HAM-RPM: NATURAL DIALOGUES WITH AN ARTIFICIAL PARTNER

W. v. Hahn, W. Hoeppner, A. Jameson, W. Wahlster
Germanisches Seminar, University Hamburg, 2000 Hamburg 13, West Germany

ABSTRACT: This paper introduces the understanding system HAM-RPM, which simulates a dialogue partner conversing about a real-life domain. After outlining the system's overall structure, we discuss three of its distinguishing features: The first is its organisation of spatial data in a redundant multiple data base, inspired by certain aspects characteristic of visual search in humans. A new algorithm for noun-phrase generation is then sketched which is sensitive to the conversational state and uses a 'worst-case-first' strategy. Finally, we describe in some detail a specific operationalisation of the notion of the communicative relevance of objects. The paper concludes with a summary of the objectives of this research.

DESCRIPTIVE TERMS: natural language processing, dialogue simulation, focus of attention, reference semantics, visual search, FUZZY

1. System Overview

The AI system HAM-RPM [11] can be classified as a parsed, content-motivated natural language system [14]. It simulates a dialogue partner conversing in German [12]. At present we are testing the system on two different domains: the interior of a living room and a natural traffic scene. The former is a typical AI toy world, the latter a real-life domain with many challenging aspects [1], [2]. The conversational setting is as follows: the system is observing the scene through an artificial eye; the user, who cannot see the scene but who is familiar with the setting or has a photograph of it, asks the simulated dialogue partner about its current state.

The system is implemented in FUZZY, a LISP-embedded PLANNER-type language which provides a number of facilities for representing and manipulating fuzzy knowledge [8].

The heterogeneous knowledge base integrates various procedural and declarative representation languages, including production rules, semantic nets, pattern-operation rules and FUZZY antecedent theorems. Large parts of the knowledge are coded into a conceptual and a referential network, both of which are composed around a set of primitive structural and assertional links [16]. In the conceptual net facts are represented which are true in all possible states of the micro-world, whereas the referential net contains facts true in the current state. So that a user unaquainted with the program can alter the knowledge base, the basic system retrieves all of its specific knowledge from external files (Fig. 1) written in a self-explanatory notation.

