Semantics and Pragmatics of Questions and Answers for Dialogue Agents

David R Traum
University of Southern California
Institute for Creative Technologies
traum@ict.usc.edu

1 introduction

In this paper, we present an implemented approach to semantics and pragmatics of questions, answers, and affiliated dialogue acts used by artificial agents to engage in dialogue with multiple partners (including humans and other autonomous agents).

There have been many accounts of why answers follow questions. On the computational side, these range from enforced structures of system directed dialogue (e.g., with if-then rules, or finite-state networks) that ensure that a system will always provide the answer following a question, or that a system will only understand an answer immediately after its own question, to fully general autonomous agent approaches (e.g., [Cohen and Perrault, 1979, Allen and Perrault, 1980] relying on plan recognition (e.g., to infer a desire for information from observation of the question) and cooperativity (to adopt an intention to provide information given a belief in the desire for such information in the other). While either approach may work for fully cooperative circumstances, neither is fully satisfactory as an account of human behavior toward questions, as pointed out in [Traum and Allen, 1994]. The former is flawed because people do not always answer questions (right away, or sometimes at all). Likewise, the latter has no explanatory power for the behavior of agents who respond to questions even when they are not inclined to be cooperative or intend not to answer. A further problem is that there are multiple different desires that could motivate asking a question (e.g., desire to know whether the addressee knows or desire to put the addressee on record as well as desire for information), and it may be difficult to infer these goals in order to start the reasoning about a response.

Two more sophisticated approaches were implemented by information-based systems in the TRINDI project [Larsson and Traum, 2000, Traum *et al.*, 1999]. The GoDiS system [Bohlin *et al.*, 1999] uses a structure of *Questions Under Discussion* (QUD), based on work by Ginzburg [Ginzburg, 1996] to model question answering. When a question is asked, it gets added to the QUD, which in turn licenses answers to the question (including elliptical short answers). Using this QUD structure overcomes the limitations of the simple structural approaches, since one can have other intervening acts, and multiple questions on QUD. The EDIS system [Matheson *et al.*, 2000], following the theory put forth in [Traum and Allen, 1994, Poesio and Traum, 1998] models the effects of a speaker asking a question as imposing an obligation on the addressee to address the question (with an answer or some other sort of response). This models the pressure for the addressee to respond, but also allows representation of agents who chose not to meet this obligation. It also provides a motivation to answer a question (or perform other addressing actions) without requiring recognition of why the question was asked.[1] For simple information-seeking domains such as Autoroute [Traum *et al.*, 1999], there is little to choose between these two accounts. Both do an adequate job of representing questions, answers, intermediate states, and observation of lack of answers or other responses. We can, however distinguish some properties of QUD and obligations to address questions:[2]

- QUD is a part of the structure of conversations (or discussions) rather than an independent relation between agents. If an agent is engaged in multiple conversations with different participants (e.g., one on the phone, another face to face in the office), a certain question might be under discussion in one conversation but not the other. Likewise a question might be under discussion in a conversation as some agents enter and leave the conversation.

- Obligations are aspects of the social fabric between agents. Even if a conversation ends, an unmet obligation may still persist — one might

[1] This approach is similar in many ways to those of [Allwood, 1994] and [Bunt, 1999]. There are subtle differences with the former, concerning the interaction of obligations to address a question with the grounding of that question. Bunt uses the term *Interactive Pressure* rather than obligations, but ascribes question answering rather to cooperative behavior with respect to the inferred desires of the speaker rather than direct obligations to respond. Unfortunately a more detailed comparison is beyond the scope of this paper.

[2] See also [Larsson, 1998], who made similar points, although with respect to BDI accounts rather than obligations

open a new conversation in order to produce an utterance (such as an answer to a question) to relieve an obligation.

- properties of the local discourse structure of conversations (such as QUD) are needed for understanding small phrases as answers to questions (or in fact in understanding even full assertions as being answers to questions).
- properties of agents (such as obligations) are required to explain the motivation for agents producing answers (or other responses) to questions (whether they are currently under discussion or not).

In the general case, for an agent to understand and participate in dialogue, it must represent both elements of conversation structure (such as QUD) and non-conversation-specific elements of the social fabric (such as obligations to address questions). We use QUD is used as part of conversation structure, licensing short answers and recognizing assertions as answers. Obligations, on the other hand, motivate specific agents to respond.

