A Scheme for Annotating Problem Solving Actions In Dialogue

Teresa Sikorski and **James F. Allen**
Department of Computer Science
University of Rochester
Rochester, New York 14627-0226
sikorski@cs.rochester.edu
james@cs.rochester.edu

Abstract

Other than speech acts occurring at the utterance level, there exist higher-level dialogue actions such as those involving the coordination of problem solving activities between dialogue participants. This paper describes a taxonomy of actions at this problem solving level, and a scheme for annotating collected corpora to capture these problem solving behaviors. We describe a methodology for performing the annotation and include results of applying these tags to the TRAINS-93 corpus.

Introduction

The goal of spoken dialogue systems is to provide a natural and flexible interface for human users. To that end, computational linguists collect corpora of dialogues between human users and annotate interesting features of the interactions. Dialogue system designers are then able to make observations about patterns of language use, evaluate the effectiveness of different strategies employed by dialogue participants, and can more effectively model human interaction in human-computer dialogue systems. Dialogue annotations can provide data with which to train statistical models, and can be used as standard by which dialogue system performance can be evaluated.

In 1996 and 1997, the Multiparty Discourse Group of the Discourse Research Initiative (DRI) held meetings in which they developed an annotation scheme appropriate for collaborative problem solving dialogues. The DAMSL framework for annotating dialogues, which is based on the DRI design, provides tags with which various characteristics of utterances within a dialogue can be annotated (Allen and Core, 1997). The DAMSL annotation scheme identifies the following orthogonal aspects of utterances:

- **Communicative Function** corresponds closely to traditional speech acts but allows a single utterance to be tagged with multiple acts. For example, in response to the question "Which train should we use?", the utterance "There is a train at Avon." should be tagged as an assertion, a suggestion, an offer, and an answer to a question.
- **Information Level** characterizes an utterance as either addressing the communication process (Communication Management), addressing the problem solving process (Task Management), or performing the task (Task).

The annotation scheme proposed in this paper provides tags for an aspect of dialogue that is orthogonal to the aspects of dialogue characterized within the DAMSL framework: the problem solving level. Task-oriented dialogues can be viewed as having a distinct dimension involving the coordination of problem solving between participants. This includes most domains studied in the computational literature, such as dialogues to build transportation plans for execution later as in TRAINS, dialogues to make decisions about how to furnish a room, dialogues to obtain information and plan academic schedules, and dialogues to coordinate the participants as they interact to perform some task such as assembling a water pump.

The problem solving level captures critical information that must be identified in computational systems if they are to be successful participants in dialogues. For instance, a system will perform very different tasks if it thinks it is negotiating what the goals are, rather than establishing a solution for already agreed upon goals, or establishing the current state of the world as a preliminary evaluation of what problems need to be tackled to achieve the goals. Problems arising from confusing discussion of goals with discussion of solutions already causes problems in the TRAINS-95 system implementation, even in its simplest domain of route planning (Sikorski and Allen, 1996). As tasks become more complex, these issues become more critical. In addition to gaining insight into how people interact using language to solve problems, annotations of problem solving acts in dialogue may be helpful for tracking topic shifts and performing intention analysis.

To demonstrate that the problem solving level is orthogonal to the traditional speech act level, we specify a speech act followed by a set of utterances that all perform the specified speech act, but perform different problem solving acts:

Assert

Establish Goal	I have to ship two boxcars of bananas and two boxcars of oranges to Dansville
Assess Situation	That would take a total of three hours
Establish Solution	We could leave the boxcars in Dansville while engine one goes on to Avon
Evaluate Solution	That will put us an hour over for the bananas

Action-Directive

Establish Goal	Let's work on getting the oranges to Dansville first
Establish Solution	Send engine three to Corning
Evaluate Solution	Let me see if this will work

