A Natural Language Understanding system for reference resolution in information dialogues.

F.C Johnson
Department of Library and Information Studies
Manchester Metropolitan University
All Saints, Manchester M15 6BH.

Abstract

The benefits of a Natural Language Understanding (NLU) system for information seeking can only be realised if the system allows for effective communication. The system should be able to deal with the interpretation of referring expressions in dialogue, such as anaphors and ellipsis. In this paper, the components which comprise a NLU system to deal with continuous dialogue are described. Given that the syntactic and semantic information can produce a suitable representation of each utterance, pragmatic information may be used to determine how this contextual information determines the interpretation of subsequent utterances. It is suggested that the approach taken allows the system to provide a cooperative response to assist the user in attaining the information seeking goal.

Introduction

The benefits of a Natural Language (NL) interface which allows users to query a database using a Natural statement of the information need have been known for sometime (Smeaton 1989, Jacobs and Rau 1988). The casual user of the system would be less alienated from the system accessed via menus or command languages. A menu may not have the expressive powers of Natural language in which the information need can be expressed directly and accurately. The former, if it is to remain accessible to casual users, must use a simple hierarchical information access structure, which imposes a specific search path on the user. For a precisely-specified need, the NL expression may provide a much faster access to the required information and, even for a less specific request, the query should be processed with fewer question prompts than with a menu system (Aragon-Ramirez and Paice 1985).

Whilst it is possible to argue in favour of unstructured NL input, Winograd and Flores (1986) highlighted the danger that such an interface can easily let down users who might be led to overestimating the system's capabilities. Black (1992) emphasises some of the complex issues which must be addressed when NL understanding is seen to go beyond the simple translation of the users NL query into a corresponding database query language. In particular, he draws attention to the various linguistic devices used in apparently simple dialogues which require pragmatic interpretation. The basic premise for pragmatic interpretation is the assumption that utterances are performed intentionally since a speaker has a purpose behind any communication. This recognition of intention plays a fundamental role in all successful communication and draws on pragmatic information comprising the linguistic context of the dialogue and the non-linguistic context of world knowledge, expectations, beliefs, plans and goals of the dialogue participants, to determine how such contextual information affects the interpretation of an utterance.

In this paper, we show how a NL understanding system can be developed to utilise low-level pragmatic information, primarily the linguistic context, to deal effectively with the linguistic devices of anaphors and ellipsis. Anaphoric expressions, such as pronouns ("it", "they") or definite noun-phrases ("the Computer course") refer to discourse entities previously mentioned in the discourse for their interpretation. An ellipsis is an utterance in which part is omitted, but which can be inferred from the previous context. It may be regarded as a type of anaphor since it also refers back to some entity which when recognised provides the context for the full interpretation of the utterance. Anaphoric expressions are commonplace in discourse used as an abbreviating device to avoid repetition. In a system which could handle elliptical input, ellipsis was found to be used in 46% of the queries posed (Hendrix 1986).

Pragmatics and reference resolution

Grice's (1976) set of guiding principles for successful and cooperative communication says that any utterance is acceptable only if there is enough information for it to be related to the dialogue so that intention can be recognised and thus fulfilled. A good example of a system developed to recognise intention and so respond appropriately comes from Allen and Perrault (1980). When the query "*Can you tell me the departure time of the London train*?" is put to system, rather than simply responding "*yes*", it acts intentionally to help the speaker attain his goal and supplies the information required, the departure time and gate.

This dialogue principle also permits the use of referring expressions in dialogue. Reference resolution presents a stimulating challenge for the researcher in NL processing: more often than not there is more than one possible antecedent, the entity to which the expression refers for interpretation. Consider dialogue [1] between an information seeker (S) and an informant (I). The pronoun "it" in S2 refers back to the BSc course in I1 despite the intervening contender "the book" in S2.

S1> I want to study Information Science at Manchester
I1> There is a BSc course in Library and Information Management at the Manchester Metropolitan University.
S2> I read a book about the advances in Information Retrieval. It said that work in AI may

be the cutting edge for future research. Does it cover AI or something similar?

