Discourse Anaphora and Anaphora Resolution in a Natural Language Interface to a Database Question Answering system

Nor Aliah bt. Mohd Zahri and
Dr. Stephen Sommerville,
Department of Computer Science
Queen Mary and Westfield College
University of London
LONDON E1 4NS
<aliah, steves@dcs.qmw.ac.uk>

0. Abstract:

Allowing users to interact with a database system in a continuous question and answering dialogue provides a conversational form of interaction between the user and the system. Such facilities give the user the opportunity to use anaphoric expressions, substitution phrases and ellipses. We examine the different forms of anaphora resolution required for a Natural Language Interface (NLI) to database question-answering systems which can support cohesive and coherent dialogue interaction. Our approach adopts a minimalist methodology in that it seeks to achieve resolutions of typical forms of anaphora and ellipsis in question-answering dialogues, without over committing the interface to computationally costly processing. This methodology suggests a natural separation of roles in which modules of the interface software function as agents co-operating to resolve syntactic, semantic and discourse features of anaphoric expressions to recognise and select candidate referents from antecedent expressions.

We investigate this architecture in the context of typical instances of anaphora and ellipsis, review alternative proposals for anaphora resolution from the computational linguistic and discourse theoretic literature and characterise the prototype test-bed we have used to evaluate our proposals. We conclude with some projections of future developments of this approach, together with current limitations of our resolution techniques.

1.0: Introduction.

Anaphora is the use of a word or phrase (anaphor) which refers to an item indirectly, by virtue of its co-reference with another word or phrase used earlier in a text or dialogue. There are several different types of anaphora, but a broad distinction can be drawn between intra-sentential (when co-reference is with a phrase occurring in the same sentence) and inter-sentential (when co-reference is with some phrase occurring antecedently in the discourse). It is an odd fact of the literature that, by and large, intra-sentential anaphora has been of chief concern to theoretical linguists representing the syntax and semantics of language; whilst inter-sentential (or "discourse") anaphora has primarily occupied the attentions of psycholinguists, concerned with the cognitive and behavioural features of discursive use -- examining how context and dialogic conventions contribute to recognition of speaker's topic and focus through a discourse.

There is little reason, though, to suggest, however, that these two forms of linguistic "cross-reference" cannot be subsumed, for computational purposes, under the same mechanisms for recognising and interpreting anaphoric expressions. This, at least, is the approach adopted here. The task for interactive systems which incorporate "natural language understanding" is to provide for the fluency and coherence of natural discourse by detecting and, when possible, supplying interpretations for the ordinary language users' frequent recourse to anaphoric expressions to focus and re-focus their utterances and to avoid stilted and repetitious sentences. This task reduces to devising **recognition procedures** (RPs) and **assignment procedures** (APs) which access stored information to supply candidate referents for each anaphor in a users' input sentences.

Our proposal, here, is to address some of the difficulties in specifying and implementing the role of a "Discourse Manager" whose primary functionality is to apply RPs and APs to supply candidate identifiers which resolve each anaphoric expression in a user's discourse. In this paper, we focus primarily upon Pronominal and Definite NP anaphors (as indicated by the shaded area, above), but our approach can be extended to incorporate other varieties of anaphora.

2.0: Forms of Anaphora:

