

An Integrated Approach to Reference and Presupposition Resolution

Robert T. Kasper, Paul C. Davis*, and Craig Roberts

Department of Linguistics
Ohio State University
222 Oxley Hall
1712 Neil Avenue
Columbus, OH 43210

e-mail: {kasper, pcdavis, croberts}@ling.ohio-state.edu

Abstract

We describe an approach to resolving definite descriptions and pronominal anaphora as subcases of a general strategy for presupposition satisfaction. Generally, a presupposition is satisfied in a context if the context contains a specific type of information and is organized in such a way that this information can be retrieved by the interlocutors. The model of discourse context we develop assumes that discourse structure is organized around a stack of questions under discussion, which plays a crucial role in narrowing the search for referents and other presupposed information. The algorithms for maintaining the discourse structures and retrieving presupposed information are presented and illustrated by several example dialogues in which human users interact with an agent to make hotel reservations.

1 Introduction

Any theory of referring expressions must take into account the discourse context in which they occur. Indeed, previous research has shown that the hierarchical organization of discourse is fundamentally related to the reference resolution process. In this paper, we show how a highly structured discourse model, in conjunction with a treatment of referring expressions as presuppositional, enables us to develop a common strategy for resolving a number of reference resolution problems, such as pronominal anaphora and definite descriptions. We also outline how this approach extends to a larger group of phenomena which we take to be presuppositional, including domain restriction, ellipsis, and lexically and syntactically triggered presuppositions. All of these constructions are presuppositional in a broad sense, in that their use assumes that certain information can be retrieved from the discourse context.

Recognizing the structure of the discourse will therefore play a crucial role in narrowing the search for referents and other presupposed information. We will illustrate our approach with four example human-computer dialogues, shown below. SYS indicates the utterances spoken by the computer system.

Example I illustrates a case of pronominal anaphora resolution (*it* in (8)), in which recognizing the hierarchical structure of the discourse is crucial for identifying the antecedent, which was introduced many utterances earlier. The overall topic of the conversation is the question of where the user can find a hotel for June 15th in New York, and this super-question both facilitates and constrains the interpretation of *it* in (8). This example is similar to the well-known examples of long-distance anaphora in task-oriented dialogues described by Grosz (1981). Our approach is consistent with previous research that uses the intentional structure of discourse to determine a set of potential antecedents for pronominal anaphora. The following examples will illustrate how a broader range of reference and presuppositional constructions may also be addressed by using the discourse structure to guide the search for relevant information.

Example I.

- 1) USER: I'm looking for a hotel for June 15th in New York.
- 2) SYS: What part of the city would you prefer?
- 3) USER: Manhattan, near Central Park.
- 4) SYS: How many nights?
- 5) USER: Just 1.
- 6) SYS: Will anyone be traveling with you?
- 7) USER: No.
- 8) USER: Oh, I want it to have a swimming pool too.

* Paul C. Davis is the recipient of a Motorola Partnerships in Research Grant.

Example II shows a definite description, *the hotel* in (7), whose referent can only be uniquely determined with respect to the indefinite hotel description (*a hotel close to Madison Square Garden*) in the question under discussion (1):

Example II.

- 1) USER: I want to make a reservation at a hotel close to Madison Square Garden.
- 2) SYS: What dates will the reservation be for?
- 3) USER: March 3rd and 4th.
- 4) SYS: Would you like a single room?
- 5) USER: Yes.
- 6) USER: Also, I'll need a conference room on the 4th.
- 7) USER: I'd prefer it if the hotel had one.

Example III involves a contextually determined domain restriction, with a quantificational determiner *every*, illustrating that domain restriction must be handled in a similar way for a broader class of expressions than those which are normally regarded as referring expressions or presupposition triggers.

Example III.

- 1) USER: Does the Holiday Inn have any vacancies for
 - a) Tuesday, 12/4 - Friday 12/7?
 - b) Thursday, 12/6 - Saturday 12/8?
- 2) SYS: Yes, several.
- 3) USER: Do they have a breakfast buffet every morning?
- 4) SYS:
 - a) Yes, Monday through Friday.
 - b) No. There's a breakfast buffet Monday through Friday, but none on Saturday.

Finally, in example IV we give a glimpse into our larger research program, where an elliptical question (3) must be resolved with respect to the question under discussion, in addition to establishing the reference of the definite description *the Marriott*, where the context might contain more than one hotel with that name:

Example IV.

- 1) USER: Which hotels near the airport have vacancies?
- 2) SYS: The Holiday Inn and Sheraton have vacancies.
- 3) USER: How about the Marriott?
- 4) SYS: No, the airport Marriott doesn't have any vacancies.