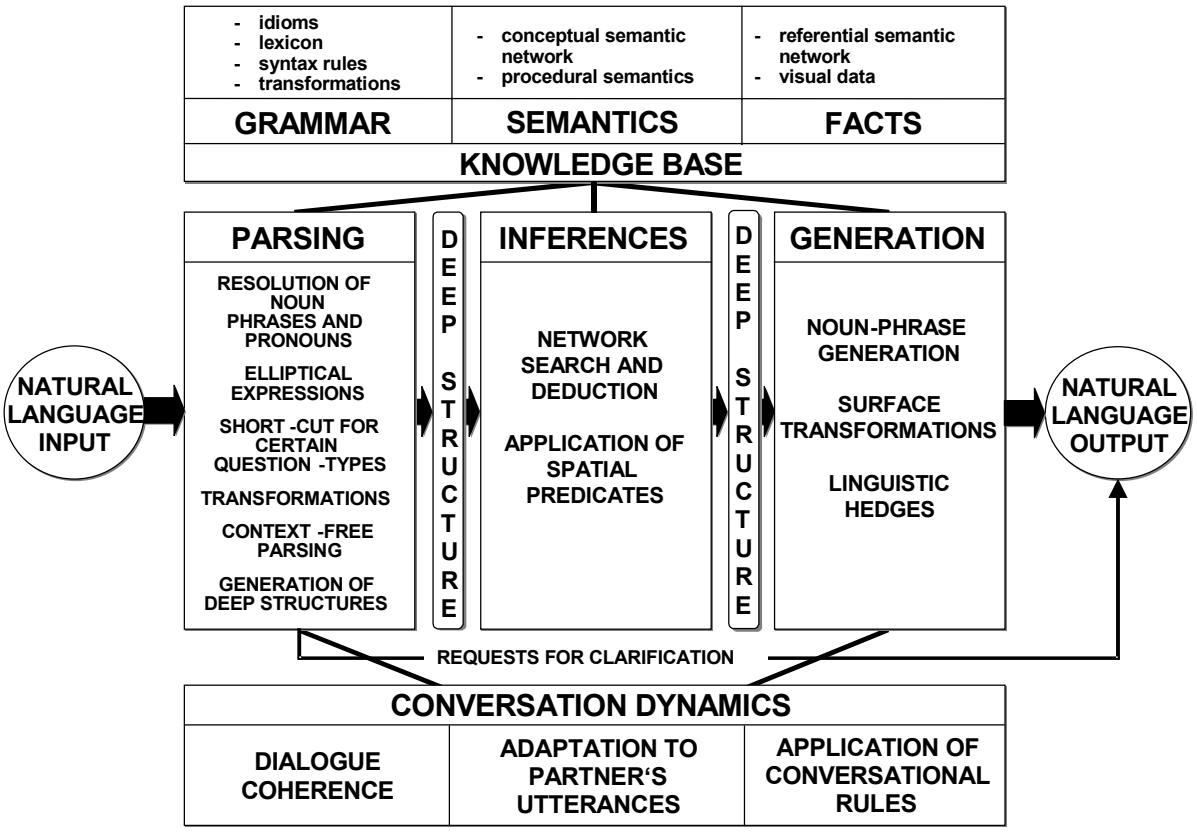

Fig. 1

During the processing of an input sentence interacting semantic, pragmatic and syntactic procedures are activated. However certain types of input, e. g. simple 'which'-questions, are processed without complete parsing by specialized and therefore very efficient inference procedures. Definite noun phrases are replaced by referential equivalents. The referents of pronouns are determined by consulting a special CONTEXT which contains a record of previously mentioned objects. The criteria applied are gender, number and recency of mention. The system can ask the user for any further information it requires at various points during the dialogue.

HAM-RPM handles elliptical expressions; this is of great importance in natural conversations, where the sentences used are by no means always complete. If the system's attempts to parse an input as a complete sentence fail, it backs up and starts an ellipsis-recognition process: the analysed elliptical expression is matched against the shallow structure of the previous sentence and expanded to a complete sentence. An improvement over the ellipsis facility of LIFER [7] is that for the construction of the deep structure of the expanded sentence full parsing is not performed; instead, particular constituents are projected onto the deep structure of the previous input.

As our emphasis, like that of the projects described in [3], [5], is on natural dialogues, the handling of fuzzy knowledge, including the generation of linguistic hedges, is of major importance for our system. A certain degree of communicative competence is embodied in the generation component of the system. Thus the form of the system output is determined in part by the current focus of attention [4].

2. The Handling of Spatial Relations

Spatial relationships are often mentioned in the dialogue situation described above, on the one hand because they form the topic of certain questions, e. g. *How many people are standing to the right of the old woman?*, and on the other hand because they are frequently used to help identify objects (see below, section 3).

Although the locations of our objects are represented simply by single points in a 2-D coordinate system, we have found it worthwhile to develop a relatively complex structuring of the visual data which makes it possible to model certain aspects of the visual search process in humans: While searching within a scene for an object which satisfies a given description, a human being can use the information available during a given fixation not only to examine an object located in the center of the visual field, but also to direct his eye movements toward potentially relevant peripheral objects [13]. This capacity, which in humans is based on low-level parallel processing, can be approximated through the use of a redundant multiple data base[1] [9]: For each point P in our coordinate system a separate CONTEXT, including P's coordinates in its name (e.g. 'C14-5') is established, which contains at any moment more or less the same information as would be available to a human being fixating the corresponding point in a photograph of the scene, i. e.

- detailed information about the object (if there is one) located exactly at point P; as well as
- a few facts about the objects located near P: their location and salience, plus other simple properties such as color, size, and object type.

Thus each object in the scene has detailed information on it entered in exactly one CONTEXT and less detailed data in a number of adjoining ones. The number of these depends on the presumed prominence of the object within the scene, and is computed by demons responsible for maintaining the correct distribution of such information among the CONTEXTs.