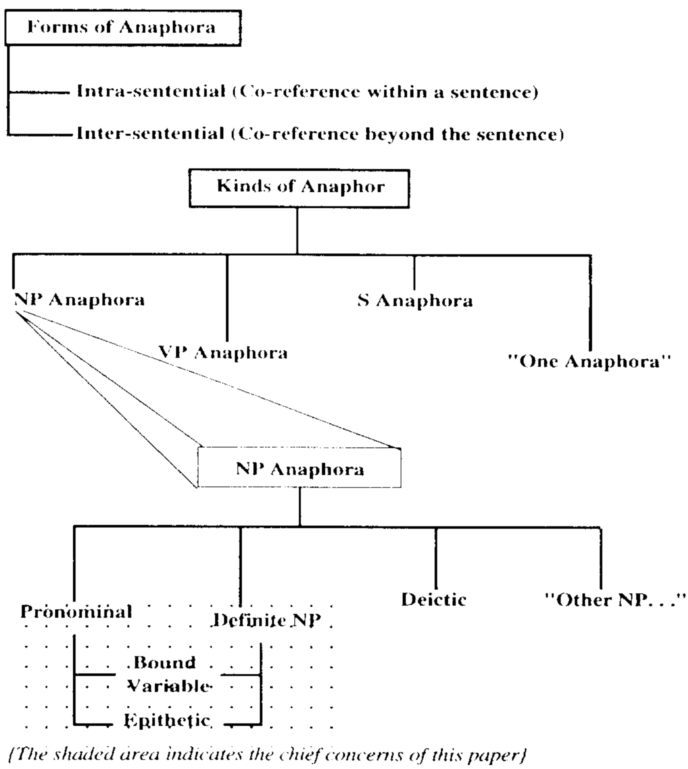

Figure 1: A Taxonomy of Anaphora Types

Figure 1 provides a summary taxonomy of the forms of anaphora which we hypothesise will occur most typically in question-answering dialogue. Our focus in the following will be upon particular kinds of NP-anaphora -- Pronominal and Definite NP-anaphors, though there is little reason to suppose that our techniques cannot be extended naturally to the other forms of NP phrase types occurring anaphorically. To clarify the domain, we include here are some examples of each:

- Pronominal anaphors include all forms of pronoun such as he, she, they, it, you, them, us him...
  Pronominal anaphora can occur both intra- and inter-sententially: viz.
  (1) John loaned a book to Mary, which she lost.
  (Intra-sentential)
  (2) John loaned a book to Mary. She lost it on the way home.
  (Inter-sentential)
- Definite NP anaphors start with a definite article e.g. 'the'
  (3) John loaned a book to Mary. The book was too expensive to buy.
- Both Pronominal and Definite NP anaphora occur in two forms:
  Bound Variable and Epithetic.
  (4) Every house in the street needed the roof mending.
  (where "the roof" is bounded by the quantifier "Every").
  (5) Ross used his bank card so much that the poor guy had to declare bankruptcy.
  (where the epithet "the poor guy" co-refers with "Ross").

For comparison we illustrate other forms of anaphoric expression in the above taxonomy, though these are not discussed in this paper:

- VP- and S- anaphora: For example in a sequence of Question-Answer interactions:
  (6) A: Have ten printers arrived from the supplier
  B: They all did, except one of them
  A: Will the printer arrive tomorrow

  where "did" in B's response co-refers anaphorically with "arrived from the supplier" and where "the printer" mentioned by the third utterance, (definite NP) points to 'one of them' (one of the ten printers). Here, "one of them" is another kind of anaphor classified as One-anaphora.
- Sentential anaphora can be illustrated by this example:
  (7) Mary owes 100 yen to John. It is a long-standing debt.

In order for an anaphor to be interpreted, it has to be matched up with the appropriate pre-existing discourse referent. This is called RESOLVING the anaphoric reference, and is achieved by identifying the ANTECEDENT of the anaphor i.e. the previously mentioned item to which the anaphor refers. Resolution can itself be analysed as an (at least) two-stage process: i) matching the anaphor to its antecedent, and ii) determining (or retrieving the co-referential value of the anaphor. Anaphors must stand for the same referent, not merely for the same words as the antecedent. For example, in the sentence

(8) "Mary liked a red dress and bought it"

"It" here must co-refers with whatever "a red dress" refers to. If "it" were merely a substitute for the words "a red dress" then "Mary liked a red dress and bought a red dress" would be equivalent to (1). Yet the two red dresses need not be the same .

3.0: Previous Approaches

There have been many approaches to anaphoric resolution (but none which provide a general solution, applicable in any environment). Researchers have illustrated resolution in a task oriented dialogue [1], text processing system [2], story-telling system [3] and information-retrieval system [4].

