[11] Tony Leal

QUESTION ANSWERING BY KEYWORD SEARCH

Antonio Leal

26 FEBRUARY 1973

THIS RESEARCH WAS SUPPORTED BY THE ADVANCED RESEARCH AGENCY OF THE DEPARTMENT OF DEFENSE UNDER CONTRACT DAHC-15-73-C-0080.

SP-3680

Many natural language processors are based on algorithms that depend heavily on sentence structure (syntax) to determine meaning. Such processors usually employ a syntax scan followed by the use of semantic interpretation rules in the construction of a "deep structure". Such a structure leads to the representation of the sentence in a formal language which may be used to access information stored in a data base. The question-answering system ENQUIRE (English Question Interpretation and Response) uses a totally different approach to understanding sentences. The approach is semantically based rather than syntactically based. That is, the meaning of the sentence is derived from the meaning of the words in it and not from word order, syntactic features (noun, verb, etc.), phrase structure, or any other grammatical considerations. No formal language is necessary since the data base is an integral part of the sentence analyzer. Actually, very few words are recognized by the processor. These "keywords" are of two types. Data base terms are specific items and collections of items found in the actual data base. General English "function" words such as "what", "and", "or", "all", "not", "they", etc., supply information about the expected type of answer as well as logical and quantitative relationships between data base terms. All other unrecognizable words are completely ignored. The price of possible mis-interpretation is paid in return for a considerably wider range of understandable sentences due to practically free-form English within the context of the data base. Although important information is lost by disregarding grammar, ENQUIRE was purposefully designed to be as pure a semantic processor as possible.

ENQUIRE has been implemented at System Development Corporation by the author and uses a portion of the National Program Library and Control Program Inventory for the Social Sciences (NPL/CPIS) developed at the University of Wisconsin. This data base contains approximately 500 programs and 1100 properties which include languages, computers, manufacturers, etc. For example, the following are some sample questions that ENQUIRE can answer.

HOW MANY ANALYSIS PROGRAMS ARE THERE?
HOW MANY OF THEM HANDLE MATRICES?
WHAT LANGUAGES ARE AVAILABLE?

After deleting unrecognizable (undefined) words, the questions appear like this:

MANY PROGRAMS ANALYSIS
MANY THEM MATRICES
WHAT LANGUAGES

In the first question, the occurrence of the words "programs" and "analysis" causes a set of programs to be constructed that have the property "analysis" associated with them. The word "many" indicates that the user wishes to know only the number of programs in the set. The second question is similar except that the previously constructed set of answers is used when searching for the "matrices" property rather than the total set of programs. In the third example, it is assumed that the user wishes to see the subset of properties called "languages".

Notice that seemingly important words such as "are", "of", and "how" are not necessary to the understanding of the questions. Complete knowledge of the data base permits the use of huristics to "second-guess" the intent of the questions without knowledge of the grammar. If the data base is non-numerical and simple enough in structure, and if the user is informed about the domain of discourse, complicated questions that depend on syntax to determine meaning are not likely to be asked. This fact has been demonstrated by experimental use of the actual ENQUIRE system.

An ENQUIRE data base in general, must consist of a finite set of objects and a finite set of properties of those objects (see Figure 1). The properties