In this paper, we present an implemented account of questioning and answering behavior of agents in multi-agent, multi-conversation situations, using both QUD and obligations as part of the relevant information state. Throughout, we provide sample representations from the implemented agents that are part of the mission rehearsal exercise project at ICT [Rickel *et al.*, 2002]. In the next section, we present several levels of representation, including levels of *semantics*, *performance* and *interactional*. In Section 3, we describe how this information state changes as agents perform questions and answers.

2 Representation of Aspects of Information State

In this section, we present the basic ontology of information state related to questions and answers. There are several distinct kinds of information. First at a *semantic* level, we have a representation of basic values, and referring expressions to these as well as (partial) propositions and questions. We also represent the *performance* level of speech acts such as an information request, assertion, or answer-act that have semantic level objects as their content. Finally, we represent the *interactional* levels, representing the information state of conversations and the social state between participants.

2.1 Semantic-level

At the semantic level, we represent *individuals*, including entities (people, vehicles, objects), locations, and concepts such as event-types and values. There are also compositional concepts, representing states, events, and relations between individuals. Propositions can be either that a state or relation holds or that an action has happened, is happening, or should happen. In any given world, a proposition may be true, false or valueless.

Questions are represented as a partial proposition (e.g., with some attributes missing from the AVM) paired with a `q-slot` field representing the attribute that is asked about.

One type of relation is that some individual P is an answer to a question Q. In general, propositions that unify with the proposition field of a question but also provide a value for the q-slot field will be answers, but answers are not necessarily limited to these propositions. For instance, an individual may be an answer if it can be combined with the q-slot field and added to the proposition field of the question to form a full proposition. Other objects may also be answers, given a defined procedure to calculate an answering proposition that unifies with the question. We can say that an answer is true if the answering proposition is true.

Semantic contents are also viewed as complete or incomplete. For each event or state, there are a set of necessary parameters. If a proposition is missing any of these, it is incomplete. For a question, if the proposition in the prop slot is incomplete (not counting the parameter indicated by the q-slot) then the question is incomplete.

What is more important for dialogue than the truth of propositions in a given world is the beliefs of agents. Our agents have representations of the current world state, including representations of important propositions as being true, false, or unknown. Agents can have beliefs based on the following sources: (1) the belief was loaded at run-time ("innate" beliefs) (2) the agent inferred the belief from other beliefs (3) the agent perceived the value from the world (simulator), (4) the agent was informed by another agent[3].

Some example representations are shown in (1). 1a-c show types of propositions: a state, an action, and a relation, respectively. 1d shows a partial proposition, the same as 1a, but missing the object-id attribute. 1e shows a question that has this partial proposition as its prop attribute, with object-id as the q-slot.

[3](3) and (4) are really forms of inference, since an agent must choose whether or not to adopt a belief, but they rely on inputs other than beliefs.

(1) a. (C262 ^attribute health-status ^object-id boy ^polarity negative ^time present ^type state ^value healthy)

b. (C216 ^agent sgt ^event secure ^patient aa ^time future ^type action)

c. (A303 ^answer A2 ^question S8290 ^relation answer-to ^type relation)

d. (P7433 ^attribute health-status ^polarity negative ^time present ^type state ^value healthy)

e. (S8290 ^prop P7433 ^q-slot object-id ^type question)

2.2 Performance level

Notions such as concepts, propositions and questions are at the semantic information level and concern abstract relationships of objects and information about the domain of discourse, without any reference to speakers or conversation. We also model the performance of *dialogue acts*, which have these sorts of semantic objects as contents and in turn have effects on the social context. For the purposes of this paper, we consider the set of dialogue acts shown in (2).[4] There are three different classes of act here. Grounding acts (including initiate, repair, request-repair, and acknowledge) manage how material is added to common ground [Traum, 1994]. Core speech acts (CSA), including assert and info-request, introduce new material to the conversation. Backward acts, such as answer and clarify-parameter, point to elements already existing in the conversation. Items shown in <> are variables. Some parameters can take multiple values, as indicated by "*". Dialogue acts are also entities that can be values in complex concepts. E.g., the contents of grounding acts are core speech acts or backward acts.