Info-Request

Establish Goal	What time did the oranges need to be there
Assess Situation	Can two boxcars be hooked to one engine
Establish Solution	Do you do you want to load the oranges into the boxcars
Evaluate Solution	Does that solve the problem

Likewise, we demonstrate that the problem solving level is orthogonal to DAMSL's Information Level by specifying each of the possible Information Level values followed by a set of utterances that are at the specified Information Level, but perform different problem solving acts:

Task Level

Establish Goal	I have to ship two boxcars of bananas to Dansville
Assess Situation	Where are the oranges
Establish Solution	Send engine two with two boxcars to Corning to pick up oranges
Evaluate Solution	Does that get us to Dansville by the deadline

Task Management Level

Establish Goal	Let's work on getting the oranges to Dansville first
Assess Situation	Will I need any other information
Establish Solution	Shall we do that
Evaluate Solution	I added wrong

Communication-Management

Establish Goal	I didn't understand what we are supposed to do
Assess Situation	Could you repeat the location of the trains
Establish Solution	Which train did you say to use
Evaluate Solution	Let me check it over

The problem solving level has been studied in previous work. Litman and Allen argued for using multiple levels of plan analysis to capture the intentions in dialogue (Litman and Allen, 1987). Their "discourse" level of analysis involved problem solving actions such as INTRODUCE-PLAN, CONTINUE-PLAN, IDENTIFY-PARAMETER, and so on. Lambert and Carberry use a similar level of analysis and added a third level to capture true "discourse" level actions (Lambert and Carberry, 1991). This earlier work differs from our current effort in two principal ways. The first difference is the level of analysis. The earlier work assigned specific acts to individual utterances whereas we are characterizing the purposes of stretches (or segments) of discourse similar to the discourse segment purposes of Grosz and Sidner (Grosz and Sidner, 1986). To obtain something at the level of an utterance-by-utterance analysis, we would have to combine our scheme with additional characterizations of the utterance intentions such as those in the DAMSL framework. The second difference is that we are attempting to design a comprehensive taxonomy that can cover full corpora of dialogue, and desire a scheme that can be used by human annotators with high reliability.

Annotation Scheme

The problem solving action refers to the underlying purpose of a segment of dialogue with respect to the development of a shared or joint solution to a problem. This is a solution that the participants develop together and requires agreement from each about what to do. Note that this restricts our scheme to collaborative dialogues. A few definitions of terms used in subsequent sections are given below.

A *goal* is a description of a situation or action that the plan is intended to achieve. It is the reason for the dialogue, or a subsection of the dialogue, and the participants will tend to continue to work on finding a way to achieve the goal until they are both satisfied with the solution, or at least one of the participants explicitly abandons the goal.

A *solution* is a sequence of actions or a set of variable assignments that is introduced in order to achieve some goal. Solutions are relevant only by virtue of the goals they achieve, and if the participants abandon a goal, then they should also abandon the solution for the goal. In contrast, if the participants abandon a solution then this does not imply that they should abandon the goal.

Goals may define an *evaluation criteria* by which different solutions can be compared. For instance, if a goal is to get a train to Avon as soon as possible, then plans that get a train to Avon can be compared on the basis of how quickly the train is expected to arrive.

With these definitions, we can now introduce the different problem solving actions that can be accomplished by a segment of dialogue:

- Establish Goal
- Assess Situation
- Establish Solution
- Evaluate Solution
- Execute Solution

Sets of utterances, contributed by different speakers, many times form a segment of dialogue that performs a single problem solving act. For example, there may be grounding behavior, assertions, acknowledgments and question/answer pairs that combine to perform a problem solving action. For this reason, problem solving acts will usually be tagged at a segment level rather than at the individual utterance level.

Note that our scheme does not require each utterance to belong to exactly one segment tagged with a single problem solving action. Some utterances may not be involved in any problem solving activity, while others may contribute to multiple problem solving actions. In the latter case, we would create overlapping segments, and tag each individually.