An example will illustrate the kind of search procedure which can take advantage of this way of structuring the visual data: To determine the referent of the noun phrase the woman standing to the left of the traffic light, assuming that the location of the traffic light has already

[1] The efficiency of this realization depends on the way in which multiple data bases are implemented; FUZZY's implementation is favorable in this respect.

been determined to be P, the system computes a list of the names of several of the CONTEXTs which lie in the area to the left of P, the CONTEXTs being close enough to each other so that it is likely that any object within this area will be mentioned in at least one of them. It then enters these CONTEXTs one at a time, within each looking for an entry indicating that a particular object is a person. When it finds one, it retrieves the object's location and switches to the corresponding CONTEXT to obtain more detailed information on it e. g. whether it is a woman (see below, section 4 for a more detailed example). Our reasons for adopting this structure of the visual data, despite its high cost in terms of memory, are three:

- It provides a framework within which hypotheses concerning certain types of visual search in humans can be formulated precisely. One of us is presently comparing such models with experimental data.
- It will permit us to explore the human-engineering hypothesis that man-machine communication can be improved through the assimilation of the internal processes of the machine to those of the man (see below, section 3).
- It permits tasks such as the one described above, which otherwise often involve long linear searches through sets of objects or points in space, to be performed by highly selective algorithms: the individual CONTEXTs are small and independently indexed, and only the relevant ones are searched. The resulting speed is vital for an on-line dialogue system.

3. Noun Phrase and Object

Two major problems of reference semantics confronting builders of understanding systems are noun-phrase (NP) resolution, i.e. the determination of the referent of a definite NP, and NP generation, i.e. the construction of NPs to identify objects uniquely. HAM-RPM's NP interpreter works on the shallow structure of the input sentence, which is generated through the replacement of multiple-word phrases and idioms by canonical expressions [10], a dictionary look-up and a simple morphological analysis. In a way reminiscent of SHRDLU [15], semantic/pragmatic processes like NP resolution are activated in HAM-RPM as soon as possible; this saves a great deal of unnecessary effort, since when existential presuppositions or selectional restrictions fail there is no further parsing, but rather feed-back to the conversational partner. The method we have developed for the inverse process, NP generation, is distinguished from earlier approaches (for an overview see [6]) by two aspects: The first is its use of a 'worst-case-first' strategy. Here is a sketch of the algorithm as applied to an object X:

1) If X has a proper name, this is used.

2) If X has no cohyponyms i. e. it has no siblings in the ISA-hierarchy the identifying NP is formed using the definite article and the type-node of the token-node.
3) If cohyponyms of X are found, it is first determined whether there are any properties of X which should not be mentioned in its description because they have already been presupposed in the question, e.g. *red* in the *question Which car is red?*; any such properties are excluded from further consideration. The algorithm then checks for the worst case, i. e. the existence of cohyponyms for which exactly the same properties are stored in the referential net as for X. If such an object Y is found, the system tries to distinguish the two in terms of spatial relations, in one of two ways: if X and Y are within a certain radius of each other, X is distinguished using an expression such as *the right hand* ... or *the second ... from the left*. Otherwise, the position of X relative to the objects in its vicinity is described. This process is recursive, because the system has to identify the objects which serve as spatial reference points for X (e.g. *the red car which is parked behind the tree which is to the left of the large building*). The system must then check whether the next-worst case obtains as well:
4) If the characterizing properties of X are a subset of those of a cohyponym Y, X is characterized in terms of the properties it lacks (e.g. *the red, but not old car*).
5) The system then looks among the properties of X for one which distinguishes it from its cohyponyms.
6) If there is no such property, all properties of X in the referential net are used, with no further attempt to find a minimal characterizing set.

The generator's second novel aspect is the kind of selectivity which is necessary in a world with a large number of objects, where

- usually there are many possible ways of uniquely identifying an object, and
- it becomes important to choose one which makes it easy for a human listener looking at the scene to locate the object.

Our generator in its present form does justice to these considerations in the spatial parts of its descriptions, since the system’s spatial search processes are biased in favor of objects as reference points which are visually easy to find starting from a given point, or which have recently been attended to (see below, section 4). By contrast, the decision as to whether to mention spatial relations or other properties in cases where both are possible (e.g. *the car in front of the big tree* vs. *the small brown parked car*) is still made without consideration of the ease of interpreting the resulting characterization. This is because the algorithm has not yet been adapted to provide for the retrieval of information on the object's non-spatial properties from he multiple data base described in section 2 above, in which the visual accessibility of a given property of a particular object is reflected in the number of CONTEXTs in which it appears.

4. The Search for Relevance

The specific criteria used to determine an object's relevance depend on the question and the context in which it is asked. One such set of criteria, appropriate within our dialogue situation to questions of the form *What's in front of the big tree?* is embodied in the FUZZY deduce procedure NEIGHBOURS, which we shall describe in order to illustrate three general principles governing the selection of relevant objects and the way in which their implementation can be facilitated through the use of certain characteristic mechanisms of the FUZZY language.