Various approaches have been recommended by researchers to resolve the referent of an anaphor. Bobrow [5], back in 1964, used a very superficial lexical matching with a little added syntax. Woods and others [6] in the LUNAR system worked on both syntactic and semantic levels, though as in Bobrow's system, LUNAR suffers from the same basic limitation, that it can resolve an anaphor only where the antecedent is of similar structure. Tufis and Cristea [7], in their IURES, system resolve anaphora by matching the syntactic structure of the current question to that of the previous question. Approaches based on 'case-frame' matching have been popular, with an emphasis on unification techniques for matching features (cases). Case frame matching requires that the case frame unifies semantically -- an approach demonstrated by Templeton and Burger [8]. Trost and others [9], in their DATENBANK-DIALOG, took a slightly different approach, though both applied the basic 'case-frame' matching concept. Trost classified anaphora into "local" and "global". Resolution of "local" anaphora is frame matching of the anaphor, while for the "global", the antecedent is determined from a discourse stack. Jerry Hobbs [10] has proposed a "parse tree searching algorithm for syntactic matching of antecedents. Other approaches include preference semantics [11], which stratify the features needed for resolution.

Rich [12] adopted a different model, whereby the resolution component consists of several modules which interact with each other to propose a candidate antecedents. The model proposes a candidate antecedent and gives a weighting of confidence levels for the antecedent. It can also act as a filter in the process of eliminating or narrowing the number of candidates for the antecedent. This has some affinities to our own approach, as will become clearer, later. Carter's [3] approach combines several existing techniques, using Sidner's theory of local focusing [13] and Wilks preference semantics with common sense inference. She adopted a "shallow processing" approach, whereby she tries to minimise reliance on world (domain) knowledge. This approach also has affinities with our proposed *minimalist* approach to resolution for Question Answering (QA) systems.

Our work looks at anaphoric resolution specifically in the context of natural language interfaces to a database QA system, where we propose a discourse-theoretic model for recognition and assignment of candidates for resolving anaphors co-referring with NPs from antecedent input questions. In this model, our hypothesis is that the syntax and semantics of users' input sentences are themselves insufficient to resolve anaphors. Discourse information, such as focus, topicalisation and cohesion, all contribute to anaphora resolution.

4.0: Kinds of information for Anaphora Resolution

In order to attain a plausible interpretation of input sentences, a Natural Language Interface system must draw upon and integrate information from many sources: lexical, syntactic, semantic, discourse and pragmatic modules. How that information is stored, retrieved and used will have a significant impact upon the efficiency, reusability and cohesiveness of the interface. Designers and researchers into NLI systems have adopted various approaches to the structural components of interface designs. In many early natural language interfaces, discourse and pragmatic interpretation had only a minimal role -- in part because of the high computational costs of storing and searching representations of previous query components. More recently, systems have increased the role of pragmatic interpretation especially in order to provide resolution of anaphoric and elliptical input expressions and to ensure that dialogue with such systems is cohesive and focused. In such systems, a crucial design issue remains one of deciding the extent and composition of stored *meta-contextual* information: how much of the intra-linguistic and extra-linguistic information about the discourse and the domain should be retained, and how best to put it to use in translating the current user question into an executable query. This question is particularly crucial for the role of a Discourse Manager in anaphora resolution; since its success will depend largely upon both how much information is retained from previous discourse, and upon the forms in which that information is retained.

The simplest model of processing a user's natural language input question -- incremental processing -- passes the input sentence, sequentially, through each of (say) four modules: morphological/lexical recognition and analysis, syntactic parsing, semantic interpretation and pragmatic contextualisation. The output from one module serves as input to the next, and pragmatic interpretation is deferred until late in the processing cycle. This approach is recommended by the fact that domain specific (contextual) information is only called upon once in the sequence of stages (in the final pragmatic phase), and remains effectively separate from the language specific modules (morphology, lexicon, grammar and semantics). Its chief shortcoming is inefficiency. If, for example, a user's question will eventually fail, because there is no match between some referring term (noun phrase) in the input sentence and some domain specific identifier, considerable processing resources will be wasted before this fact is detected at the pragmatic phase.