The remainder of the paper is organized as follows. In section 2 we discuss our assumptions about the

structure of discourse and the related background literature. In section 3, we present algorithms which we have developed in a partially completed implementation of a natural language dialogue system where users interact with an automated hotel reservation booking system. In section 4, we discuss the use of the algorithms and discourse structures to resolve the reference and presupposition problems shown in the above examples. In the final section, we highlight the contributions of our approach and discuss future plans related to this research.

2 Background: Discourse Structure

We assume the general theoretical framework of Roberts (1996), where discourse is formally characterized as a game of intentional inquiry. As in Grosz & Sidner (1986), discourse is organized by the interlocutors' goals and intentions and the plans, or strategies, which conversational participants develop to achieve them. Following Stalnaker (1979), the primary goal of the language game is communal inquiry, i.e., interlocutors attempting to share information about their world, with the repository of that shared information characterized as the interlocutors' *common ground*, CG. The set of acceptable moves in the game are defined by the (conventional and conversational) rules of the game, and are classified on the basis of their relationship to the goals. Ignoring imperatives, there are two main types of moves (see also Carlson 1983): questions and assertions. If a question is accepted by the interlocutors, this commits them to a common discourse goal, finding a satisfactory (asserted) answer; like the commitment to a goal in Planning Theory, this strong commitment persists until the goal is satisfied or else shown to be unsatisfiable. The accepted question becomes the immediate topic of discussion, the *question under discussion*. An assertion is a move which proposes an addition of information to the CG.

Roberts defines the structure of a discourse at a given point, its *Information Structure*, as a tuple which includes (among other things) the ordered set of moves in the discourse (M), CG, and the set of the questions currently under discussion at that point (QUD). The QUD is ordered by order of utterance and is updated in a stack-like fashion,¹ with questions popped when they are answered (or determined to be practically unanswerable). The ordered set of questions under discussion corresponds to the hierarchical intentional structure of the discourse. The QUD in this structure constitutes the set of *discourse goals* of the interlocutors; the discourse goals are only a subset of the set of common goals of the interlocutors, their *domain goals*, and

¹However, all elements of the QUD list are accessible during the interpretation of an utterance. Only the top element is writable, but any entry is readable.

the discourse goals are subordinate to, and subserve the domain goals. Hence, the requirement that interlocutors stick to the question under discussion is just an instance of the more general commitment to plans; and in turn, in a fully integrated theory we would expect that domain goals and plans would influence interpretation as directly as the discourse goals represented by the questions under discussion.

Any move in a discourse game is interpreted with respect to the Information Structure of the discourse at that point. There are two main aspects to the interpretation of any given move: its *presupposed content* and its *proffered content*, the latter including what is asserted in an assertion and the non-presupposed content of questions and commands. When an utterance presupposes a proposition p , then in order for the utterance to be felicitous in the context, p must be entailed by the CG (Stalnaker 1979). But in addition, any move in a discourse is interpreted by interlocutors under the Gricean meta-presupposition of Relevance, with Relevance formally defined as follows in Roberts' framework:

- (1) A move m is Relevant to the question under discussion q iff (i) m is an assertion such that $CGU\{m\}$ entails a partial answer to q , or (ii) m is a question whose complete answer contextually entails a partial answer to q .

(1(i)) tells us that the interpretation of an assertion will be constrained so as to yield a partial answer (possibly via contextual entailment) to the question under discussion. (1(ii)) tells us that the QUD in a felicitous Information Structure is constrained by Relevance so that each question on the QUD must address the (prior) question below it on the stack. Of course, (1) correctly predicts a variety of classical Gricean conversational implicatures, now characterizable as contextual entailments. But Roberts argues that Relevance is also crucial in *presupposition resolution*, broadly construed to include anaphora resolution, the interpretation of ellipsis, and domain restriction (Roberts 1995), as well as lexically and syntactically triggered presuppositions.

We will also assume the general approach to anaphora resolution argued for in Roberts (1999). The CG is augmented with a set of discourse referents familiar to the interlocutors, the *Domain* of the discourse context. All definite NPs, including pronouns and demonstratives as well as definite descriptions using *the*, presuppose both *weak familiarity* and *informational uniqueness*. Weak familiarity (cf. the slightly different notion of familiarity in Heim 1982) is the theoretical realization of anaphoricity, and is licensed by existential entailments of the common ground, not requiring an explicit NP antecedent or even perceptual salience of the intended referent:

- (2) **Weak Familiarity:** A discourse referent i is weakly familiar in a context C ($i \in \text{Domain}(C)$) and C encodes the information that i has properties P_1, \dots, P_k iff the Common Ground of C entails the existence of an entity with properties P_1, \dots, P_k .