The following sections contain excerpts from the TRAINS-93 corpus, which is a set of over 90 dialogues in a cargo shipping domain (Heeman and Allen, 1995). In these dialogue fragments, a **u** preceding an utterance indicates that it was spoken by the person assuming the role of a human manager (**u**ser), whereas an **s** indicates that it was spoken by a person assuming the role of a computer assistant (**s**ystem). Overlapping speech is indicated by putting the overlapped words within square brackets with the right bracket followed by a parenthesized index. Words within brackets having the same index were spoken simultaneously. Silence within an utterance is indicated by <sil>.

Establish Goal

A test for determining whether a set of utterances should be tagged as Establish Goal is to decide whether they address the question of *what needs to be achieved.* Segments belonging to this category are those in which the participants:

- specify a goal
- establish constraints or filters for valid solutions
- state preferences or evaluation criteria for solutions

A segment establishes a goal if the end result of the segment is that the participants have agreed on a goal. Such a segment might be as simple as one participant explicitly stating the goal and the other implicitly accepting it by starting to work on it when they get the turn. In the following fragment, the user states the goal and then the system acknowledges the user's utterance, indicating that the goal has been agreed to.

```
u:  I need to get as many boxcars of oranges as I
    can to Bath by seven AM
s:  Okay
```

In other cases, the goal might be developed through an extended segment including clarifications, corrections, and modifications by both participants, as in the following:

```
u:  okay <sil> um <sil> <noise> I have to get
    three boxcars of bananas <sil> to Bath <sil>
    and two tankers of orange juice to Dansville
s:  okay so three boxcars of bananas to [](1) Bath
    <sil> and <sil> two
u:  [ Bath ](1)
u:  [ tankers ](2) of orange juice
s:  [ and ](2)
s:  yep
u:  to Dansville
s:  okay
u:  a- by <sil> twelve noon
s:  okay
```

Assess Situation

The test for determining whether a set of utterances should be tagged as Assess Situation is whether the segment discusses questions of any of the following forms:

- *Are we able to do x?*

- *What is true about the world?*
- *What is true of the world assuming the solution under discussion is carried out?*

Segments to be tagged as Assess Situation include those that establish capabilities and the state of the world. In these segments, the plan remains unchanged, but the dialogue participants may be exploring the plan and determining what the implications of the plan are.

Segments that establish capabilities involve establishing information about constraints on how and when actions can be performed by different participants in order to facilitate the the decision-making process. This can be as simple as one participant telling another some information that might be useful, as in

```
s:  it takes an hour to load them <sil> just so
    you know
```

Or the segment might involve establishing information about how actions might interact, as in the following segment that establishes that two trains can run at the same time:

```
s:  okay th- the <sil> engine that you took from
    Avon you can be doing that at the same time
    as you're taking the one from Elmira
```

Situation assessments can be explicit questions about the characteristics of the domain. Such questions will occur more frequently in dialogues involving novice users.

```
u:  can two engines travel on a track simultaneously
s:  um <sil> only if you connect them together
u:  okay
```

Segments that assess the situation may also establish certain facts about the world, or about the expected changes in state as the proposed course of action occurs.

```
s:  okay so we'll have them loaded by seven a.m.
    and then to Bath makes four more hours <sil>
    so we'll get to Bath at <sil> eleven a.m.
u:  eleven a.m.
u:  okay
```

A common purpose of segments in this class is to find out information about the initial state, as in:

```
u:  where are the engines
s:  the engines are at Avon, Bath and Corning
```

Establish Solution

A test for determining whether a set of utterances should be tagged as Establish Solution is to decide whether they address the question of *how the goals are to be achieved.* Segments belonging to this category are those in which the participants propose components to a solution or resources to use to accomplish a task. If successful, such segments result in the participants agreeing to include that proposed constituent in the solution.