An alternative model enables stored pragmatic information (intra-linguistic: relating to preceding sentences, and extra-linguistic: relating to database domain) to contribute to translation of the user's current sentence at any phase of processing. In this model, each of the above four processing modules is regarded as a (semi-) autonomous agent (or "Knowledge Source") which co-operates with other agents to supply a translation of the input sentence. For example, early in the processing of the user's input, if the lexical module detects the presence of a pronoun, the reference of which may not be determinable from the semantics of the current sentence, alone, then stored information about referents of noun phrases from preceding queries may be called upon to provide candidate referents for the pronoun. In this sense, pragmatic information may be used at any stage in the processing of the user's input.

Such a model resembles the architecture of some knowledge-based problem-solving systems. In the "expert systems" of Hayes-Roth's HEARSAY-II and Craig's CASSANDRA [14], the processing of current input is shared between a number of knowledge sources who "read from" and "write to" a shared data-structure -- a global blackboard. Agents may process input sequentially, each completing their task before the next agent begins processing, or they may intervene to contribute results to the blackboard at different times, relative to the updated state of the input on the "blackboard". Viewed thus, simple, non-discourse related input questions may be translated into query form in the familiar, incremental manner of traditional Natural Language Processing systems. Where discourse and context specific information can provide information essential to input translation, however, suitable intervention to schedule retrieval of information from previous questions, or from database specific knowledge sources, can reduce the time each module takes to complete its contribution. Typically, such non-sequential processing of input sentences will occur when translation depends upon contextual and discourse information from preceding questions in the user-system dialogue.