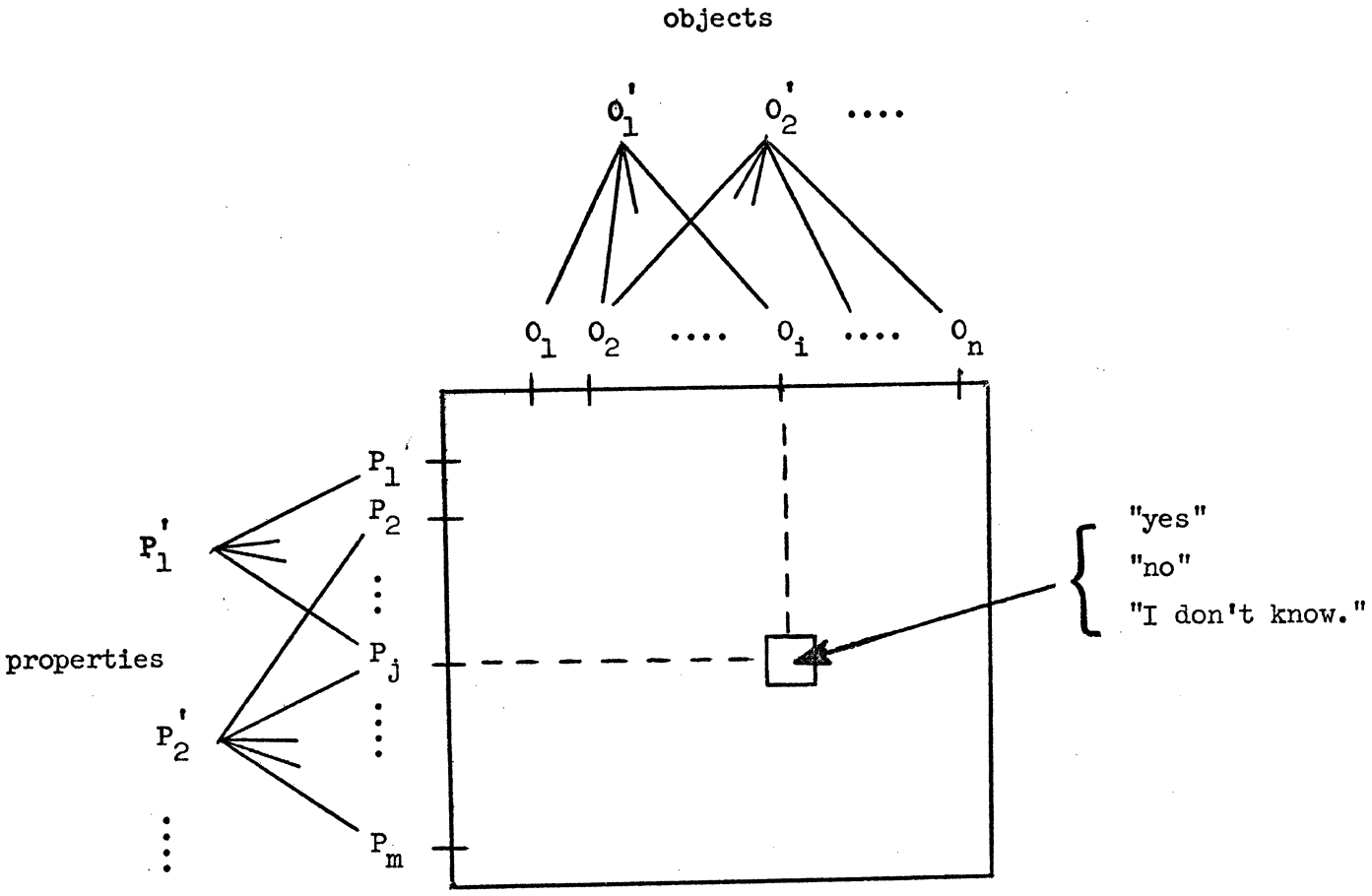

Figure 1. ENQUIRE Data Base Structure

may be thought of as functions that map each object into one of three possible values--"yes", "no", or "I don't know". If there is no possibility of an "I don't know", the data base is said to be "closed". Such is the case with the currently implemented program data base. The objects and properties may be divided into groups (subsets) that are not necessarily mutually exclusive. Each of the objects, properties, and groups must have a unique name.

In the program data base, the objects are names of specific programs such as "copy", "edit", "cluster", etc. The properties are keywords that describe the nature of the program: "analysis", "IBM-360", "FORTRAN", etc. Included as keywords are the various computers that the programs run on, as well as the languages they are written in and the program's manufacturer. Thus, three groups of properties exist: languages, computers, and manufacturers. In addition to the group names, the word "program" identifies the entire set of objects and the word "property" or "keyword" identifies the entire set of properties.

The function words that ENQUIRE recognizes are divided into five pragmatic categories: interrogatives, anaphoric reference words, logical words, quantifiers, and negatives.