Dialogue [1].

The fact that "it" clearly refers to the BSc course may be owing to semantic information that courses cover subjects, but so do books. Pragmatic information, of the relation between intention and focus, what is currently talked about and available for subsequent reference, is needed to recognise the shift in focus from the book back to the course. The speaker's intention at S1 is to establish the existence of the specified course: which becomes the focus of S1 and I1. At S2, the speaker enters a 'subdialogue' with the intention to talk about an aspect of the course, its subject matter. The question posed in the present tense referring back to the BSc course indicates the close of the subdialogue and the expected return to talk about the previous focus. This relationship has been used to structure dialogue into focus domains for anaphor resolution in the work of Grosz and Sidner (1987) and can also be observed in dealing with ellipsis. Consider the interpretation of the elliptical utterance, S2, in dialogue [2]:

S1> Is there a course in Maths at Manchester University?
I1> There is a BSc course.
S2> in Computer Studies?
I2> There is a BSc course in Computation at Manchester.
S3> What are the entrance-requirements?
I3> Both courses require 3 'A' levels, one in Maths.

Dialogue [2].

The utterance "*in Computer studies*" can be seen to refer to and replace the current focus, "the BSc Maths course at Manchester University" in S1 to give the interpretation that the speaker intends to find out about courses in Computer Studies at Manchester University. Furthermore, if the intention recognised is to carry out a breadth search, to continue the dialogue to gather details about both the courses (perhaps to compare and eventually eliminate one in favour of the other) then both courses remain in focus for the interpretation of S3 where the cooperative response providing the entry requirements for both courses is provided.

This is a brief discussion on the processing of referring expressions. Fuller accounts of the information sources required for treating this discourse phenomena are given in Carter (1987), Grosz and Sidner (1986), for example. It does, however, illustrate the nature of the complexities which arise, referred to in Black (1992), when dealing with dialogue rather than a series of isolated utterances. Our aim is to address the need to recognise speakers' intention for effective reference resolution in information seeking dialogue to allow for a more natural and successful dialogue. It is, however, stressed at the outset that the amount of knowledge needed in a system depends on the level of analysis required. The aim here is to demonstrate that, given an adequate representation of the utterance, a focus stacking mechanism can maintain the structure of dialogue to suggest the candidates from which to select an interpretation. The system, which interprets dialogue on the subject of educational courses, is intended to be a framework into which more rich knowledge about pragmatics can be incorporated for a fully cooperative dialogue.

The architecture of the system

An overview of the steps taken in processing utterances in the system is shown in Figure 1. The vertical lines show the main flow of control as an utterance is passed through the various stages utilizing the syntactic, semantic and pragmatic information sources available to the system, shown in italics.

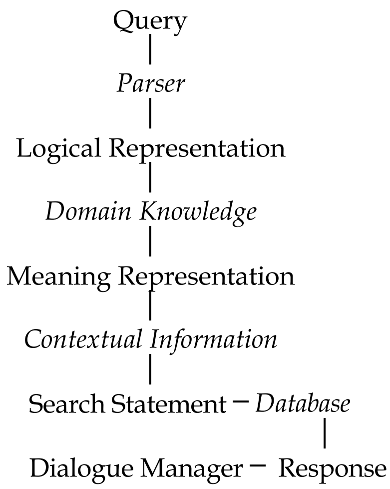

Figure [1]: The system's architecture.

The components employed in processing the dialogue are described below.

The Parser and Grammar

Syntactic analysis is the study of how words in a sentence are grouped into phrases and phrases into larger phrases. Such an analysis involves the use of a grammar which accounts for the major structures of well formed sentences and a parsing algorithm to specify how to apply a grammar to a sequence of words to produce a structural representation of the sentence. The choice of the grammar formalism, in this case a bottom-up chart parser using a generalised categorial grammar, is not considered to be too important although further detail on the rationale is given in Trogstad (1988).