Informational uniqueness only requires that the discourse referent which satisfies the definite's familiarity presupposition be unique among the discourse referents in the context in satisfying the definite's descriptive content. These two constraints suffice to characterize the presuppositional content of definite descriptions:

- (3) **Presuppositions of Definite Descriptions (informal):** Given a context C , use of a definite description NP_i presupposes that there is a discourse referent weakly familiar in C which is the unique weakly familiar discourse referent which satisfies the (possibly contextually restricted) descriptive content of NP_i .

Unlike Russell's (1905) theory, this does not generally entail semantic uniqueness, although in certain special contexts it will yield the same effect via pragmatic means. Definite descriptions may have their descriptive content contextually enriched in the same way that domain restriction works for operators generally, i.e., via Relevance to the question under discussion. This will be illustrated in our discussion of example 4 below. Many apparent counterexamples to the presupposition of uniqueness for definite descriptions are solved by appeal to this principled contextual enrichment, as discussed at length in Roberts (1999). Pronouns carry an additional presupposition of maximal salience:

- (4) **Presuppositions of Pronouns (informal):** Given a context C , use of a pronoun Pro_i presupposes that there is a discourse referent i in C which is the unique weakly familiar discourse referent that is both maximally salient and satisfies the descriptive content suggested by the person, number and gender of Pro_i .

This amounts to an additional, conventional restriction on the search space for pronominal antecedents, implemented along the general lines suggested by Grosz & Sidner, and explains the differential distribution of pronouns and definite descriptions. We will discuss how maximal salience is implemented in terms of the QUD stack in §4. These presuppositional constraints result in a straightforward theory of anaphoric reference which explains a broad range of data and can be extended to a treatment of demonstrative NPs as definites, as well.

```

process_utterance (U)
%%% 1. Determine contextually interpreted meaning.
ULF = parse(U)
(CULF, CDRS) = determine_CULF(ULF, Level)

%%% 2. Update discourse structures.
If presuppositions remain,
  attempt to accommodate them by adding information from system database to CG.
  If accommodation fails (system has no information or system information is
    inconsistent with CG), indicate non-acceptance of move.
If U is an assertion:
  assert CULF to CG,
  update QDL of QUD[top] (i.e., merge CDRS into CDRS of QDL)
If U is a question:
  push new QDL entry <ULF, CULF, CDRS> onto QUD

%%% 3. Call back-end application.
Perform SYSTEM action (e.g., query or update database)
Perform SYSTEM dialogue move if necessary (e.g., generate a response)

```

```

determine_CULF (U, Level)
  if atomic_formula(U) % contains no presuppositional operators
    return (U, {})
  else (U must contain an operator)
    return resolve_term(U, Level + 1)

```

Figure 1: Presupposition resolution algorithm

3 Resolution Algorithms

In Figures 1, 2, and 3 (shown later in §4.4), we show simplified, pseudo-coded versions of the algorithms which drive the presupposition resolution process. Of central importance in this process is the maintenance of the QUD stack. Each entry on the stack is represented by a Question Data Log (QDL), an ordered triple which contains the utterance's logical form (ULF), its Contextually Understood LF (CULF), and a set of current discourse referents (CDRS). QDL entries represent information about units of discourse structure which roughly correspond to the discourse segments developed by Grosz and Sidner.

Process_utterance is the top-level function invoked for each discourse utterance. The utterance is parsed to yield a logical form representing its context-independent meaning (ULF). This ULF is further processed by determine_CULF, the goal of which is to produce a refined logical form (CULF) and a set of discourse referents (CDRS) by resolving presuppositions with respect to the current context. Presuppositions are represented in the logical form by certain operators, including def, pronoun, λ (for wh-questions), and WHEllipsis. The terms introduced by these operators, as well as other generalized quantifier terms, are processed by the resolve_term function (see Figure 2). The set of

presuppositional operators listed in this algorithm covers the examples that we will discuss, but is not intended to be exhaustive. After resolve_term has processed a presuppositional term, the variable that it binds will appear on the CDRS list, and will either be identified with a set of referents from the common ground or be unanchored (indicated by the empty set or '?'). Once the CULF and CDRS are determined, the discourse structures, including the CG and QUD, are updated, depending on the type of conversational move (i.e., assertion or question). After the dialogue model has been updated, the CULF is sent to the back-end application (e.g., to query or update its database), and the system may generate utterances as needed.

