Annotating Continuous Understanding in a Multimodal Dialogue Corpus.

Carlos Gómez Gallo[†], Gregory Aist[◦], James Allen[†], Will de Beaumont[⋆]
Sergio Coria[‡], Whitney Gegg-Harrison[◇], Joana Paulo Pardal[∗], Mary Swift[†]

Department of Computer Science, University of Rochester, Rochester, NY, USA[†]
Department of Brain and Cognitive Science, University of Rochester, Rochester, NY, USA[◇]
Institute for Human and Machine Cognition, Pensacola, FL, USA[⋆]
Computer Science and Engineering Department, Arizona State University, Tempe, AZ, USA[◦]
Spoken Language Systems Lab, Lisbon, Portugal[∗]
Universidad Autónoma de México, Mexico City[‡]
cgomez@cs.rochester.edu

Abstract

We describe an annotation scheme aimed at capturing continuous understanding behavior in a multimodal dialogue corpus involving referential description tasks. By using multilayer annotation at the word level as opposed to sentence level, we can better understand the role of continuous understanding in dialogue. To this end, we annotate referring expressions, spatial relations, and speech acts at the earliest word that clarifies the speaker's intentions. Word-level annotation allows us to trace how referential expressions and actions are understood incrementally. Our corpus has intertwined language and actions which help identify the relationships between language usage, intention recognition, and contextual changes which in turn can be used to develop conversational agents that understand language in a continuous manner.

1 Introduction

In this paper we describe an annotation scheme aimed at capturing continuous understanding interaction in the Fruit Carts corpus (Aist et al., 2006). This corpus is a collection of multimodal dialogue interaction between two humans, where the first (the speaker) gives spoken language instructions to the second (the actor), who responds by manipulating objects in a graphical interface. The Fruit Carts domain was designed to elicit referring expressions from the director that are ambiguous in various ways, including prepositional phrase attachment and definiteness. The point at which the responder resolves the ambiguity can be observed through their actions in response to the spoken instructions. While the long-term goal of this corpus collection is to model incremental language processing in a spoken dialogue system, in this paper we concentrate on the highly interactive nature of the human dialogue in the corpus and how to represent it in an annotation scheme.

Our annotation scheme for these interactions is centered around the idea of marking the roles, referential expressions, spatial relations in the speaker's speech acts at the word level, as soon as they can be unambiguously identified. This contrasts with traditional utterance-level annotation, since our scheme requires us to break acts down into smaller constituents labeled at the word level.

Previous research in psycholinguistics has shown that continuous understanding plays a major role in language understanding by humans e.g., (Tanenhaus et al., 1995; Altmann and Kamide, 1999; Traxler et al., 1997). Various researchers have proposed software methods for continuous understanding of natural language adapting a wide variety of techniques including finite state machines (Ait-Mokhtar and Chanod, 1997), perceptrons (Collins and Roark, 2004), neural networks (Jain and Waibel, 1990), categorial grammar (Milward, 1992), tree-adjoining grammar (Poller, 1994), and chart parsing (Wiren, 1989). Recently, dialogue agent architectures have been improved by different strategies that adhere to continuous understanding processing (Stoness et al., 2004; Aist et al., 2006). Therefore the work we present here will be a great help to understanding relationships between language and action, and the further development of dialogue agents.

2 The Data

The Fruit Carts experiments involve referential description tasks in which the speaker is given a map showing a specific configuration of fruits and geometric shapes in different regions (see map on upper middle panel in Figure 1). The speaker's task is to instruct the actor to reorganize the objects so the final state of the world matches the map first given. The speaker gives spontaneous spoken instructions to the actor on how to go about manipulating the objects. The actor responds to the instructions by moving the objects, but does not speak. As a result the corpus captures a two way human-human dialogue. Thus we have a complex interaction of language and real world actions through a visual and auditory interface.