Semantic Information

An important part of the processing is to map the syntactic analysis of the utterance into some symbolic structure to represent meaning in a form suitable for further processing. The principle of compositionality, that the meaning of a sentence is derived from the meaning of its component parts, is used in NL processing (Calder

1986) since it provides a formality to the question of how syntactic and semantic information interact. The structural meaning representation of a sentence can be obtained using First Order Logic (FOL). FOL representations are computationally tractable especially when the semantic representations quantify over events (Davidson 1980). This is suited to the deductive reasoning which is required to interpret the FOL representation into its contextual meaning.

Many systems (e.g., McCord 1980) enhance meaning representations with logical properties to express the semantic relations between the NL expressions with a subcategorization of objects on sortal semantic grounds. The objects are categorized into sorts and expressions referring to individuals are typed to indicate what sort is denoted. For example, the verb "teach" could be labelled as taking a subject with type HUMAN, as in "*the student taught the children*". However "*Do secondary schools teach Philosophy*?" would then be rejected according to the selectional restriction which is imposed to eliminate semantically anomalous word senses. For this reason a sorted logic is not employed in this system. Compositional semantics is employed only to obtain a domain independent semantic representation, a logical form which provides a set of assertions about the structural relationships among the entities. Also, the fact that a separate module is retained for the application of domain-knowledge means that no parsing of an utterance will be blocked by early application of semantic checking.

A second step is needed to evaluate this representation. This involves describing the utterance in terms of the world knowledge base or domain semantics. A similar two step approach to semantics is employed in systems described in Bronnenburg et al (1980), Bunt (1985), and Rich et al (1987).

Pragmatic Information

The amount of information in the domain dependent semantics (the world knowledge) in the system is guided by Carter's quantitative Maxim. Carter (1987) proposes a shallow approach to anaphor resolution in which extensive world knowledge is not required. He describes it as a 'shallow approach' since linguistic knowledge, particularly about focusing, is exploited to resolve anaphors so that limited quantities of world knowledge are used only when necessary. Thus, the word "restaurant" is defined as "a place where people eat food" rather than using a complex script (Schank and Abelson, 1977) with detail about waiters, tables, menus and so on. As our aim is comparable to that of Carter's, the amount of domain-dependent information provided about a word is only enough to be considered basic to its meaning. The context is represented as a set of discourse entities derived from the previous utterances and integrated with the world knowledge to encompass anything that can be subject to subsequent reference. This contextual information is passed to the dialogue manager which maintains the dialogue's structure and makes available the current focus for subsequent reference. Thus as each utterance is posed to the system, it is passed to the dialogue manager to determine its relation with the current focus. The effect of this interpretation is to update the focus state according to the relation, or intention, recognised.

We now show how these basic information sources are integrated to develop a system capable of handling the linguistic devices of anaphor and ellipsis, as discussed above.

Representation in Logical Form.

The input to the parser is a string partitioned into words. The output is a logical form (LF) representation of the utterance. For example, the LF representation of "*is there a course in Computation*?" states that there exists some X such that X is an instance of a course and X is in B where B has the name computation:

E(X,(inst(X,course)&
name(B,computation)&in(X,B)))

The LF representation is constructed using the following system components, a categorial grammar enhanced with compositional semantics and a parser. Categorial grammars have become increasingly popular as a basis for parsers and have been extensively documented elsewhere (Barry 1988, Steedman 1990, Dowty 1987, Moortgat 1987, Morrill 1988).

All categorial grammars have two basic components:
1. A lexicon which assigns the words of a language with one or more syntactic categories. For example, the category assigned to a determiner, np/n, states that a determiner combines with a noun, category n, to form a np.
2. A small set of syntactic rules for combining categories. The simplest combinatory rules where X and Y stand for any syntactic categories are,

(f) X/Y + Y => X [forward application]
(b) Y + X\Y => X [backward application]

The first rule, (f), states that a string consisting of a substring of category np/n (a determiner) followed by a substring of category n (a noun) may be assigned of category np (a nounphrase). While the second rule, (b), states that a string consisting of a substring of category np followed by a substring of category s\np (a verb-phrase) may be assigned the category s (a sentence).