The algorithms presented here have been implemented in Common Lisp, using the Loom knowledge representation framework (MacGregor (1991)) to maintain the common ground and background knowledge of the hotel application domain. Several components, e.g., the match&substitute and add_domain_restriction functions, have not yet been implemented in a fully general way, and currently handle only simplified cases. The examples discussed in the next section demonstrate how the resolution procedure works.

```

resolve_term(Term, Level)
Let OP      = top-level operator of Term
    VAR     = top-level variable of Term
    RESTR   = top-level restriction of Term
    NS      = top-level nuclear scope of Term

%%% Process embedded formulas inside-out
(RESTR1, CDRS_R) = determine_CULF(RESTR, Level)
(NS1, CDRS_NS)  = determine_CULF(NS, Level)

if OP is a non-presuppositional operator:
    DomainRestr = add_domain_restriction(VAR, RESTR1, QUD)
    return (OP[VAR,DomainRestr,NS1], CDRS_R  $\cup$  CDRS_NS)

else (handle according to OP type)
case OP=pronoun: % must be anaphoric reference
    RANKED_REFERENTS = rank_accessible_referents(QUD, RESTR1)
    REFERENT_SET = maximal_elements(RANKED_REFERENTS)
    If singleton(REFERENT_SET),
        %%% assume REFERENT_SET = {INST}, substitute INST for VAR in NS1
        return(NS1[VAR->INST], {(VAR REFERENT_SET)}  $\cup$  CDRS_NS)
    else report no salient referents or failure of uniqueness presupposition

case OP=def: % possible anaphoric reference
    REFERENT_SET = all_accessible_referents(QUD, RESTR1)
    If singleton(REFERENT_SET),
        return(NS1[VAR->INST], {(VAR REFERENT_SET)}  $\cup$  CDRS_NS)
    else if |REFERENT_SET| > 1,
        report failure of uniqueness presupposition
    else % no salient antecedent, retrieve referent from common ground
        DomainRestr = add_domain_restriction(VAR, RESTR1, QUD)
        REFERENT_SET = retrieve_referents(VAR, DomainRestr, CG)
        If singleton(REFERENT_SET),
            return (OP[VAR,DomainRestr,NS1], {(VAR REFERENT_SET)}  $\cup$  CDRS_R  $\cup$  CDRS_NS)
        else if |REFERENT_SET| > 1,
            report failure of uniqueness presupposition
        else % attempt to accommodate later
            return (OP[VAR,DomainRestr,NS1], {(VAR {})}  $\cup$  CDRS_R  $\cup$  CDRS_NS)

case  $\lambda$ :
    DomainRestr = add_domain_restriction(VAR, RESTR1, QUD)
    case non-top-level OR non-question: % non-presuppositional
        return (OP[VAR,DomainRestr,NS1], CDRS_R  $\cup$  CDRS_NS)
    case wh-question: % presupposes some object satisfies DomainRestr
        REFERENT_SET = retrieve_referents(VAR, DomainRestr, CG)
        return (OP[VAR,DomainRestr,NS1], {(Var REFERENT_SET)}  $\cup$  CDRS_R  $\cup$  CDRS_NS)
    case polar-question:
        return (OP[VAR,DomainRestr,NS1], CDRS_NS)

case WH_Ellipsis:
    resolve_WH_Ellipsis(Term, Level)
    % shown in Figure 3

```

Figure 2: resolve_term algorithm

4 Discussion of Examples

In this section, we discuss the examples given in the introduction, and highlight how the presupposition resolution algorithms can be used to resolve pronouns, definites, and quantifiers in general (i.e., reference related presuppositions, under our view) as well as other presuppositional phenomena, such as elliptical questions.² We illustrate the crucial changes which take place to the QUD data structures, allowing effective resolution of referents and presuppositions.

While the Utterance LF (ULF) describes only the literal content of an utterance, the CULF, along with the CDRS, can be thought of as a record of what the utterance really means, in the context in which it is said. For example, the following (ULF, CULF, CDRS) triple illustrates the QDL structure that results from question (2) of Example II (*What dates will the reservation be for?*):

```
(λ{x, date(x), def{y, reservation(y), for.time(y, x)}],
λ{x, date(x), def{y, reservation(y) ∧
    ∃[z, hotel(z) ∧ near(z, MSG), at.loc(y, z)],
    for.time(y, x)}],
{(x:date ?)(y:reservation ?)})
```

Each discourse referent in the set of CDRS is shown in the form (variable:type instance). One fact to keep in mind when viewing the examples is that questions always produce a new QDL entry on top of the QUD stack, and therefore a new CULF and CDRS, while answers may update the CDRS of the current entry on top of the QUD stack, but never produce a new one.