(2) a. (A1 ^action info-req ^actor <speaker> ^addressee <adr>* ^content <Q> ^type csa)

b. (A2 ^action assert ^actor <speaker> ^addressee <adr>* ^content <P> ^type csa)

c. (A3 ^action answer ^actor <speaker> ^addressee <adr>* ^answer <SA> ^question <Q> ^type backward)

d. (A4 ^action clarify-parameter ^cand <cand>* ^context <SA> ^parameter <role> ^type backward)

e. (A5 ^action repair ^actor <speaker> ^cgu <cgu> ^content <SA>* ^conversation <CON> ^parameter <role> ^type grounding ^value <filler>)

[4] The actual agents use a wider set of actions, see [Traum and Rickel, 2002].

f. (A6 ^action request-repair ^actor <speaker> ^cgu <cgu> ^content <SA2>* ^conversation <CON> ^type grounding)

g. (A7 ^action acknowledge ^actor <speaker> ^cgu <cgu> ^content <SA>* ^conversation <CON> ^type grounding)

h. (A8 ^action initiate ^actor <speaker> ^responder <adr>* ^cgu <cgu> ^content <SA2>* ^conversation <CON>* ^type grounding)

2.3 Interaction Level

There are several elements of interactional context that are related to questions and answers. These include

- Social State – including obligations to act and social commitments to propositions between agents
- Conversations – which have a rich structure including
  - participants
  - turn
  - common ground structure, including common ground units
  - QUD

Participants in a conversation are categorized into active participants (who are speakers and addressees of utterances in the conversation) and overhearers. At most one of the active participants in the conversation may have the turn. Common ground units (CGUs) represent the transition of information from private to common. Each CGU has an initiator, one or more responders, a state, and contents, which include both core speech acts and backward acts, as well as effects such as obligations and commitments. The state represents how this content is moving from private to shared. The initial state represents private knowledge. The final state represents information that is considered grounded. State 1 represents material that has been introduced by the initiator but needs acknowledgement to be considered grounded. State 2 represents material that requires further repair by the initiator as well as acknowledgement by the responder to be grounded. There is also a dead state for material that has been cancelled and will not be part of common ground. In addition to the units themselves, there is a `recent-cgus` stack ordering the active common ground units. See [Traum, 1994] for more details. QUD is one or more semantic question objects that are currently under discussion in the conversation.

3 Dialogue Dynamics

The preceding section described and exemplified the static elements of information state related to questions and answers in conversation, in this section we describe the dynamics: how dialogue acts are recognized as parts of utterances, how the interactional state is updated on the basis of performed dialogue acts, and how agents are motivated to perform acts on the basis of aspects of the interactional state.

3.1 Recognizing Dialogue Acts

In our agents, audio signals are processed by a speech recognizer which produces a stream of words corresponding to an utterance, and a semantic parser produces an interpretations including a sentence mood, an addressee (if indicated with a vocative), and a semantic representation, one of the types in Section 2.1. There are many ways in natural language to refer to individuals, so there is a further step of matching the referring expressions to internal representations of entities. In the simplest case, e.g., names, the referring expressions are replaced with the correct identifier. In the case of more ambiguous expressions, such as definite descriptions or anaphors, a referring expression will fill the slot in the semantic representation. (3) shows a referring expression for "he" in the case where two candidates are currently considered (having both the general properties such as singular and male, and equally accessible from context). When the agent has decided on an anchor for the referring expression that would be present in a ^ref slot.

(3) (R6 ^cand boy ^cand medic ^gender m ^lex he ^num sing ^type ref)

Info-reqs and assertions are recognized by the sentence mood, assuming also appropriate contents (questions and propositions, respectively). Answers are recognized given an assertion in a conversation which unifies with a question in QUD in that conversation and moreover fills in a value for the q-slot. Individual words can also be interpreted as answers, if they fill in a value for a q-slot of a question on QUD. Alternative questions or wh-words can be recognized as clarification-requests, given content in the most recent CGU that would provide context for the question.

`Initiate` acts are recognized when core acts are recognized and there is no ungrounded CGU with the speaker as initiator. A `repair` act is recognized when a CGU is in state 2, and the utterance provides one of the candidates from a clarify-parameter act in the CGU. An `acknowledge` act is recognized when there is a CGU in state 1, and the responder provides a backward act pointing to an act in the cgu. Acknowledgements are also interpreted from certain keywords such as “ok”, “affirmative” “yes” “uh huh’, “roger” and “understood”. Follow-up questions by a responder are also seen as acknowledgements of CGUs with only backward acts. A `request-repair` act can be recognized either by repair elicitors such as “say again”, or when a clarify-parameter action is recognized.

As well as interpretation of human speech, in a multi-character setting agents must also understand the speech of other agents. Rather than requiring speech recognition to be performed here as well, we take some shortcuts, allowing agents to send partially interpretations of their messages to other agents within perceptual range. These messages contain the words that were spoken and may also contain semantic interpretations and dialogue acts as would be interpreted. Even these short-cuts require further processing, since agents will differ in their internal representations of the context.