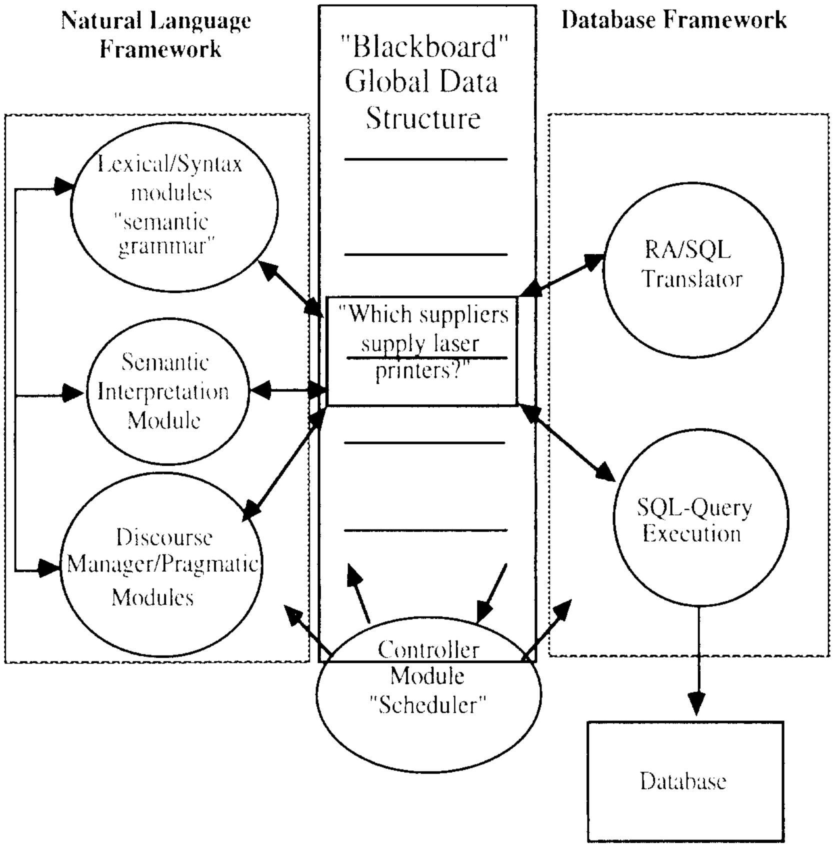

Figure 2: Blackboard Architecture for discourse-based anaphora resolution

In our Discourse-based model (fig. 2), we propose that in order to provide a minimalist approach to the resolution process, it is sufficient to retain information about the preceding sentences parse structures (with embedded semantic features), together with information about the focus elements. The model proposes that, from a given input question, various types of focus element can be derived. In the normal case, the Discourse Manager makes available information on a focused item, which is assigned to the Active Discourse Focus. Other constituents acts as a potential focus element, i.e. items to which subsequent user questions may refer anaphorically, which are assigned to the Potential Discourse Focus. When an anaphor in a current sentence is resolved, the Active Discourse Focus may or may not be changed. If the current focus changes (for example if a new subject Noun Phrase is used), the change will move the old Active Discourse Focus into the Dormant Discourse Focus (DDF), and the new focus element will be assigned to the Active Discourse Focus. Any Potential Discourse Focus, which does not contribute as a candidate antecedent for an anaphor, is replaced by a new set of potential discourse focus elements derived from the current question. This strategy for resolving anaphors makes use of the principle (called a "Relevance Principle" by Sperber and Wilson) advocated by Ruth Kempson, whereby ". . . implicit in the concept of maximal relevance [to resolving anaphors] is a trade-off between maximum amount of information (contextual implications) for minimal processing costs." ([15], p. 214)

5.0: Processing approaches

In a NLI which incorporates anaphoric resolution, there are a number of ways in which different tasks can be assigned to the separate (lexical, syntactic, semantic and pragmatic) components of the interface. One ("Early Evaluation") assigns anaphoric resolution chiefly to lexical and syntactic interpretation, whereby any occurrence of an expression of a particular category (say, a pronoun) is considered to be anaphoric. In the event that subsequent resolution fails to provide the correct referent, the expression is immediately considered to be non-anaphoric. This approach which evaluates an expression at an early stage is efficient in recognising expressions as anaphoric, yet, given the meagre information available from lexical and syntactic processing alone, it is less adequate in uniquely assigning an interpretation. In other words, it is highly vulnerable to lexical ambiguities, and can fail for expressions whose anaphoric behaviour is not determinable from their lexical or syntactic category (for example, definite Noun Phrases: "the car", "the red dress").

An alternative ("Late Evaluation") is to allocate the task of anaphoric resolution to semantic interpretation, perhaps giving the discourse/pragmatic module only a minor role. When all information required to identify and resolve an anaphor has been gathered, only then will the resolution process be carry out. This approach assumes initially that no occurrence of an expression is anaphoric. The system will process the whole input sentence first. When this process fails, it will consider whether any constituent expression is anaphoric. This cautious approach delays the resolution process of a genuine anaphoric expression. Indeed, such an approach seems to rest on an assumption that anaphora occurs only rarely in ordinary "Question-and-Answer" dialogue -- an assumption which is surely contestable.

A third possibility ("Interleaved"): the discourse/pragmatic module may take primary responsibility for anaphoric resolution, with minor contributions from semantic and syntactic components. In this option, the resolution process requires the Discourse Manager to take an active role. Here the semantic interpretation task is merely to provide lexical information about anaphors in the input query. Upon detection of an anaphor, the Discourse Manager is activated early in processing the input sentence. It requires that the Discourse Manager recognise and provide sufficient information for resolution. As Kempson [15] points out, anaphoric resolution should use discourse information to provide a cohesive relation between anaphor and antecedent. Semantic dependency is not linguistically sufficient to provide for cohesion in discourse relations.

It would be too computationally expensive for the Discourse Manager to store in advance all the information that might be needed to resolve anaphors (as in the late evaluation strategy). On the other hand, it should retain the minimum information needed to resolve the most frequently occurring cases of recognisable anaphora (as in the early evaluation strategy). It requires some balance between the two to avoid the retention of too large a store of information for resolution, and to avoid large processing commitments early in the interpretation of input sentences.