4.1 Pronominal Anaphora: Example I

We will focus on the resolution of the pronoun *it* in the final utterance (8). We claim that at any time there is a set of accessible entities in the discourse, and when a pronoun is used in a discourse felicitously (i.e., as constrained by Relevance), there needs to be a unique maximally salient discourse referent for the pronoun belonging to this set of accessible entities. Under our approach, the set of accessible entities is represented by the union of the CDRS sets of all entries on the QUD stack. Saliency is a partial ordering on this set determined primarily by two factors. First, the members of the CDRS of each entry on the QUD stack are more salient than those for all entries below it on the stack. Second, the relative saliency of discourse referents within the CDRS of a single QDL entry is determined by local constraints, such as those given by centering theory (cf. Grosz, et.al. (1995)), or the theory of focusing

²The careful reader will note that these dialogues contain additional reference resolution problems, such as one-anaphora (example II) and a nonplural antecedent for *they* (example III), etc., not discussed here for brevity.

developed by Suri and McCoy (1994). Our overall approach could be adapted to use any theory of local coherence to determine a partial ordering over the CDRS within a discourse segment corresponding to a single QUD, but it is similar to Suri and McCoy's approach in allowing the CDRS of prior questions to be stacked. Further explanation of how centering constraints can be integrated with our approach is given by Roberts (1998). In our implementation of pronoun resolution (see Figure 2), the function `rank_accessible_referents` gives the partial ordering of the accessible entities from the QUD, filtering out all entities that are incompatible with the agreement features of the pronoun, which are represented in the restriction component of a pronoun term.

In processing this dialogue, the system treats (1) as a question (requests and statements of need and desire should be coerced to questions), and produces (CDRS 1), which is the set of discourse referents mentioned in sentence (1).

```
(CDRS 1)
{(x:person user)(y:hotel ?)(z:date D1)(w:city NYC)}
```

As the system attempts to find out more specific information (imagine that it is filling out a template), it asks subquestions, such as (2), (4), and (6). After each subquestion, a new entry is added on top of the QUD stack, and therefore a new CDRS as well, e.g., the set of discourse referents in the top QUD entry after (2) is (CDRS 2).

```
(CDRS 2) {(w:city NYC)(x:area ?)(y:person user)}
```

When a subquestion is answered, as in (3), the CDRS of the current QUD is updated, e.g., the referent (x:area ?) becomes (x:area Manhattan), and a new referent introduced in the answer is added: (z:area CentralPark). However, once a question is completely answered it is popped off the stack. Thus, after (3) is completely processed as an answer to (2), the stack is popped, and subquestions are also popped after processing (5) and (7). Therefore, when we arrive at (8), the QUD stack is just as it was after (1), since all of the intervening subquestions have been popped. This approach accounts for the observation that more recently mentioned entities, such as *Manhattan* or *Central Park*, are less likely as antecedents for *it* than those from (CDRS 1), which are closer in terms of hierarchical discourse structure.

In order to determine the antecedent for *it*, `rank_accessible_referents` only has to consider (CDRS 1), returning a subset from which (x:person user) is removed, because a person, being animate, does not match the restrictions of *it*. Thus, the search for possible antecedents has been significantly

constrained by using the CDRS associated with the QUD. Among the remaining elements, the most likely antecedent is (y:hotel ?), which we call an *unanchored discourse referent*, since it is not yet bound to an actual instance of a hotel. This might be ranked highest by some versions of centering theory, because it is a direct complement of the verb, while the other referents were introduced by adjunct phrases (*for June 15th* and *in New York*). In general, however, pragmatic plausibility must be considered as an additional filter when determining whether a candidate is a potential antecedent. For example, (z:date D1) can be ruled out because it is not plausible for dates to have swimming pools.

4.2 Definite Descriptions: Examples II–IV

Although definite descriptions can often be identified with antecedents from the CDRS in essentially the same way as pronouns (since the set of CDRS is a subset of the CG Domain), they are not required to corefer with a maximally salient discourse referent. Therefore, our algorithm specifies three ways for a definite reference to be resolved. First, we check whether the CDRS accessible on the QUD stack contains a unique element that matches the restriction of the definite operator. Second, if there is no salient antecedent of the appropriate type, then we attempt to find a unique entity in the CG which satisfies the restriction. Third, if this fails, we use accommodation where possible to introduce an entity from the application's database into the CG.

In example II, we focus on the resolution of the *hotel* in sentence (7). We first look for an appropriate antecedent in the CDRS accessible on the QUD stack, as in our treatment of pronominal anaphora, so we need to trace the stack for this dialogue. A request is made by the user in (1), followed by a series of specific questions generated by the system. The QUD after (1) has the following CDRS:

(CDRS 1) {(x:person user) (y:reservation ?)
(z:hotel ?) (w:place MSG)}

Subquestions are asked in (2) and (4) and answered in (3) and (5), respectively, so the QUD stack is pushed and popped, but at (6), it is at the same state as it was after (1). (6) is interpreted as a request, so a new entry with (CDRS 6) is pushed onto the QUD on top of the QDL for (1).