The Fruit Carts domain was devised in order to facilitate the study of continuous understanding of natural language by machines. As such, it contains various points of disambiguation based on factors including object size, color, shape, and decoration; presence or absence of a landmark; and phonetic similarity of geographically close regions of the map (e.g., "Morningside" and "Morningside Heights" are close together.) For example, the objects were designed such that describing the entire shape required a complex description rather than a prenominal modifier. For example, a square with stripes could also be referred to as "the stripey square", but a square with diamonds on the corner cannot be referred to as *"the corner-diamonded square". We thus chose a set of shapes such as "a small square with a diamond on the edge", "a large triangle with a star on the corner", "a small triangle with a circle on the edge", and so forth. Table 1 shows an excerpt of a dialogue in the corpus.

The five basic operations in the Fruit Carts domain are choosing, placing, painting, rotating an object. The order in which these operations are performed is up to the speaker and the actor. All of the operations are fully reversible in the domain. For example, an object can be returned to the default color (black) by painting it black. This eliminates the need to handle "undo" which is in general a substantial complicating factor for dialogue systems.

The following excerpt illustrates the interaction between the speaker's commands and the actor's actions. Full sentences take several interactions to complete and there is a combination of visual and auditive interaction. When the speaker utters a command, the actor executes it as soon as he/she has gathered enough information about what to do. During execution, the speaker may give feedback by confirming, correcting, or elaborating as he/she feels appropiate.

SPK> In Morningside there needs to be a triangle with a star on its hypotenuse
ACTR> (*actor moves triangle*)
SPK> Right there and then it needs to be rotated um
ACTR> (*actor waits*)
SPK> to the left
ACTR> (*actor rotates triangle*)
SPK> keep going
ACTR> (*actor keeps rotating*)
SPK> right there
ACTR> (*actor stops*)

Table 1: Example of a Fruit Carts Dialogue

The corpus consists of digital videos of 104 dialogues. Each of the 13 participants, recruited from the university community, directed the actor, played by a human, in 8 different referential description tasks. Each of these task scenarios ranged from 4 to 8 minutes in duration. The number of utterances in each scenario ranges from 20 to more than 100. There are approximately 4000 utterances total in the corpus, with an average length of 11 words per utterance.

We are basing our scheme on well developed speech act tagging hierarchies such as DAMSL (Core and Allen, 1997) and DIME-DAMSL (Pineda et al., 2006). There is a limited amount of previous work related to the current paper. One example is Reitter and Stede (Reitter and Stede, 2003) which discusses markup allowing for underspecification of the meaning of contributions, but the work in their paper was done at a sentence-by-sentence level or higher (vs. at a word-by-word level in the current paper.) Some authors use the term *incremental annotation* to refer to the human-computer interaction process of successively annotating the same text with additional details (Molla, 2001), (van Halteren, 1998). This process is related to our work in that not all of the text is annotated at the same time. They focus on multiple passes over the same text, while we focus on a left-to-right continuous annotation done (in principle) in a single pass.

