvements in fluency, and there is no visible ceiling other than an economic one to the fluency which can be achieved. The following list of sentences gives a representative sample of the kinds of questions which the system can understand and the degree of variability permitted.

1. (List samples with Silicon)
   (Give me all lunar samples with Magnetite)
   (In which samples has Apatite been identified)
   (How many samples contain Titanium)
   (Which rocks contain Chromite and Ulvospinel)
   (Which rocks do not contain Chromite and Ulvospinel)
2. (What analyses of Olivine are there)
   (Analyses of Strontium in Plagioclase)
   (What are the Plag analyses for breccias)
   (Rare earth analyses for S10005)
   (I need all chemical analyses of lunar soil)
   (What is the composition of Ilmenite in rock 10017)
   (List the analyses of Aluminum in vugs)
   (Nickel content of opaques)
3. (Which samples are breccias)
   (What are the igneous rocks)
   (What types of sample are there)
   (What is the number of phases in each sample)
4. (Give me the K / Rb ratios for all lunar samples)
   (What is the specific activity of A126 in soil)
   (Give me all references on fayalitic Olivine)
   (Which rock is the oldest)
   (Which is the oldest rock)
5. (What is the average analysis of Ir in rock S10055)
   (What is the average age of the basalts)
   (What is the average Potassium / Rubidium ratio in basalts)
   (In which breccias is the average concentration of Titanium greater than 6 percent)
   (What is the average concentration of Tin in breccias)
   (What is the average concentration of Tin in each breccia)
   (What is the mean analysis of Iridium in type B rocks)
   (I want the average composition for glasses in dust)
6. (Modal Plag analyses for S10058)
   (Modal Olivine analyses)
   (Give me the modal Olivine analyses for S10022)
   (Give me all modal analyses of Plag in lunar fines)
   (List the modes for all low Rb rocks)
7. (Which samples have greater than 20 percent modal Plagioclase)
   (Which samples are more than 20 percent Plag)
   (How many rocks have greater than 50 ppm Nickel)
   (Which samples contain more than 15 ppm Barium in Plag)
   (How much Titanium does S10017 contain)
   (How much Nickel is in rock 10046)
   (How old is S10047)

Completeness of LUNAR

The criteria for logical completeness is a level of achievement that is not generally met by current data management systems with artificial request languages, much less by a system that recognizes natural language. The formal query language used for the retrieval component of LUNAR fares better than most data management systems in this respect since it is fundamentally an extension of the predicate calculus, but there are still some extensions which the language requires in order to fully achieve logical completeness.

More stringent than the incompleteness of the formal request language, there are limitations in the completeness of the subset of English handled by the system. This arises largely from the difficulties of parsing conjunction constructions, but there are also problems such as the ambiguity of the scopes of quantifiers. With some further work on conjunctions, the subset of English currently handled by the grammar should become a very convenient language to use. However, semantic interpretation techniques for this subset are far from complete and much further work is needed here.

CONCLUSION

What we have accomplished

The current LUNAR prototype represents a significant step in the direction of the goals of natural language understanding. Within the range of its data base, the system permits a scientist to ask questions and request computations in his own natural English in much the same form as they arise to him (or at least in much the same form that he would use to communicate them to another person). This is borne out by the performance of the system during the demonstration at the Second Lunar Science Conference. The system answered most of the questions dealing with its data base which were asked by the investigators during the demonstration. The effort required to cast the request into a form suitable for execution in the data base is assumed by the English processor, which translates the requests into compact "disposable" programs which are then executed in the data base. The English processor therefore functions as an automatic programmer which will convert the user's request into a tailor-made program for computing the answer. The English processor knows the ways in which geologists habitually refer to the elements, minerals, and measurements contained in its data base; it knows the specific details of the data base table layouts; and it knows the correspondence between the two. Thus, for example, the user need not know that the mineral Olivine is abbreviated OLIV in the data base, that the concentrations of Titanium are recorded in terms of the percentage of TiO2, that the class of rocks referred to variously as "type A," "high alkali," or "fine grained crystalline" are encoded as "TYPEAS" in the data base. These facts are "known" by the natural English processor, and the user's request is automatically translated from the form in which he may ask it into the proper form for the data base.

Where we stand

Although our current system does indeed exhibit many of the qualities that we have outlined as our goals, we are still far from achieving the goal as stated. The knowledge that the current system contains about the use of English and the corresponding meanings of words and phrases concerns those English constructions which pertain to the system's data base of chemical analysis data, which has a very limited and simple structure. Indeed this data base was chosen as an initial data base because its structure was simple and straightforward. In order to incorporate additional data bases into the system, it will be necessary to provide the system with information about the ways that the users will refer to those data bases in English, the vocabulary they will use, the ways they will use that vocabulary, and the "meanings" of the words and constructions in terms of the data base tables. For some tables (those whose structure is as simple and direct as the chemical analysis table), this process may be a direct extension of the current facility and may require only the addition of new semantic rules for interpreting the new words and constructions. For other applications, however, this will require much greater sophistication in both the linguistic processing and the underlying semantic representations and inference mechanisms. One type of data which will require considerable advancement is the representation and use of data which describes surface and structural features of the samples. This data does not fit conveniently into a table or a paradigm, and the techniques for storing it, indexing it, and providing access to it for retrieval and inference remain to be developed. Indeed, it is in the handling of non-tabular information of this sort that natural language querying may hold its greatest promise, but such potential is as yet undeveloped.

REFERENCES

1. Bobrow, D. G., Murphy, D. P., Teitelman, W., *The BBN-LISP System*, BBN Report 1677, Bolt Beranek and Newman Inc., Cambridge, Mass., April 1968.
2. Myer, T. R. and Barnaby, J. R., *TENEX Executive Language*, Bolt Beranek and Newman Inc., Cambridge, Mass., January 1971.
3. Ornstein, S. M., Heart, F. E., Crowther, W. R., Rising, H. K., Russell, S. B., Michel, A., "The Terminal IMP for the ARPA Computer Network," *AFIPS Conf. Proc.* Vol. 40, pp. 243-254, 1972.
4. Simmons, R. F., "Answering English Questions by Computer: A Survey," *Communications of the ACM*, vol. 8, No. 1, pp. 53-72, January 1965.
5. Watt, W. C., "Habitability," *American Documentation*, vol. 19, No. 3, July 1968.
6. Winograd, T., *Procedures as a Representation for Data in a Computer Program for Understanding Natural Language*, MAC TR-84, MIT Project MAC, February, 1971.
7. Woods, W. A., *Semantics for a Question-Answering System*, Ph.D. Thesis, Harvard University, Cambridge, Mass., August 1967.
8. Woods, W. A., "Procedural Semantics for a Question-Answering Machine," *AFIPS Conference Proceedings*, vol. 33, 1968.
9. Woods, W. A., *Augmented Transition Networks for Natural Language Analysis*, Harvard Computation Laboratory Report No. CS-1, Harvard University, Cambridge, Mass., December 1969.
10. Woods, W. A., "Transition Network Grammars for Natural Language Analysis," *Communications of the ACM*, vol. 13, pp. 591-602. October 1970.
11. Woods, W. A., *An Experimental Parsing System for Transition Network Grammars*, BBN Report 2362, Bolt Beranek and Newman Inc., Cambridge, Mass., May 1972.
12. Woods, W. A., Kaplan, R. M., and Nash-Webber, B., *The Lunar Sciences Natural Language Information System: Final Report*, BBN Report 2378, Bolt Beranek and Newman Inc., Cambridge, Mass., June 1972.