The parsing algorithm accepts a word, finds its category and store this information in a chart in the form of edges. When each word is parsed and added to the chart, an attempt is made to reduce its category according to some combination rule with that of each arc meeting it from the left. If the reduction is successful, a new arc is added and the process is repeated calling itself recursively. The procedure is illustrated below which shows the resulting arcs in order of generation for the parse, "I studied Maths".

(a)	I	arc(0, np, s(0)).
(b)	STUDIED	arc(s(0), s\np/np, s(s(0))).
(c)	I STUDIED	arc(0, s/np, s(s(0))).
(d)	MATHS	arc(s(s(0)), np, s(s(s(0)))).
(e)	I STUDIED MATHS	arc(0, s, s(s(s(0)))).

The parser is using forward and backward application rules and an additional metarule to permit the incremental left to right reading[1]. The first call produces the arc asserted as (a). The next call produces (b) which meets arc (a) on its left hand side and a combinatory metarule (r(b)) is called to reduce the string to (c). The procedure then fails and the next call produces (d), which meets arc (c) and so reduces to (e).

With the parser and grammar in place, each lexical entry is extended to include semantics in the form:

Word:= syntax:semantics.

The semantic field contains a semantic template, a pattern of well-formed formulas of First Order Logic (FOL). The rendering of the semantics into FOL was based on Jowsey's (1987) simplified version of Montague grammar which has also been used to represent texts in first order logic for document indexing and retrieval (Sembok and van Rijsbergen 1990). The logical representation of an utterance can now be obtained by applying the above syntactic rules for combining categories. This is illustrated below in the categories assigned to determiners, nouns and verbs to show the derivation of a sentence.

The syntactic category of a determiner states that it combines with a noun to form a noun-phrase (np). The semantic information is given as a conjunction of predicates with any quantifiers represented as functors at the head of the list. For example, "a" is represented with the existential quantifier, E (standing for "there exists" X) while "the" is represented with I (standing for iota, "there is some unique" X). In the semantic templates shown below, the derived np will have some quantified variable, X, which has the semantics of the noun, (A), to be combined, and a prediction, (P), for any additional information there may be about the referent X.

Word:=	a	the
Syntax:	np/n	np/n
Semantics	[X*P]*E(X,A&P)/[X*A]	[X*P]*i(X,A&P)/[X*A]

[1]To avoid spurious ambiguity where the parser produces syntactically distinct but semantically equivalent derivations for the same sentence, an equivalence check described in Barry (1988) is employed to favour the left to right readings.

Thus, the determiner "a" combined with a noun "student", with the semantics [X*inst(X,student)] gives the np [X*P]*E(X,inst(X,student)&P). The head noun may be further modified by adjectives or prepositional phrases to convey the properties of the noun.

In the semantic representation of the vp, use is made of Davidson's (1980) treatment of verbs in which a semantic representation quantifies over events. The event variable leads to a simple semantic treatment of sentential modifiers and of control verbs looking for an infinitive sentential complement, such as "*I studied in Manchester*" where "in Manchester" can be predicated of the past event of studying.

An intransitive verb only combines with a subject, and a transitive verb takes a subject and an object. Intransitive verb semantics, as shown for the verb "work" below ([X*event(E)&work(E)&arg0(E,X)), when combined with its subject np, fills the prediction, P, in the np semantics ([X*P]*E(X,inst(X,student)&P)). F, the sentence semantics, is a conjunction of predicates including the semantics of the subject np and the information coming from outside which in this case is the verb semantics, to give (E(X,inst(X,student)&event(E)&work(E)&arg0(E,X)).

```
Word:= work                   study
Syntax:      s\np                     s\np/np
Semantics    F\[[X*event(E)&F\[X*Z]*F/
             work(E)&                 [ [Y*event(E)&
             arg0(E,X)]*                 do(E)&
             F].                   arg0(E,X)&
                                       arg1(E,Y]*
                                         Z]
```

Transitive verbs seek a subject and an object so an extra argument in the vp semantics is necessary. The vp is created with the object np so that the prediction in this np is filled with information from the vp and arg1 of the vp is the variable marker of the object np. The sentence semantics comes from the subject np whose prediction, P, is filled with the semantics of the object np.