NEIGHBOURS (see Fig. 2) is a generator in the CONNIVER sense i.e. when it is called with an argument like (IN-FRONT-OF OBJECT13) it returns exactly one neighbour of OBJECT13, but it can be restarted to produce more.

```
1  (PROC NAME: NEIGHBOURS DEMON: TIMES-DEMON (?RELATION ?OBJECT1)
2    (GOAL (LOCATION !OBJECT1 ?LOCATION1))
3          (FOR ?CONTEXT &(ADJOINING-CONTEXTS !RELATION !LOCATION1)
4               (CONTEXT !CONTEXT)
5               (FOR FETCH: (SALIENT ?NEIGHBOUR)
6                           (GOAL (!RELATION !NEIGHBOUR !OBJECT1))
7                           (GOAL (NEAR !NEIGHBOUR !OBJECT1))
8                           (Z + 1 (RECENTLY-MENTIONED !NEIGHBOUR))
9                           (SUCCEED? !NEIGHBOUR ZACCUM))))
```

Fig. 2

Principle 1 The search for relevant objects should be so designed that the most relevant ones will tend to be found early. This is accomplished in NEIGHBOURS in the two iteration loops in lines 3 through 9. Lines 3 and 4 provide for the CONTEXTs 'in front of ' OBJECT13 to be searched starting with the nearest and moving away (the LISP procedure ADJOINING-CONTEXTS returns an appropriately ordered list of CONTEXTs). A further, more general technique for ensuring that the most relevant objects will be found early is applied in line 5, in which a basic feature of the FUZZY language is exploited: Each entry in the associative data base has associated with it a 'z-value', e.g. ((SALIENT OBJECT1) .4). The system automatically keeps the entries ordered according to z-value, so that an iterated FETCH generates entries in monotonic order of z-value. Here, this means that, within each CONTEXT, the objects are retrieved for examination in decreasing order of salience.

Principle 2 The relevance of an object is often a function of several factors which can be combined numerically. The four factors in our example appear in lines 5 through 8: the visual

salience of the neighbour; the degree to which it is truly in front of OBJECT13 (i.e. as opposed to being merely 'more or less in front of' it); and the extent to which it has been mentioned recently.[1] The FUZZY consequent theorems called in lines 6 and 7 (by pattern) and in line 8 (by name) return, like the FETCH of line 5, both a skeleton and a z-value representing how high the neighbour rates according to the criterion in question. These z-values needn't be explicitly manipulated, as they are automatically multiplied together by the 'procedure demon' called TIMESDEMON specified in line 1. Their product, the value of ZACCUM, which can be interpreted as the total relevance of the neighbour in question, is returned in line 9 as the z-value associated with it.

The function Z in line 8 permits the z-value to be changed before it is passed to the demon; in this case it is increased by 1 so that objects not recently mentioned will not be judged entirely irrelevant.

Principle 3 There must be a rule for determining how exhaustive the search for relevant objects should be and how many of them should be mentioned. In our example, the rule has the form 'Find six objects with a relevance above the threshold of .2, and mention the three most relevant of these', where the numbers 3, 6 and .2 are values of parameters which depend on the course of the dialogue, for example the number of times the human questioner has expressed a desire for a detailed answer.

The FUZZY iteration in Fig. 3 causes those objects generated by NEIGHBOURS with a z-value above .2 to be stored temporarily in a separate CONTEXT called

```
(FOR TRY:  (NEIGHBOURS) (IN-FRONT-OF OBJECT13) ZVAL: .2
           (CONTEXT (QUOTE STM))
           (ADD  &(VAL)  (ZVAL))
                            ...)
```

Fig. 3

STM; when this CONTEXT has collected six elements, three FETCHes will automatically retrieve the three most relevant, because of the ordering in terms of z-value.

5. Current Status of Implementation

A non-compiled version of HAM-RPM, implemented in Fuzzy, is currently running on the PDP-10 of the Fachbereich Informatik in Hamburg under the TOPS10 operating system. Incorporating all of the features described in this paper, it occupies 55k of core and requires

[1] All references to objects during the dialogue are recorded in a separate CONTEXT 'references', which is also consulted during pronoun resolution.

from 1 to 15 seconds of CPU time to generate a response. The program is user engineered, robust, and debugged enough so that it can be run by naive users.