We propose an interleaved method of processing anaphoric expressions, which activates the Discourse Manager as and when necessary to search for candidate referents. In an interleaved approach, the Discourse Manager has responsibility for supplying the information necessary for resolution, depending on the types of anaphor involved. It evaluates the information to hand, and what more is required. In the initial stage, only minimal information is made available for resolution. If the resolution does not provide a suitable antecedent referent, then it is the role of the Discourse Manager to retrieve more information to reduce the number of candidate resolvers or to propose a candidate.

6.0: Minimalist Approach -- problems in anaphora resolution

The forms of anaphora used in a QA system are typically fewer than the forms which appear in ordinary conversational discourse. It is thus a practical issue of balancing computational efficiency against completeness in proposing a strategy for anaphora resolution in such a system. It is not appropriate to use too rich an algorithm to solve esoteric problems of reference determination. What is required is a solution which is feasible for an interface to a database management system. It should also be a solution which enhances the users' ability to decompose complex queries into sequences of simpler questions and one which removes some of the cognitive overhead of having to repeat frequently used referential phrases. By constraining the algorithm to resolve some but not all of the wide range of anaphoric types, we can have a practical and feasible solution which supports QA dialogue.

How do we identify the most central types of anaphor that would occur most often in a database question-answering system? Among the frequent types of anaphor used in a database Question Answering systems are pronominal expressions and definite noun phrase anaphors. The use of substitution phrase such as "one" or "ones" and "one of them" too is common in a database dialogue. Other types of substitution phrase such as verbal substitution "do" and clausal substitution "so" and "also" seem to appear less frequently in a database question-answer sessions. These surmises, however, are based on limited samples of simulated QA dialogues, and await more extensive sampling and trials for their proper confirmation. At present, our approach has focused primarily upon Pronominal and Definite Noun Phrase Anaphora -- though, in principle, we believe it can be extended to more esoteric forms of anaphoric reference.

The minimalist methodology aims at a "minimal" coverage of varieties of anaphoric expression and a "minimalist" approach to identifying the referent of these expressions. Our approach constrains the use of anaphoric expression to pronominal, definite noun phrase, substitution phrase and certain forms of ellipsis. A "minimalist" approach to resolving the referent requires minimal usage and maintenance of discourse and pragmatic information. It also recognises the inherent limitations of current Natural Language Understanding systems in respect of the amount and kind of "World (domain) Knowledge" which the system can store and use. Given the frequency with which users may employ anaphora in "Question-and-Answer" dialogue, it is sometimes strategic to fall back upon the user's own linguistic ability to re-identify an unresolved anaphor, rather than delay interpretation through extensive semantic and pragmatic searches.

7.0: Prototype testbed for evaluation of the architecture.

To test our hypotheses, a prototype testbed has been implemented. The testbed, which is based on the architecture described earlier (fig. 2), uses test cases to evaluate our hypotheses. The computational model -- Discourse Model-- is implemented in Sicsctus Prolog, C programs and SQL. It runs on a Sun-4 in real-time. The grammar used is a feature-based grammar. It is an embedded PATR-like language, which looks like PATR as described by Gazdar and Mellish [16]. The parser is a left-corner bottom-up recogniser that presupposes the existence of a set of rules written in a PATR-like notation. This left-corner parser is due to Pereira and Sheiber [17]. Questions from the user are translated into SQL, through an intermediate Meaning Representation Language -- in this case "Relational Algebra". The SQL queries retrieve the required information from an Ingres database. Test cases provide positive evidence for the hypothesis that commonest varieties of pronominal and definite noun-phrase anaphora can be resolved through interleaved recognition and assignment of candidate topic and focus elements from previous user questions.