Often, such segments consist of a single action describing a series of solution components, as in:

```
u:  okay <sil> I'm gonna take engine three then
    from Elmira and I'm going to take it to Bath
u:  and <sil> get the two boxcars <sil> and then
    I'm gonna take them back up to Corning
u:  for <sil> and then load them with oranges
    there
```

At other times, a component of a solution might be specified by negotiation and discussion as in:

```
u:  take engine number two from Elmira
s:  yep
u:  with <sil> two [ boxcars ](1)
s:  [ okay ](1)
u:  go to Corning
s:  yep
u:  t- to load up the [ oranges ](2)
s:  and did you also want to hook up the two
    tankers
u:  yes
s:  good <sil> yeah
s:  and then you want to go to Elmira to make the
    orange juice
u:  right
```

Evaluate Solution

A test for determining whether a set of utterances should be tagged as Evaluate Solution is to decide whether they address the questions like *"Does this solve the problem?"* or *"How well does this solve the problem?"*. By performing this communicative action, the participants evaluate whether a proposed course of action satisfies a goal. Such actions appear after a course of action has been proposed, and may point out a problem with the proposed solution or address whether the proposed solution is missing some necessary component.

Segments that evaluate the solution include those in which one participant gives a direct opinion of the proposed solution, as in:

```
s:  that'll probably work
```

At other times, the segment might evaluate a solution with respect to implicit goals in the domain, such as a goal not to have trains interfere with each other, as is seen in the following segment:

```
s:  okay wait a minute let me make sure you're not
    gonna run into anybody along the line here
u:  okay
s:  um
s:  yeah if you did that you would run into one of
    the engines which is coming from Elmira
```

Segments may also contain contributions from both participants in evaluating the solution, as in the segment

s: okay so that'll take <sil> two hours to go to Corning so it'll be eight a.m. and one more hour to go to Dansville so that'll be nine a.m.
u: okay so we met times for both deliveries right
s: um yep
u: okay
s: good

Many times, after a solution to a problem has been proposed, the participants iterate through components of the solution explicitly checking that all constraints have been met and ensuring that no misunderstandings have occurred. In the sample dialogue, there was a time constraint that gets checked. The two segments below correspond to the evaluations that occurred after courses of action had been developed to satisfy the two subgoals that were implicitly defined by the top-level conjunctive goal.

s: okay so we'll have them loaded by seven a.m. and then to Bath makes four more hours <sil> so we'll get to Bath at <sil> eleven a.m.
u: eleven a.m.
u: okay

Execute Solution

In the TRAINS domain, the task is to develop plans that are to be executed later. For this reason, we don't encounter utterances that actually execute a solution in the TRAINS corpus. We include the Execute Solution tag for domains in which execution of the solution involves the performance of some dialogue act. For example, Execute Solution acts will occur in Information Retrieval domains when an Inform act provides a system user with requested information.

Untagged Utterances

Utterances that are abandoned or unintelligible are not tagged with a problem solving act. Also excluded from the problem solving annotation scheme are utterances that perform purely communicative functions, such as turn-holding and turn-grabbing, and utterances that do not pertain to the task at hand (e.g. non sequiturs).

Segments with Multiple Tags

It is common for a segment to both assesses the situation and evaluate the solution with respect to that situation assessment. The following dialogue fragment is an example of such a segment:

s: let's see we're still <sil> still got a problem here I <sil> think <sil> um um it should be <sil> another <sil> oh <sil> oh I see it uh I <sil> it only <sil> takes one hour to load <sil> um <sil> any number of boxcars
u: oh
u: so total
s: yeah
u: okay
u: so maybe that would work <sil> because we were only off by an hour
s: right

When specifying a task goal, there is often some situation assessment that is especially relevant to the goals or constraints on possible solutions. Sometimes the act of establishing the goal and the act of assessing the situation occur in the same segment, as in the following example:

u: I have to ship a boxcar of oranges to Bath by eight a.m. <sil> and it's midnight <sil>
s: okay