3.2 Dialogue Act Updates

Once dialogue acts have been recognized, the information state of a dialogue agent must be updated so that this can serve as the context for future interpretations as well as motivations for actions. In this section we outline the effects to the information state described in Section 2.3 for each of the dialogue acts described in Section 2.2. Some effects are universal for all kinds of acts. For example, if there is an obligation in the social state for an agent to perform a type of act, then performance of any act of that type relieves the obligation.[5]

3.2.1 Grounding Act Updates

Grounding acts (including initiate, acknowledge, repair, and request-repair) have three types of effects on the information state: (1) updating the state of the CGU that they are a part of, (2) adding or changing the contents of the CGU, and (3) changing the part of the information state external to the CGU.

An `initiate` act will create a new CGU, adding it to front of the recent-cgus list, and setting its state to 1. Contents of the act will be added to

[5]Things are slightly more complex when grounding is taken into account. “Performance of an act” for core speech acts and backward acts requires that the action has been grounded, meaning participation by the responder/addressee as well as the original speaker. Grounding and lower level acts are assumed to be performed autonomously without requiring feedback from another agent to be assumed complete.

the contents of the CGU. In addition, the act creates an obligation for the responder to perform a grounding act related to the CGU.

A `request-repair` act performed by the responder in state 1 will change the state of the CGU to state 2, meaning that a repair by the initiator will be required, followed by an acknowledgement by the responder for the material to be grounded. As a grounding act, performance by the responder will relieve the obligation to perform a grounding act on this CGU. Moreover, it will introduce an obligation for the initiator to repair. The contents of the act will generally indicate the nature of the problem the responder has with grounding the CGU.

A `repair` act will change some of the content of a CGU. The changes to the state depend on the actor (initiator or responder) and the previous state. If the actor is the initiator, then the resulting state will be 1 (meaning only acknowledgement by responder needed for grounding). This act will also (re-)introduce an obligation for the responder to perform a grounding act. The content of a repair, unlike other grounding acts, will be some part of the content of the CGU, rather than new contents to be added. The `parameter/value` pair of the repair act will be added to this content, perhaps replacing previous values. One `repair` act is required for each parameter that needs updating.

An `acknowledge` act by the responder will move the state of the CGU from 1 to final. It will also transfer all of the effects from acts that are contents of the CGU from the CGU to other parts of the information state.

3.2.2 Core Speech Act Updates

An `info-req` act introduces an obligation for the addressee to address the act. In addition, when an info-req is added to an open CGU in a conversation, the Question content is put on QUD of that conversation. `Assert` acts add commitments of the speaker to their content propositions.

An `answer` act adds commitments to the proposition that their answer parameters answer the question parameter. In addition, grounded answers to a question in QUD will remove the question from QUD.
`Clarify-parameter` acts do not have direct effects, but combined with `request-repair` grounding acts help specify the desired repairs.

3.3 Selection of next acts

When agents are in a conversation, some of the things they consider when forming new utterances include: (1) the turn (2) obligations to ground (3) obligations to repair (4)whether previous utterances have been completely understood (5) (potential) obligations to address an info-request[6] (6)beliefs about true answers to a question

An obligation to ground will lead to an intention either to acknowledge, repair, or request repair, depending on whether the agent thinks it has sufficiently understood the utterance (has values for all necessary parameters). However, intentions to acknowledge are subordinated to intentions to perform backward acts such as answering a question, which will also have a function of acknowledging.

3.4 Generation of dialogue acts

Following the selection of dialogue acts, natural language generation processes produce the words that can be recognized as performing the acts. Generation proceeds in the following phases (1) Content planning (2) sentence planning (3) realization.

In the content planning phase, the agent decides how to perform the selected action(s). In particular, in this phase, the agent decides how to address an info-req, and which proposition to fill in as an answer. While interpreting the input question, the agent calculates all propositions that it thinks are true answers to the question. While this may seem difficult, in practice it is not very complex, since the agent only considers only basic propositions rather than derived propositions that might be constructed using logical operations. For instance, given the question "who is in the landing zone", only propositions of the form "X is in the landing zone", for some entity X that can be in a location will be considered as candidate answers. Of those, only some of the candidates that the agent knows about will it actually think are in the landing zone. Further filtering of the true answers will also take place, based on focus and assumed common knowledge. A novel element in deciding on which true answers to express is the use of the agents emotions. Our agents include an emotional model [Marsella and Gratch, 2002], which is constantly appraising the person-environment relationship for how the goals are affected, and who can be blamed or credited. Some propositions are indicated as "concerns" of the agent, used for coping strategies. We use these concerns as additional filters, so that an agent will prefer to report true answers that are of highest concern. For example, if a conversation is started with a question like "What happened here?", there can be many true answers indicating events that occurred in the present