(CDRS 6)
{(x:person user) (v:conf-room ?) (u:date D4)}

In order to interpret the definite description anaphorically, we search for discourse referents whose type satisfies the explicit *hotel* restriction within the set of all accessible CDRS, viz., the union of CDRS 6 and CDRS 1. Since this set contains exactly one referent (z) which matches the *hotel* type,

the uniqueness presupposition is satisfied and z is selected from CDRS 1 as the antecedent.

It is also possible for a definite description to have no explicit antecedent, as in *the Marriott* in sentence (3) of example IV. In such cases, an empty set of referents will be returned by `all_accessible_referents`, and our algorithm will attempt to retrieve a referent from the common ground. Before resolution, the content of this description is DEF 3, in which the variable ?NS is a placeholder for the unspecified nuclear scope of the *def* operator.

(DEF 3)
def[y, *Hotel*(y) \wedge *Named*(y, *Marriott*), ?NS]

The restriction of this term is obtained from the lexical entry for *Marriott*, which contains the information that it refers to a hotel, in addition to specifying its name. Although we rely on domain-specific knowledge in assuming that it refers to a hotel, we believe this assumption is reasonable, because the proper names for hotels can be automatically acquired from the hotel database used by the application.

Now suppose that there are a number of *Marriotts* in the area. In an empty discourse context, this reference would have an unsatisfied uniqueness presupposition, so the system would need to ask the user which *Marriott* was intended. However, in this case, uniqueness can be established by searching the QUD for an appropriate domain restriction, which can be conjoined with the explicit restriction given in (DEF 3). Since domain restrictions can be contextually supplied for most restricted operators, we interpret (DEF 3) as if there were an additional conjunct, which is schematically represented by `QUD_RESTR(x)` in (DEF 3').³

(DEF 3')
def[y, (*Hotel*(y) \wedge *Named*(y, *Marriott*) \wedge `QUD_RESTR(x)`),
?NS]

As in our treatment of anaphora, the key to constraining the search for an appropriate domain restriction is the QUD structure of the discourse. The entry on top of the QUD corresponds to question (1) of example IV, whose CULF is (simplified):

(CULF 1) $\lambda[x, \textit{Hotel}(x) \wedge \textit{Near}(x, \textit{Airport})],$
 $\exists[y, \textit{Date}(y), \textit{HasVacancyOn}(x, y)]$

To determine whether any implicit domain restriction can be added to *the Marriott*, our algorithm calls `add_domain_restriction` to search the QUD for predicates that match the same basic type as the

³We do not actually include an explicit conjunct for the domain restriction in our implemented logical forms, because an implicit domain restriction may be added to virtually any restricted operator, as motivated by Roberts (1995), and it is of course possible for no new information to be added by domain restriction.

explicit restriction, *Hotel*. In (CULF 1) it finds the restriction $Hotel(x) \wedge Near(x, Airport)$, which can be added in place of the virtual QUD_RESTR(x) conjunct in (DEF 3') to further restrict the domain for *the Marriott*. This restriction (DEF 3'') is then used by `retrieve_referents` to find a matching referent in the CG.

(DEF 3'') $def\{y, Hotel(y) \wedge Named(y, Marriott) \wedge Near(y, Airport), ?NS\}$

It is important to note that the familiarity presupposition for a definite description does not require its referent to be previously mentioned in the discourse. In sentence (1) of Example III, the referent for *the Holiday Inn* does not yet exist in our representation of the common ground, because the system initially has no knowledge that the user is aware of any particular Holiday Inns. In such cases, no objects are returned from the CG by `retrieve_referents`, and the *definite* presuppositional term will remain with an unknown referent in the output of `determine_CULF`. Our approach to accommodation for such unsatisfied presuppositions (in step 2 of `process_utterance`) is to look for a referent in the application's private database of facts about the domain of hotels, since this database represents all of the world knowledge that the system has available. If it finds a unique hotel named Holiday Inn, we can assume this hotel satisfies the user's presupposition. On the other hand, if it turns out that there are either no hotels named Holiday Inn in the database, or multiple Holiday Inns, the system could report the failure of these presuppositions, rather than giving an uninformative simple negative answer to the user's question (1).