World Knowledge

The next step is to evaluate the logical form representation to demonstrate the system's 'understanding' of an utterance and query a database for a response. This is achieved by mapping the above output onto the ontology in which knowledge about the domain is expressed. This world model is represented as a hierarchy of classes with instances and properties. For example, the root class 'award' can be represented with three subclasses of, degree, diploma and exam, where *isa* specifies a subclass relation and *inst* specifies instances of a subclass.

This information can be expressed as Prolog facts:

```
isa(diploma,award).
isa(degree,award).
inst(hnd,diploma).
inst(bsc,degree).
```

Further information can be added to the world knowledge database as necessary.

```
inst(maths,subject).
inst('computer-studies',subject).

isa(university,institution).
inst('manchester-university',university).
```

In addition, rules are used to map the predicates in the LF to predicates in the world knowledge representation. This gives a one-to-one correspondence between the words in the lexicon and the entities and relations in the database.

The world knowledge pertaining to the relations held between words in the domain is expressed in Prolog rules of the form:

```
Predicates => Relation :- Condition.
```

Each predicate in the logical representation of the utterance can be represented by some relation in the database if the corresponding condition in the above world model can be satisfied. For example, the following rules state that a course has a subject, is taught at a level, its outcome is an award, and is provided by an academic institution e.g., a Maths course is at degree level to award a BSc, and is provided by an University.

```
name(Y,Z)&in(X,Y)  => subj(X,Z)      :- inst(Z,subject).
hasproperty(X,Z)   => level(X,Z)     :- isa(Z,award).
hasproperty(X,Z)   => awardname(X,Z) :- inst(Z,A), isa(A,award).
name(Y,Z)&at(X,Y)  => provider(X,Z)  :- inst(Z,I), isa(I,institution).
```

An example query put to the system is shown in Conversation [1] below to illustrate how meaning is computed. The logical form is mapped onto the world knowledge using the rules, the resulting meaning representation or search statement can be matched against the database to provide the response.

```
input>    is there a course in Maths at UMIST?
response> course, c5 has subject Maths, provider UMIST.

rules>   name(Y,Z)&in(X,Y)  =>  subj(X,Z) :- inst(Z,subject)
  name(Y,Z)&at(X,Y)  =>  provider(X,Z) :- inst(Z,I), isa(I,institution)
```

Logical Form	Meaning Representation	Topic
E(X,(((inst(X,course)& name(D,maths)& in(X,D))& name(F,umist)& at(X,F))	E(X,[inst(X,course), subj(X,maths), provider(X,umist)])	[inst(c5,course), subj(c5,maths), provider(c5,umist)]

Conversation [1].

Providing no anaphor or ellipsis are to be resolved, the output, the meaning representation is matched against the database to get a response. The response is created by using canned expressions and the information in the meaning representation. For example, [inst(X,course), subj(X,maths), provider(X,umist)] and the entity, c5 in the database which satisfies these conditions, would provide the canned response, "course, c5 has subject Maths, provider UMIST."

The dialogue manager then takes the meaning representation instantiated with the discourse entity marker obtained from the response, c5, and stores this in the focus stack as the current topic to be made available for subsequent reference.

Focus and Reference Resolution

We can now demonstrate how the focus stack is used to determine how each subsequent utterance in continuous dialogue relates to the context for interpretation. A set of dialogue rules, based on Watchel's (1989) dialogue grammar, are used to create expectations for the continuation of dialogue invoked by the relation held between the utterance and the context. These rules allow a speaker to introduce a new topic into the dialogue, continue a dialogue to gather further details about a topic or enter a subdialogue to talk about related aspects of the topic.

The contextual information available for interpretation is asserted as a Prolog fact with the following information:

conversation([dialogue(N), type, exchange(N), move(N), move(N)])

The entire dialogue, labelled 'conversation', is made up of any number (N) of dialogues each of which introduce a new topic into the conversation. The type of dialogue, **new** dialogue, or **continue** dialogue, or **subdialogue**, is labelled according to the rule invoked.