Very frequently solutions are developed when one participant asks a question that tests the feasibility of a step, and the step is implicitly adopted if it is feasible. Such exchanges should be tagged as both Assess Situation and Establish Solution. In the following dialogue segment, one participant is asking questions that at a surface level simply assess the situation. Reading through the subsequent dialogue, however, we find that the negotiated interpretation of the segment is both a situation assessment and a proposal for a certain course of action.

u: can I attach two boxcars <sil> from Dansville to engine three
s: yeah
u: and how long would it <sil> take <sil> to get <sil> to <sil> the Ava- Avon
s: uh three hours

Assess Situation and Establish Solution also cooccur in the following dialogue fragment where a step is explicitly proposed and is then implicitly rejected by an utterance that informs the user of some system capability that renders the step infeasible. The utterance that proposes the step is tagged only as Establish Solution. The systems response is tagged as both Assess Situation and Establish Solution.

u: have engine two carry the five boxcars of oranges to Bath
s: an engine can only carry three loaded boxcars
u: oh okay

In the following dialogue fragment, the user responds to the system's question about whether the proposed solution meets time constraints by giving more information about the constraints, which is an Establish Goal act, and by implying that the proposed solution is an acceptable solution, which is an Evaluate Solution act.

```
u: um <sil> actually there's no time requirement for
   the orange juice so um <sil> I can take my <sil>
   pretty time
s: okay
```

We encountered rare instances in the TRAINS-93 corpus where a segment performed both an Establish Solution act and an Evaluate Solution act:

```
s: or we could send engine E <sil> three <sil> I mean
   s- start it before the one with the boxcars <sil>
   I mean would i- <sil> would <sil> because the
   oranges are getting there in plenty of time
u: okay
```

Often subgoals will be introduced that address both *what* needs to be done (and are therefore tagged as Establish Goal) and introduce steps of a solution (and are therefore tagged as Establish Solution). An example of such a segment is given below:

```
s: <sil> um well <sil> we also need to make the orange
   juice <sil> so we need to get [ oranges <sil> to
   Elmira ](1)
u: [ oh we need to pick up ](1) oranges
u: oh [ okay ](2)
s: [ yeah ](2)
u: alright
```

Annotation Methodology

Dialogues were stored in separate files and had been preprocessed to partition them into numbered utterance units with the speaker of each utterance indicated.

Our procedure for training annotators was to have them read the manual describing the annotation scheme and then have them tag the same dialogue independently. When the annotators finished tagging the training dialogue, a meeting was held to review the results and to rectify misunderstandings. The manual was updated to clarify points of confusion.

After training was complete, each dialogue was assigned to two annotators who made a first pass at the annotations independently. Meetings to discuss the annotations were held for each dialogue, and the pair of annotators produced a reconciled version of the annotated dialogue.

Results

Two annotators having approximately the same level of expertise each tagged thirteen dialogues (1111 utterances) from the TRAINS-93 corpus with the problem solving actions described in this paper. After completing each dialogue, the annotators met to compare their tags, and to create a reconciled version of the annotated dialogue. When necessary, the annotators established policies on how to deal with certain problematic dialogue phenomena. These policies were then added to the annotation manual, and were applied when annotating subsequent dialogues.

Category	Actual Number	Num in Dispute	Kappa Score
A	193	32	.87
E	56	32	.70
G	55	19	.80
N	79	23	.83
S	117	49	.72

Table 1: "Partial Credit" Reliability Scores

Category	Actual Number	Num in Dispute	Kappa Score
A	374	161	.68
E	55	28	.73
G	65	33	.73
N	155	59	.78
S	327	78	.83
A+E	35	33	.51
A+G	3	8	-.34
A+S	64	105	.13
E+G	7	9	.35
E+S	2	2	.50
G+S	24	13	.72