4.3 Generalized Domain Restriction: Example III

Consider next the quantificational determiner *every* in sentence (3) of example III. It should be clear that the user is not asking about every morning for all time, but only about all mornings during the planned trip. As with definite descriptions, our algorithm allows the restriction of most operators with semantically contentful restrictions⁴ to be further specified by information from the QUD, so the interpretation of *every morning* will differ depending on whether the dialogue began with question (1a) or (1b). Now, if it is the case that the Holiday Inn has a breakfast buffet on weekdays only, it is important for the system to answer (3) appropriately, as in (4a) and (4b), depending upon the context created by (1a) and (1b).

⁴Domain restriction is not usually applicable to pronouns and other expressions that have little explicit content, because these expressions depend on recovering a salient antecedent in order to determine the type of the referent, rather than searching for a particular type of object in the common ground.

To determine the domain restriction for *every morning*, `add_domain_restriction` searches the QUD for predicates that match the same basic type as the explicit restriction, *morning*. In this case, we take the basic type to be a temporal entity, so it will search for temporal descriptions in the QUD.⁵ By using the QUD stack to constrain the search, *every* will quantify over any temporal entities that are found at a level of discourse structure closest to the current segment, but crucially not over every temporal entity in the entire common ground. Thus, to determine the response in (4a), only the date range mentioned in (1a) is relevant, and a positive response can be given, since the question relates to weekdays. In (4b) however, the date range includes a Saturday, so the system should generate a negative response.

4.4 Elliptical Questions: Example IV

Example IV is a somewhat more complex dialogue, including an elliptical question as well as several definite descriptions. It illustrates how our approach generalizes to the larger class of presuppositional constructions which we identified in the introduction. Let us focus on the interpretation of sentence (3), *How about the Marriott?*, which is assigned the following ULF:

(ULF 3) $Wh_Ellipsis[\phi, Question(\phi), \phi[X \rightarrow (def\{y, Hotel(y) \wedge Named(y, Marriott), ?NS\})]]$

ϕ is a variable referring to some contextually salient question, and the definite description corresponding to *the Marriott* is to be substituted for some term (X) within ϕ . Recall that the variable $?NS$ is a placeholder for the unspecified nuclear scope of the *def* operator.

Our algorithm processes the logical form of an utterance *inside-out*, i.e., the embedded context resolution problems are handled first, so it first resolves the *def* term corresponding to *the Marriott*, as we discussed in §4.2 on definite descriptions, and `add_domain_restriction` produces the refined description (DEF 3''):

(DEF 3'') $def\{y, Hotel(y) \wedge Named(y, Marriott) \wedge Near(y, Airport), ?NS\}$

Next, the top-level *Wh_ellipsis* term in (ULF 3) is resolved, according to the `resolve_WH_Ellipsis` algorithm of Figure 3. ϕ must be a question, so we retrieve the question on top of the QUD stack, and attempt to identify ϕ with its CULF (CULF 1).

(CULF 1) $\lambda\{x, Hotel(x) \wedge Near(x, Airport), \exists\{y, Date(y), HasVacancyOn(x, y)\}\}$

⁵A complete explanation of this situation might require the system to infer the domain goals of the user. However, when the QUD contains some descriptions of the appropriate type, we can use them as an approximate domain restriction, thereby avoiding the computational expense of full plan inference.

```

resolve_WH_Ellipsis(Term, Level)
  %% Assume nuclear scope of Term is of the form:  $\phi[OldExpr \rightarrow NewExpr]$ 
  while QUD stack is not empty {
    QUD-CULF = CULF of QUD[top]
    QUD-CDRS = CDRS of QUD[top]
    (NewExpr, CDRS1) = determine_CULF(NewExpr, Level)
    if NewExpr is a generalized quantifier,
      let SubstLF = match&substitute(QUD-CULF, restriction(NewExpr), NewExpr)
    else (NewExpr is a predicate)
      let SubstLF = match&substitute(QUD-CULF, NewExpr, NewExpr)
    if null(SubstLF)
      or SubstLF is not interpretable as a subquestion of QUD-CULF,
        pop(QUD)
    else return (SubstLF, priority_union(CDRS1, QUD-CDRS))
  }
  %% priority-union(X,Y) is like set union, but when some members of X
  %% and Y have the same type, only the member of X is included in the result.