The two moves are the query-response pair in our information seeking dialogues which comprise an exchange in a dialogue. In addition, the search statement derived from the meaning representation is associated with the exchange. The topic which is associated with the dialogue is obtained, as before, by instantiating the variables in the meaning representation with the response found in the database.

For example the above utterance "*is there a course in Maths at UMIST*?" would invoke a rule, **D0s(query)& new topic => create new dialogue**. This says that the context is one in which the speaker has uttered a query introducing a new topic and the consequent is to create a new dialogue about the topic "Maths course at UMIST", as shown in Conversation [1a]:

input> is there a course in Maths at UMIST.
response> course c5 has subject Maths, provider UMIST.

[dialogue1,	new,	exch1,	move1	move2]
[inst(c5,course),		[inst(X,course),		
subj(c5,maths),		subj(X,maths),		
provider(c5,umist)]		provider(X,umist)]		

Conversation [1a].

This contextual information makes available the entities which can be used to resolve anaphoric reference or ellipsis.

Anaphor Resolution

Referring expressions occur in continuous dialogue as opposed to a series of isolated utterances. The range of anaphor dealt with is limited to definite nps but allow an interesting variety of user interactions. Anaphoric reference is detected by the use of the definite article "the" when the LF representation is translated into a meaning representation. When this is passed to the contextual interpreter the anaphoric description is matched against the current dialogue topic. For example, in Conversation [2], the meaning representation of the anaphoric expression, "the course" in S2>, ([inst(X,course),provider(X,umist)]), is matched against the topic of the current dialogue, ([inst(c5,course), subj(c5,maths)]).

S1> is there a course in Maths.
I1> course c5 has subject Maths.
S2> is the course at UMIST.
I2> course c5 has subject Maths, provider UMIST.

Conversation [2].

Since one of the predicates matches the topic of the current dialogue topic, the anaphor is resolved by replacing the variable in the meaning representation with the constant which identifies the course in the topic. For example, the meaning representation becomes ([inst(c5,course), provider(c5,umist)]) with the anaphor resolved. This context where the current topic is referred to matches the rule, **DOs(anaphor)& refers to topic dialogueX => update topic & create new exchange in dialogueX**, which states that a new exchange should be created in the dialogue to gather more information about the topic. Accordingly, the topic is updated following the exchange, S2 and I2 in the above dialogue, by adding the new information which, in this case, is the information provider(c5,umist):

[dialogue1,	continue	exch2,	move3,	move4]
([inst(c5,course),		([inst(c5,course),		
subj(c5,maths),		provider(c5,umist)])		
provider(c5,umist)])				

Associative referring expressions are also recognized where there is some relation, in the world model, between the antecedent and the referring expression. For example, in continuing the Conversation [2a], the associative reference "*the entrance requirements*?" relates, by way of an attribute, to the topic of a course specification in dialogue1. The utterance is understood as "*what is the entrance requirement for the Maths course c5*?", as is shown in the resulting search statement beneath exch3. As before, the consequent is that the topic is updated and a new exchange in dialogue1 is created.

S3> what is the entrance-requirement.
I3> course c5 has subject Maths, requires
2 a-level(Maths) 3 o-level(English)

[dialogue1,	continue,	exch3,	move5,	move6]
inst(c5,course),		[inst(A,entreq),		
subj(c5,maths),		inst(c5,course),		
attribute(c5,		attribute(c5,A]		
(exam(a-level,2,maths)				
exam(o-level,3,english))]				

Conversation [2a].

In this way, the dialogue manager is able to maintain the focus of the dialogue according to the rule invoked. Now, if the speaker enters a subdialogue to talk about a related aspect of the focus, then a 'subdialogue rule' could be invoked to recognise this shift. This would ensure that the previous focus remains available for reference when the speaker closes the subdialogue. Such a rule, which maintains two or more items in focus, is used for ellipsis resolution.