[6]The obligation is potential if the info-req has not yet been grounded.

location. Some of these may be filtered based on common ground or assumed knowledge of the hearer, but otherwise it may be difficult to prefer one to another and provide natural answers without a lot of fairly involved commonsense reasoning. Using the emotion filter, the agent can report on the one that is causing itself the strongest emotion. While this may not be ideal, in a Gricean cooperative sense, it does often mimic human behavior. In the content selection phase the agent also decides for a clarification-request which parameter to clarify and which candidates are possible to fill in necessary underspecified parameters in a proposition (which might be the content of a question).

In the sentence planning phase, decisions about which forms to use are made, including the decision of whether to provide an elliptical short answer or full declarative sentence, partly on the basis of whether a question on QUD will provide the elided material. The realization phase produces a syntactic tree with the words, which is augmented with gesture indicators and sent to speech synthesis and body rendering. See [Traum *et al.*, 2003] for more details on the generation process.

4 Dialogue Example

As an example, we present a dialogue extract taken from our Mission Rehearsal Bosnia scenario. This extract consists of a part of a single conversation between three characters, a Lieutenant who has just arrived on the scene, his Sergeant, and a Medic, who is hunched over an injured boy. The Lieutenant is a human trainee, the other characters are virtual agents.

(4)

1	L	What happened here?
2	S	There was an accident sir
3	L	Who's hurt?
4	S	The boy and one of our drivers
5	L	How bad is he hurt?
6	S	The driver or the boy?
7	L	The boy
8	S	Tucci?
9	M	The boy has critical injuries
10	S	Understood

This dialogue fragment contains several examples of the dialogue acts described above. utterance 1 is interpreted as an initiate act and an info-req. It places on S an obligation to ground, a potential obligation to address the question, and adds this question to QUD. In 2, S produces the most emotionally charged response, rather than an aggregation of all true answers. 2 is interpreted as an assertion and an answer to the question on QUD. This utterance relieves the obligation to ground and potentially relieves the now actual obligation to address the info-req. 3 is treated as an acknowledgement of the second CGU as well as initiating a third CGU. It also is interpreted as an info-req. The effects are to relieve the obligation to address the first question as well as adding a potential obligation to address this new question. In this case, the Sergeant provides multiple answers,in 4, since there was no filtering condition to prefer one answer over another. Generation aggregates the two and produces a short answer, which is understandable given the question on QUD. The question in 5 is seen as underspecified, since the agent is unable to resolve the pronoun “he” between the two candidates of “boy” and “driver”. This leads to the clarification-request and a request-repair grounding act in 6. After 7, the Sergeant fully understands the question, but in this case, because he assumes that Tucci, the medic, knows better, he redirects the question to the medic. He is able to use a simple vocative in this case, using the question on QUD (which he assumes Tucci also shares, as a participant in the conversation) as the content. 9 is again an answer to the question on QUD, motivated by Tucci's obligation to the sergeant. 10 acknowledges, and removes Tucci's obligation. Since the Lieutenant is also part of the conversation the Sergeant assumes that Tucci's answer relieves his own obligation and does not need to repeat the answer.

5 Discussion

The framework presented above is sufficient to allow agents to engage in question and answer behavior in multi-agent conversations such as the example in the previous section, as well as multi-conversation interactions. In these situations, QUD or obligations alone are not sufficient to handle the range of desired behavior, since one must distinguish local conversation structure from normative behavior and motivation of individual characters. Turn-taking can help agents decide when to answer and when to wait for other agents to speak, but obligations are needed to decide *who* should answer a question. Complexities still arise, however, in terms of when to see a question as finally resolved (and removed from QUD), how to (re-)introduce questions that one is obliged to answer, but are not currently on QUD, when to answer a question on QUD that someone else (or no one) is obliged to answer. Also there is the issue of whether an answer by one agent resolves the obligation for another agent to answer. We assumed positive (assuming co-presence of the obliger), but this is not always correct (e.g., in a classroom setting where a teacher wants a particular student to reply).