Interrogatives determine the type of answer that is expected. The recognized interrogatives are "what", "which", "tell", "print", "give", "who", and "many". All of them except "many" are synonymous with "what". In a question such as

IS ALGOL A LANGUAGE?

where only "ALGOL" and "language" are recognized, it is assumed that a "yes/no" answer is expected due to the absence of an interrogative.

Once a set of objects is constructed as an answer to a question (whether or not the set is actually printed), it is kept and may be referred to in subsequent

sentences with anaphoric words: "they", "them", "their", "one", "ones", "it", "its", "these", "those", "except", "other", and "others". In this way, a series of questions may be asked about a common referent. For example:

HOW MANY CAI PROGRAMS ARE THERE?
WHAT ARE THEY?
ARE ANY OF THEM WRITTEN IN FORTRAN?
WHICH ONES RUN ON THE CDC/6600?

The words "they", "them", and "ones" in the examples above refer back to the list of CAI programs established in the first question. The answers to any immediately previous question are additionally saved and may be referred to by "these" or "those". If the last question above had been asked as follows:

WHICH OF THOSE RUN ON THE CDC/6600?

the word "those" would cause reference to the set of CAI programs written in FORTRAN that was established in the previous question and not to all CAI programs established in the first question. A new referent replaces the old one when either "these" or "those" is used or when a question is asked that contains no anaphora. If a question contains the names of specific programs as well as anaphora, the anaphoric words are ignored. For example:

EDIT IS WRITTEN IN FORTRAN, ISN'T IT?

Logical words determine whether unions or intersections are to be taken on constructed subsets of programs. For example, the answer to the following question:

HOW MANY PROGRAMS ARE WRITTEN IN FORTRAN OR PL/1?

would be the union of the FORTRAN programs with the PL/1 programs. The word "and" causes an intersection. "Neither/nor" is simply the intersection of the complements, i.e., the phrase:

NEITHER FORTRAN NOR PL/1

is the same as:

NOT FORTRAN AND NOT PL/1

If no logical words appear in the question, "and" is assumed. For example,

LIST THE IBM FREQUENCY ANALYSIS PROGRAMS.

Here, "IBM", "frequency", and "analysis" are distinct individual properties for which an intersection is desired.

Quantifiers ("all", "any", "some") aid in determining the type of expected answer in much the same way that interrogatives do. If no quantifier appears in the question, the first 20 answers are printed, after which the program asks the user if he wishes to see any more by typing:

MORE?

The user is expected to respond with either "yes" or "no". Use of the word "all" causes the entire list of answers to be printed regardless of its length.

The only word treated in a context-sensitive manner is "not". It is assumed that "not" will be used in conjunction with properties and that the exclusion of specific programs will be accomplished with "except" or "other" (see anaphora). For example:

HOW MANY PROGRAMS DO NOT SOLVE CUBIC EQUATIONS?
WHAT PROGRAMS OTHER THAN EDIT ARE WRITTEN IN BAL?