Ellipsis Resolution

Ellipsis is recognized if a sentence construction was not found during interpretation of the logical form representation. As with anaphor resolution, the meaning representation is passed to the contextual interpreter for its full interpretation. Ellipsis, shown in Conversation [3], at S3, matches and replaces a predicate in the topic of the dialogue and can be dealt with by a simple substitution procedure. For example, the representation of the elliptical utterance, [subj(X,maths)], matches the functor of the predicate subj(c8,computation) in the current dialogue topic. Notice that this creates a search statement which appears cooperative accounting for all requirements specified as the dialogue built up the current topic, [inst(X,course), subj(X,maths), provider(X,umist], duration(X,D)).

S1> Is there a course in Computation at UMIST.
I1> course c8 has subject Computation, provider UMIST.

```
[dialogue1,      new,           exch1,    move1,   move2]
([inst(c8,course),        ([inst(X,course),
subj(c8,computation),           subj(X,computation),
provider(c8,umist)])            provider(X,umist)])
```

S2> how long does the course last.
I2> course c8 has subject Computation, provider UMIST,
duration 3-years

```
[dialogue1,      continue,      exch2,          move3,   move4]
([inst(c8,course),        ([inst(c8,course),
subj(c8,computation),           duration(c8,D)])
provider(c8,umist)])
duration(c8,3-years)])
```

S3> in Maths.
I3> course c9 has subject Maths, provider UMIST,
duration 3-years

```
[dialogue2,      subdialogue,   exch3,          move5    move6]
([inst(c9,course),        ([inst(X,course),
provider(c9,umist),             provider(X,umist),
subj(c9,maths)])                subj(X,maths)])
duration(c9,3-years)])          duration(X,D)])
```

Conversation [3].

The relationship between the dialogue context and the elliptical utterance invokes the rule, **DOs(ellipsis)& refers to topic dialogueY => create subdialogue in dialogueY.** As a result, the current context being a subdialogue ensures that both the topic of the subdialogue, labelled dialogue2, and the topic of the previous dialogue, dialogue1, remain available for reference. For example, a subsequent utterance "*What are the entrance-requirements?*" could be a request relating to both the courses. Or if the speaker continues the dialogue with "*What about courses at Manchester University?*" the system would appear to be helpful if it understood this to refer to both Computer and Maths courses. The dialogue manager rules ensure that the available context is taken into account in interpretation and the consequence of invoking the rules updates the dialogue context is accordingly. That is, the speakers intention is recognised as its context changing effect in terms of what is in focus.

Conclusion

In this paper, we have shown how the various components of a NL understanding system may be developed to deal effectively with information seeking dialogue. Particular attention was given to two linguistic devices which rely on contextual information for their interpretation. The meaning representation built for each utterance provides this linguistic context in the form of a 'topic' which the speaker has focused attention on. When an utterance refers to the topic, a dialogue rule is invoked to determine the effect the interpretation of this utterance has on the current focus state and updates it accordingly.

It is suggested that these rules indirectly capture intention and allow for a more cooperative conversation. That is, the system interprets the utterance in a way which will help the user attain his goal quickly and easily. Anaphoric reference allows the speaker to refer to, and thus gather more information about, the current dialogue focus. Ellipsis, another form of abbreviation in information dialogue, allows the speaker to quickly shift the focus of attention. The system maintains the focus, what is made available for subsequent reference following the interpretation of anaphors or ellipses, to ensure that it is able to provide the most cooperative response given this contextual information.

The framework in which the system has been built presents an opportunity for future enhancement, perhaps to incorporate further pragmatic information about speakers' plans and goals. However, with the development of NL processing techniques to handle increasingly complex dialogue, a word of caution. Research in both NL processing and Information Science is needed to discover more about how people search for information. Dialogues with prototype systems, such as the one described here, ought to be collected and analyzed with a view to understanding information seeking behaviour. A NL understanding system may claim advantages over a command or menu driven system if empirical evidence demonstrates its ability to efficiently handle information dialogues.

References

Allen, J.F. and Perrault C.R. Analyzing intention in utterances. *Artificial Intelligence*, 15(3), 1980, 143-178.

Aragon-Ramirez, V. and Paice, C.D. Design of a system for online elucidation of natural language search statements. *Proceedings of Informatics*, 8, 1985, 163-190.