Acknowledgements

I would like to thank many colleagues for contributing to this work: the MRE project members for providing a current context to explore these ideas and other crucial parts of the autonomous agents that use these dialogue models, also Massimo Poesio, Jonathan Ginzburg, Staffan Larsson, and Robin Cooper for helping me develop my current views on questions and answers. The work described in this paper was supported by the Department of the Army under contract number DAAD 19-99-D-0046. Any opinions, findings and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the Department of the Army.

References

[Allen and Perrault, 1980] James F. Allen and C. Raymond Perrault. Analyzing intention in utterances. *Artificial Intelligence*, 15(3):143–178, 1980.

[Allwood, 1994] Jens Allwood. Obligations and options in dialogue. *Think Quarterly*, 3:9–18, 1994.

[Bohlin *et al.*, 1999] Peter Bohlin, Robin Cooper, Elisabet Engdahl, and Staffan Larsson. Information states and dialogue move engines. In *Proceedings of the IJCAI99 workshop: Knowledge And Reasoning in Practical Dialogue Systems*, pages 25–31, 1999.

[Bunt, 1999] Harry C. Bunt. Dynamic interpretation and dialogue theory. In Michael M. Taylor, F. Néel, , and Don G. Bouwhuis, editors, *The Structure of Multimodal Dialogue, Volume 2*. John Benjamins, Amsterdam, 1999.

[Cohen and Perrault, 1979] Phillip R. Cohen and C. R. Perrault. Elements of a plan-based theory of speech acts. *Cognitive Science*, 3(3):177–212, 1979.

[Ginzburg, 1996] J. Ginzburg. Interrogatives: Questions, facts and dialogue. In Shalom Lappin, editor, *The Handbook of Contemporary Semantic Theory*. Blackwell, Oxford, 1996.

[Larsson and Traum, 2000] Staffan Larsson and David Traum. Information state and dialogue management in the TRINDI dialogue move engine toolkit. *Natural Language Engineering*, 6:323–340, September 2000. Special Issue on Spoken Language Dialogue System Engineering.

[Larsson, 1998] Staffan Larsson. Questions under discussion and dialogue moves. In *Proceedings of TWLT13/Twendial '98: Formal Semantics and Pragmatics of Dialogue*, 1998.

[Marsella and Gratch, 2002] Stacy Marsella and Jonathan Gratch. A step towards irrationality: using emotion to change belief. In *In proceedings of AAMAS 2002: First International Joint Conference on Autonomous Agents and Multi-Agent Systems*, June 2002.

[Matheson *et al.*, 2000] Colin Matheson, Massimo Poesio, and David Traum. Modelling grounding and discourse obligations using update rules. In *Proceedings of the First Conference of the North American Chapter of the Association for Computational Linguistics*, 2000.

[Poesio and Traum, 1998] Massimo Poesio and David R. Traum. Towards an axiomatization of dialogue acts. In *Proceedings of Twendial'98, 13th Twente Workshop on Language Technology: Formal Semantics and Pragmatics of Dialogue*, 1998.

[Rickel *et al.*, 2002] Jeff Rickel, Stacy Marsella, Jonathan Gratch, Randall Hill, David Traum, and William Swartout. Toward a new generation of virtual humans for interactive experiences. *IEEE Intelligent Systems*, 17, 2002.

[Traum and Allen, 1994] David R. Traum and James F. Allen. Discourse obligations in dialogue processing. In *Proceedings of the 32^{nd} Annual Meeting of the Association for Computational Linguistics*, pages 1–8, 1994.

[Traum and Rickel, 2002] David R. Traum and Jeff Rickel. Embodied agents for multi-party dialogue in immersive virtual worlds. In *Proceedings of the first International Joint conference on Autonomous Agents and Multiagent systems*, pages 766–773, 2002.

[Traum *et al.*, 1999] David Traum, Johan Bos, Robin Cooper, Staffan Larsson, Ian Lewin, Colin Matheson, and Massimo Poesio. A model of dialogue moves and information state revision. Technical Report Deliverable D2.1, Trindi, 1999.

[Traum *et al.*, 2003] David Traum, Michael Fleischman, and Eduard Hovy. Nl generation for virtual humans in a complex social environment. In *Working Notes AAAI Spring Symposium on Natural Language Generation in Spoken and Written Dialogue*, March 2003.

[Traum, 1994] David R. Traum. *A Computational Theory of Grounding in Natural Language Conversation.* PhD thesis, Department of Computer Science, University of Rochester, 1994. Also available as TR 545, Department of Computer Science, University of Rochester.