```

Figure 3: resolve_WH_Ellipsis algorithm

We must now find a term within (CULF 1) for which the term corresponding to *the Marriott* can be substituted. Our match&substitute algorithm looks for terms whose restrictions specialize a common basic type, so it again finds the restriction on the (top-level) λ -term containing the *Hotel* predicate in (CULF 1):

$\lambda\{x, Hotel(x) \wedge Near(x, Airport), \dots\}$

The operator and restriction of this term are replaced by those from (DEF 3ⁿ) and the variables are unified, but the nuclear scope of (DEF 3ⁿ) is unspecified, so the nuclear scope of (CULF 1) remains unchanged in the result:

(3ⁿ) $def\{x, Hotel(x) \wedge Named(y, Marriott) \wedge Near(x, Airport), \exists\{y, Date(y), HasVacancyOn(x, y)\}\}$

(3ⁿ) is (almost) the CULF for *How about the Marriott?*, but it must be noted that it should be interpreted as a polar question, since the λ -term characteristic of a wh-question has been replaced by a definite description.⁶

Thus, both the elliptical question and the domain restriction of the definite description are processed by the same overall strategy: They are interpreted by incorporating information contained in the question under discussion.

5 Conclusions

We have described an integrated approach to resolving presuppositions, which includes pronominal and definite reference resolution. Central to our approach is the maintenance of discourse structures, especially the QUD stack, which captures the hierarchical organization of the discourse. By identify-

ing the presuppositions associated with each kind of construction, and recovering presupposed information from a unified discourse information structure, the search space for relevant contextual information is restricted in a general way.

Realizing the full potential of this approach to discourse structure requires recognizing the question under discussion even when it is not stated explicitly as a question. We have found that requests and statements of need and desire should be coerced to questions, but a general implementation of this coercion process remains to be completed. An even more challenging problem involves inferring the domain goals of the user that are related to the question under discussion. As will be readily apparent to most researchers in NLP, these problems are intertwined with larger AI problems, such as plan recognition, which are beyond the scope of this paper, but must ultimately be solved or approximated in any realistic application. As noted in our discussion of domain restriction (§4.3), it is sometimes possible to extract some explicitly mentioned content from the QUD which is related to the user's domain goals, but we recognize that this is only an approximate solution.

Although we have shown several examples where this approach to discourse structure is successful, a comprehensive empirical evaluation still needs to be performed to determine how frequently particular presupposition problems occur in actual corpora, and to assess what proportion of the actual occurrences are resolved effectively by our algorithms. The first part of this evaluation could be addressed by searching a corpus for expressions that trigger presuppositions, but assessing the actual performance of the algorithms on existing corpora would be more difficult, because it would require the acquisition of background knowledge about other do-

⁶When all top-level λ -terms in a wh-question have been replaced, it is interpreted as a polar question.

mains than the hotel reservation application that we are currently working on.

References

- Lauri Carlson. 1983. *Dialogue Games: An Approach to Discourse Analysis*. Reidel, Dordrecht.
- Barbara Grosz and Candace Sidner. 1986. Attention, intentions, and the structure of discourse. *Computational Linguistics*, 12(3):175-204.
- Barbara J. Grosz, Aravind K. Joshi, and Scott Weinstein. 1995. Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, 21(2):203-225.
- B.J. Grosz. 1981. Focusing and description in natural language dialogues. In B. Webber A. Joshi and I. Sag, editors, *Elements of Discourse Understanding*, pages 84-105. Cambridge University Press, New York.
- Irene Heim. 1982. *On the Semantics of Definite and Indefinite Noun Phrases*. Ph.D. thesis, University of Massachusetts at Amherst.
- Robert M. MacGregor. 1991. Using a description classifier to enhance deductive inference. In *Proceedings of the Seventh IEEE Conference on AI Applications*, pages 141-147.
- Craige Roberts. 1995. Domain restriction in dynamic semantics. In A. Kratzer E. Bach, E. Jelinek and B.H. Partee, editors, *Quantification in Natural Languages*. Kluwer Academic Publishers.
- Craige Roberts. 1996. Information structure in discourse: Towards an integrated formal theory of pragmatics. In Jae-Hak Yoon and Andreas Kathol, editors, *OSU Working Papers in Linguistics, Vol. 49: Papers in Semantics*. (also available via the Internet at <ftp://ling.ohio-state.edu/pub/roberts/infostructure.ps>).
- Craige Roberts. 1998. The place of centering in a general theory of anaphora resolution. In A.K. Joshi M.A. Walker and E.F. Prince, editors, *Centering Theory in Discourse*. Clarendon Press, Oxford.
- Craige Roberts. 1999 (to appear). Uniqueness in definite noun phrases. *Linguistics and Philosophy*. (also available via the Internet at <ftp://ling.ohio-state.edu/pub/roberts/uniqueness.ps>).
- Bertrand Russell. 1905. On denoting. *Mind*, 66:479-493.
- Robert Stalnaker. 1979. Assertion. In Peter Cole, editor, *Syntax and Semantics, Volume 9*.
- Linda Z. Suri and Kathleen F. McCoy. 1994. Raft/rapr and centering: A comparison and discussion of problems related to processing complex sentences. *Computational Linguistics*, 20(2):301-317.