Barry, G.D. Parsing strategies for extended categorial grammars. *Unpublished MSc Dissertation*, UMIST Manchester, 1988.

Black, W.J. Rule-based systems, natural language processing and abstracting. *In:* Anne Morris *ed*. *The application of expert systems in libraries and information centres*. London: Bowker-Saur, 1992, 68-91.

Bronnenburg, W.J., Bunt, H.C., Landsbergen, S.P.J., Scha, R.J.H., Schoenmakers, W.J., and Van Utteren, P.C. The question answering system PHLIQA1. *In*: L. Bolc *ed*. *Natural language communication with computers*. Munich: Hauser, 1980.
Bunt, H.C. *Mass terms and model theoretic semantics*. Cambridge: Cambridge University Press, 1985.
Calder, J., Klein, E., Moens, M., and Zeevat, H. Problems of dialogue parsing. *Research Paper EUCCS/RP-1*, University of Edinburgh, 1986.
Carter, D. *Interpreting anaphors in natural language texts*. Chichester: Ellis Horwood, 1987. Davidson, D. *The logical form of action sentences*. Reprinted in his Essays on Actions and Events, Oxford: Clarendon Press, 1980.
Dowty, D. Type-raising, functional composition and non- constituent conjunction. *In*: R. Oerhle et al *eds. Categorial grammar and natural language structure*. Dordrecht: D.Reidel, 1987.
Grice, H.P. Logic and conversation. *In:* P. Cole and J.L Morgan *eds*. *Syntax and semantics, Vo1 3: Speech acts*. New York: Academic Press, 1975.
Grosz, B. and Sidner, C.L. Attention, intention and the structure of discourse. *Computational Linguistics*, 12(3), 1986, 175-204.
Hendrix, G, Sacerdoti, D, Sagalowicz, D, and Slocum, J, Developing a natural language interface to complex data. *ACM Transactions on database systems*, 3(2), 1978, 105-147.
Jacobs, P.S. and Rau, L.F. Natural language techniques for intelligent information retrieval. *Proceedings 11th ACM SIGIR Conference*, Grenoble 1988, 85-99.
Jowsey, H.E. Montague grammar and first-order logic. *In*: N.J. Haddock, E. Klein and G. Morrill *eds WPs in Cognitive Science*. 1987, 143-194.
McCord, M.C. Slot grammars. *American Journal of Computational Linguistic*s, 6(1), 1980, 31-43.
Moorgat, M. Lambek theorem proving. *Papers on the ZWO worsop*, TLI Amsterdam working papers, June 1987, 87-94.
Morrill, G. Extraction and coordination in phrase structure grammar and categorial grammar. *Unpublished Phd Thesis*, Centre for Cognitive Science, University of Edinburgh, 1988.
Rich, E. Wittenburg, K. Barnett, J. and Wroblewski, D. Ambiguity procrastination. *6th international Conference on Artificial Intelligence* vol. 2, Seattle, Washington, July 1987, 13-17.
Schank, R. and Abelson, R. *Scripts, plans, goals, and understanding*. Hillsdale, N.J.: Lawrence Erlbaum Associates, 1977.
Sembok, T.M.T and van Rijsbergen, C.J. SILOL: A simple logical-linguistic document retrieval system. *Information Processing and Management*, 26(1), 1990 111-134.
Smeaton, A. F. Information retrieval and natural language processing. *Informatics* 10, Cambridge 1989 1-15.
Steedman, M. Grammar, interpretation and processing from the lexicon. *In*: W. Marsten-Wilson *ed. Lexical representation and process*. MIT Press, 1990, 463-504.
Trogstad, P.J., Black, W.J., and Johnson, F.C. Implementation strategies for robust database interfaces. *Digest of the I.E.E Computing and Control Division Colloquium on NLU*, London, 14th Nov 1988.
Wachtel, T. Discourse structure in LOQUI. *In*: J. Peckman *ed. Recent developments and*

applications of natural language understanding. England: Kogan Page, 1989.
Winograd, T. and Flores, F.C *Understanding Computers and Cognition*. Norwood N.J: Ablex, 1986.