The following is a sample session with ENQUIRE. The program prints an asterisk when it is ready for a question.

```
*LIST THE PROPERTIES.
     ACCESS ACHIEVED ADDITIVF-MODEL ADJACENT ADJUSTED AGE AGGREGATED
AGREEMENT AIKEN ALGEBRAIC ALGOL ALGORITHM ALPHA ALPHABETIC
ALPHANUMERIC ALTERNATIVE AMPLITUDE ANALOGUE ANALYSIS ANGLE
     MORE?
*YES
     ANNUITY ANOVA ANSWER APPROXIMATE APPROXIMATION AREA ARGUMENT
ARITHMETIC ARRAY ARRIVAL ASCENDING ASSAY ASSEMBLER ASSOCIATION
ASSUMPTION ASYMMETRICAL ASYMPTOTICALLY-EFFICIENT
ASYMPTOTICAL ASYMPTOTIC AUGMENTED-MATRIX
     MORE?
*YES
     AUTOCORRELATION AUTOCOVARIANCE AUTO-SPECTRA AVERAGE AXIS BAL
BALANCED BALANCE-SHEET BANK BARTLETT BARTOS BAR-PLOT BASE-E BASE-10
BASE-2 BASHARIN BASIC BATCH BAUMANN BAYESIAN
     MORE?
*NO
*HOW MANY ANALYSIS PROGRAMS ARE THERE?
     106
*HOW MANY OF THEM HANDLE MATRICES?
     34
*WHICH OF THOSE ARE WRITTEN IN FORTRAN?
     CAP CLUSTER CORREL DATSIM FACTAN FSCORE IUFAC LAG MDSCAL MULCVR
SUBMTX TSSA
*WHAT ARE CLUSTER'S PROPERTIES?
     ADDITIVE-MODEL ANALYSIS CLUSTERING COEFFICIENT CORRELATION ETA
FACTORS FORTRAN GROUP GUTTMAN IBM360/40 IBM 360/67 MATRICES
MULTIDIMENSIONAL PHI PHIMAX RESULT SCALE SPACE
*ARE THERE ANY OTHER CLUSTERING PROGRAMS WRITTEN IN FORTRAN?
     YES.
*WHAT COMPUTERS DO THEY RUN ON?
     CAP: IBM360/40 IBM360/67
     MULTYP:  CDC6400 DEC-PDP10 UNIVAC1108
     OSIRIS-II-LEVEL-2: IBM360/40 IBM360/67
*IS COBOL A LANGUAGE?
     NO.
*WHAT LANGUAGES ARE AVAILABLE?
     ALGOL ASSEMBLER BAL BASIC PL/1 SNOBOL4 SPITBOL FORTRAN
FORTRAN-II FORTRAN-IV FORTRAN-V FORTRAN-63 LISP WATFIV WATFOR WATIV
*WHAT COMPUTERS SUPPORT ALGOL PROGRAMS?
     BURROUGHS5500
*DOES THE BURROUGHS5500 ACCEPT PROGRAMS NOT WRITTEN IN ALGOL?
     NO.
```

```
*GIVE ME SOME PROGRAMS WRITTEN IN PL/1 AS WELL AS FORTRAN-IV.
     THERE ARE NONE.
*ARE THERE ANY BAL PROGRAMS?
     YES.
*WHAT ARE THEY?
     COPY EDIT ERROR ITEM MATCH MERGE MERMAC QUEST RECODE SCORE
SELECT SEQUENCE SORT TOTAL
*HOW MANY PROPERTIES ARE ASSOCIATED WITH EACH ONE?
     COPY : 6
     EDIT : 7
     ERROR : 10
     ITEM : 21
     MATCH : 8
     CONTINUE?
*YES
     MERGE : 12
     MERMAC : 5
     QUEST : 18
     RECODE : 9
     SCORE : 16
     CONTINUE?
*YES
     SELECT : 26
     SEQUENCE : 12
     SORT : 8
     TOTAL : 19
*LIST ALL OF THE PROPERTIES OF THE BAL PROGRAMS.
     COPY : BAL DISK FILE IBM360 TAPE
     EDIT : BAL DISK DRUM FIELD IBM360 RECORD TAPE
     ERROR: BAL DATA ERRORS IBM360 ITEMS LISTING MULTIPLE-CHOICE
SCORING STUDENT TEST
     ITEM : ACHIEVED ALTERNATIVE BAL BISERIAL CORRELATION DATA
DISTRIBUTION FREQUENCY HISTOGRAM IBM360 INDIVIDUAL ITEMS PERCENTAGE
POINT PROPORTION RAW RESPONSE SCORING STANDARD STATISTIC TEST
     MATCH : BAL COMPARISON FIELD FILE GROUP IBM360 MATCHING RECORD
     CONTINUE?
*NO
*WHO ARE YOU?
     ENQUIRE
*WHAT LANGUAGE ARE YOU WRITTEN IN AND WHAT COMPUTER DO YOU RUN ON?
     LISP IBM370/145
*WHO WROTE YOU?
     LEAL
```

The keyword search approach that ENQUIRE uses is a very practical way of

accessing simply structured information in a non-numerical database. It takes no more than 3 to 5 seconds to answer a question as opposed to minutes that syntax analyzers may spend. The same approach that is used to interrogate the database may be used to store information directly in it. Once a user has become familiar with the types of questions that ENQUIRE can answer, he may concentrate more fully on the data he wishes to retrieve rather than on the precise way he must construct his question.