6. Goals and Methodological Principles

HAM-RPM's character as a system for the investigation of dialogue structures and the associated cognitive processes implies certain methodological principles:

1. the principle of consistent dialogue simulation

Communication with the system will consist exclusively of utterances appropriate to a natural dialogue between two human partners, even when it is meta-communicative in nature.

2. the principle of the pragmatic/semantic foundation

The emphasis will be on the pragmatic/semantic aspects of non-technical conversations. By this we mean not so much the superficial linguistic features as the type of speech-act sequences which are involved in this kind of dialogue.

3. the principle of complete implementation

A complete running version of the system will be maintained at all times, so that changes and extensions can be evaluated in relation to the model as a whole.

4. the principle of domain-independence

The system's domain-independence will be ensured by a clear distinction between the basic program and all representations of the knowledge specific to a particular type of scene.

Two goals on which we shall concentrate in the near future are:

- the improvement of the meta-communicative capacity of the system – while adhering to the methodological principle of consistent dialogue simulation – with a view to the extension of its repertoire of communicative strategies.
- the development of a systematic representation of time, so that e.g. the movements of objects in a sequence of pictures can be described.

Acknowledgement

This research is currently being supported by the Deutsche Forschungsgemeinschaft.

References

[1] Badler, N.I. (1975): Temporal Scene Analysis: Conceptual Descriptions of Object Movements. Technical Report No. 80, Dept. of Computer Science, University of Toronto

[2] Bajcsy, R. / Joshi, A.K. (1977): Partially Ordered World Model and Natural Outdoor Scenes. Report UP-MS-CIS-77-63 (presented at Workshop on Computer Vision, Univ. of Mass., Amherst)

[3] Bobrow, D.G. / Kaplan, R.M. / Kay, M. / Norman, D.A. / Thompson, H. / Winograd, T. (1977): GUS: A Frame-Driven Dialog System. *Artificial Intelligence, 2, April 1977, 155-173*

[4] Grosz, B.J. (1977) : The Representation and Use of Focus in a System for Understanding Dialogs. *Proceedings of the 5th IJCAI, Cambridge, Mass., 67-76*

[5] Hayes, P.J. / Rosner, M.A. (1976): ULLY: A Program for Handling Conversations. *Proceedings of the AISB Summer Conference, Edinburgh, 137-147*

[6] Heidorn, G.E. (1977): Generating Noun Phrases to Identify Nodes in a Semantic Network. *Proceedings of the 5th IJCAI, Cambridge, Mass., 143*

[7] Hendrix, G.G. (1977): The LIFER Manual. A Guide to Building Practical Natural Language Interfaces. Technical Note 138, Stanford Research Institute.

[8] LeFaivre, R.A. (1974): Fuzzy Problem Solving. Technical Report No. 37, Univ. of Wisconsin, Madison

[9] McDermott, D.V. (1974): The CONNIVER Reference Manual. MIT-AI-Memo No. 259a

[10] Parkison, R.C. / Colby, K.M. / Faught, W.S. (1977): Conversational Language Comprehension Using Integrated Pattern-Matching and Parsing. *Artificial Intelligence, 9, 111-134*

[11] Wahlster, W. / v. Hahn, W. (1976) : Einige Erweiterungen des natürlichsprachlichen AI-Systems HAM-RPM. *Laubsch, J.H. / Schneider, H. -J. (eds.): Dialoge in natürlicher Sprache und Darstellung von Wissen. Freudenstadt, 204-225*

[12] Waltz, D.L. (ed.) (1977): Natural Language Interfaces. *Sigart Newsletter, No. 61, Febr. 1977, 16-65*

[13] Williams, L.G. (1967) : The Effects of Target Specification on Objects Fixated During Visual Search. Sanders, A.F. (ed.): *Attention and Performance. Amsterdam: North-Holland, 355-365*

[14] Wilks, Y. (1974): Natural Language Understanding Systems within the AI Paradigm: A Survey and Some Comparisons. Memo Nr. 237, AI-Lab., Stanford Univ.

[15] Winograd, T. (1972): Understanding Natural Language. N. Y.: Academic

[16] Woods, W.A. (1975): What's in a Link? Foundations for Semantic Networks. Bobrow, D.G. / Collins, A. (eds.) : *Representation and Understanding. N. Y. : Academic, 35-82*