8.0: Summary and Conclusion

There is no doubt that pattern matching and case-frame based approaches, perhaps with the addition of options for users to choose candidate resolvers, could generate a flexible solution to the otherwise intractable problem of anaphoric resolution in a Natural Language Interface. It is applicable, however, only to a restricted range of input sentence types. They may be suitable for small, domain-dependent systems, but not for generic problem solving interfaces. If a Discourse Manager takes a major role in resolution, it creates the opportunity for extension into more cohesive interfaces -- interfaces which provide for a more "conversational style" of interaction -- not merely in small domain-dependent systems, but in problem solving and knowledge-based systems.

It may also make the separation of Natural Language Interfaces from database-specific software more perspicuous, thereby assisting in the implementation of interfaces which may be more transportable to other domains. Transportability to other Relational Database Systems can be achieved by using SQL, since SQL is a de-facto standard query language.

Lastly, in proposing an architecture which draws upon existing "knowledge-based systems" technology, with the interaction of a number of different knowledge agents, there may be some doubt as to the efficiency and speed as a contributory factor to the usability of such a system. Progress, however in increasing the speed of computer systems is ever advancing, so that many of the features that we foresee as advantageous in this work, could in future be improved by access to high speed performance computing.

Nonetheless, much remains to be tested within this prototype system: thus, far we have made claims only for its utility in relation to resolution of pronominal and definite NP anaphors. We will report elsewhere (forthcoming) on current efforts to extend this approach to other less tractable forms of anaphoric and elliptical question types.

References:

[1] Grosz, B.J., "The representation and use of focus in Dialogue understanding". Technical Note 151, Artificial Intelligence Centre, SRI International Menlo Park, California, 1977.

[2] Grosz, B.J. and Sidner, C.L. (1986) "Attentions, intentions and the structure of discourse", in Computational Linguistics, Vol. 12, 3, 175-204

[3] Carter, D., *Interpreting anaphors in natural language text.*, Ellis Horwood Series in Artificial Intelligence, 1987.

[4] Rumelhart, D., "Notes on a schema for stories", In D. Bobrow and A. Collins (eds), *Representation and Understanding*, N.Y.: Academic Press, 1975.

[5] See: Reinhart, T., "Co-reference and Bound Anaphora: A restatement of the anaphora questions.", In *Linguistics and Philosophy*, 6, 1983.

[6] Woods, W.A., Kaplan, R.M., Nash-Webber, B., *The Lunar sciences Natural Language Interface System: Final Report*, Cambridge, Mass: Bolt, Beranek and Newman, 1972.

[7] Tufis, D and Cristea, D, "IURES: A human engineering approach to natural language question-answering systems", In *Artificial Intelligence: Methodology, Systems, Applications* , 1985.

[8] Templeton, M and Burger, J, "Problems in Natural Language Interface to DBMS with examples from EUFID", In *Proceedings of the Conference on Applied Natural Language Processing*, Dordrecht, 1983.

[9] Trost H, Burcherger E, Heinz Wolfgang, Hortnagl C and Matiasek J., "DATENBANK-DIALOG: A German Language Interface for Relational Databases", In *Applied Artificial Intelligence*: 1:181-203, 1989.

[10] Hobbs, J. and Shieber, S.M., An algorithm for generating quantifier scopings", *Computational Linguistics*, 13, 1987.

[11] Wilks, Y., "A preferential, pattern-seeking, semantics for natural language inference", *Artificial Intelligence*, 6, 1975.

[12] Rich, E., *Artificial Intelligence*, New York, NY: McGraw Hill 1983.

[13] See: Grosz, B. and Sidner, C.L., in Walker, D.E., *Understanding Spoken Language*, New York, NY: North Holland, 1978.

[14] Craig, I., *The CASSANDRA Architecture*, Chichester: Ellis Horwood, 1989.

[15] Kempson, R., "Definite NPs and Context-Dependence: A unified theory of anaphora", in Reasoning in Discourse and Context, London: Academic Press, 1986.

[16] Gazdar, G. and Mellish, C., *Natural Language Processing in Prolog*, Wokingham: Addison Wesley, 1989.

[17] Pereira, F. and Shieber, S.M., "Prolog and Natural Language Analysis", *CSLI Lecture Notes*, 10, Chicago: UP, Stanford, 1987,