Table 2: "All-or-nothing" Reliability Scores

Tables 1 and 2 show the results of our evaluation of the reliability with which are annotators are able to use our coding scheme. Our evaluation uses the kappa coefficient of agreement, as proposed by Carletta (Carletta, 1996). The kappa statistic applies to classification tasks where the categories being judged are independent. Unfortunately, in our coding scheme that is not strictly the case. For each utterance in the dialogue, decisions are made for each category as to whether the utterance belongs to that category. Table 1 scores agreement for each basic category depending on whether the utterance was tagged by both annotators as either belonging to that category or not. For example, if one annotator tagged an utterance as belonging to both Assess Situation and Establish Solution, and the other annotator had tagged the utterance as only Establish Solution, then an error would only be levied against the Assess Situation category. Table 2 uses much more stringent scoring in which errors are assessed whenever the entire set of tags for an utterance is not identical between annotators. In the previous example, the scores in Table 2 reflect errors levied against both the Assess Situation category, and the category representing the set containing Assess Situation and Establish Solution.

The first column of the tables indicates the problem solving act, where A=Assess Situation, E=Evaluate Solution, G=Establish Goal, S=Establish Solution and N=Not Included. In Table 2 we also have entries for all the sets of acts that were encountered in the annotated dialogues. The second column of the table indicates the number of times the action (or combination of actions) was tagged in the reconciled versions of the annotated dialogues. The third column indicates the number of times one annotator had tagged an utterance as belonging to that category, but the other annotator did not. The fourth column gives the kappa score.

The reported kappa score was calculated as follows:

$$K = \frac{(PA-PE)}{(1-PE)}$$

where

$PA = \frac{(Total-Errors)}{Total}$

$PE = PC^2 + (1-PC)^2$

$PC = \frac{ActualNumber}{Total}$

$Total$ = Total number of utterances in tagged corpus

$Errors$ = Number of utterances where the tag was disputed (Column 3 of the Tables)

$ActualNumber$ = Number of utterances on which the tag appears in the reconciled corpus (Column 2 of the Tables)

The kappa scores in Table 1 show that three of the five categories can be tagged reliably according to Carletta's standards (kappa >.80). The scores for the remaining two categories indicate reliability levels that allow for tentative conclusions to be drawn (kappa >.67). In Table 2, we find acceptable reliability for the five basic categories, but unacceptable scores for most of the categories that combine tags. The scores reported in Table 2, however, use the more more stringent calculation that gives no credit for utterances where the annotators agreed on one of the tags. For this reason, Table 1 gives a more accurate estimation of the reliability of our annotation scheme whereas Table 2 gives a more revealing diagnosis of the points of confusion in the scheme.

Acknowledgments

This work was supported in part by National Science Foundation grants IRI-9528998 and IRI-9623665. Thanks to Mark Core for his comments on a draft of this paper.

References

James Allen and Mark Core, "DAMSL: Dialog Annotation Markup in Several Layers," Available from http://www.cs.rochester.edu/research/trains/ annotation, Draft 1997.

Jean Carletta, "Assessing Agreement on Classification Tasks: The Kappa Statistic" *Computational Linguistics*, 22(2), 1996.

Barbara Grosz and Candace Sidner, "Attention, Intentions, and the Structure of Discourse" *Computational Linguistics*, 12(3), 1986.

Peter Heeman and James Allen, "The TRAINS-93 Dialogues," TRAINS Technical Note 94-2, Department of Computer Science, University of Rochester, 1995.

Lynne Lambert and Sandra Carberry, "A Tripartite Plan-based Model of Dialogue," In *Proceedings of the Twenty-ninth Annual Meeting of the Association for Computational Linguistics*, 1991.

Diane Litman and James Allen, "A Plan Recognition Model for Subdialogues in Conversation," *Cognitive Science*, 11, 1987.

Teresa Sikorski and James Allen, "The TRAINS-95 System Evaluation," TRAINS Technical Note 96-3, Department of Computer Science, University of Rochester, 1996.