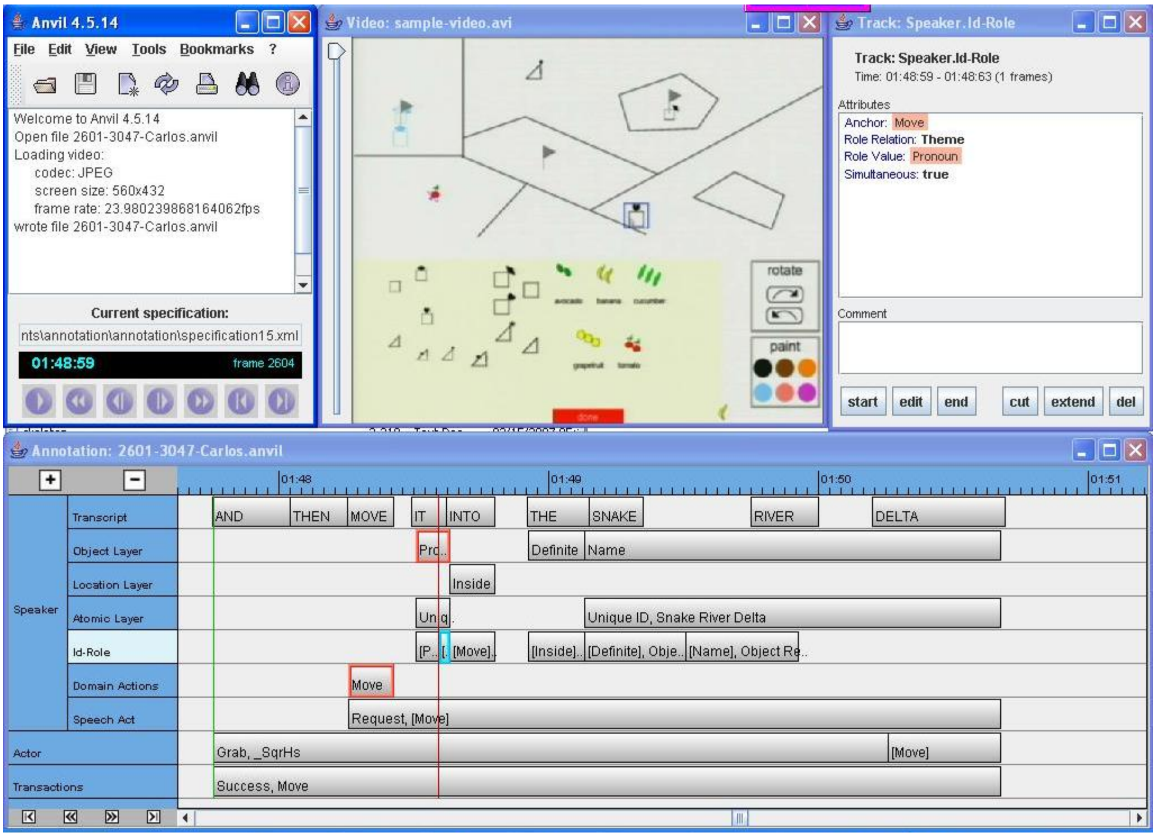

Figure 1: Annotation of utterance "*and then move it to the snake river delta*"

3 The Tool

Since the corpus we are using has speech and visual modalities on top of speech transcripts, we chose the annotation tool Anvil (Kipp, 2004) for its capabilities to show all these modes concurrently. To familiarize the reader on how Anvil works and how our scheme represent continuous understanding, consider a simple annotation for the utterance "and then move it to the snake river delta". In this sentence there are two referring expressions (i.e. "it" and "snake river delta"), and a Move action annotated in the object and domain action layer respectively (see Figure 1).

The multilayer annotation will be described in detail in the following sections. For now, simply notice the four panels Anvil uses (Figure 1). The lower panel contains the transcript and labels all time aligned with the playable video. The upper middle panel shows the video for a particular session. The upper right panel contains the attributes and the attributes values of the highlighted green box in the Id-role Layer. The upper left panel provides the play, stop, forward buttons to control the playing video.

In the utterance depicted in Figure 1, the speaker is requesting the actor to *Move* a *Square with a Diamond on the Side* to a region called *Snake River Delta*. On the Object Layer we can see the two main entries corresponding to the referring expressions in the utterances (i.e. pronoun "it" and name "snake river delta"). One layer down, the Location Layer, specifies the spatial relation, namely that speaker wants the object (i.e. theme) to be *inside of* the *Snake River Delta* region. The Id-role Layer identifies the "it" as the instance of the theme role and "into the snake river delta" as the instance of the location role, both of the Move action.

Figure 1 shows two links by highlighting the boxes with cyan and orange colors. The green box on the Id-role Layer identifies the Spatial Relation *Inside* (orange box) as an instance of the Location role of the Move action (orange box) on the Domain Action layer.

The Speech Act Layer contains the Request act performed by the speaker which links to the domain action Move (link is not shown in this picture). On the Actor Layer, there is a label for the action of holding the previously introduced (and selected) object without moving it. The actor then proceeds to move it to the target region as it is interpreted. The Transaction Layer shows the committed actions between the speaker and actor finished successfully. In the next section, we explain each of these layers in detail.

4 The Scheme

Important characteristics of our scheme include the fact that we annotate the speaker's intentions. This implies that even when certain domain actions, objects or locations are not fully specified in the speech input, the annotation includes the necessary information to execute the commands. For example if speaker says "an avocado in central park", we construct a Move action even though the verb or command to trigger the action was omitted.

Marking up labels at the word level has a strong implication as well. We are constructing an incremental interpretation of actions, referential expressions, and spatial relationships. Traditionally speech acts have been the smallest unit of annotation. However, in this project we break them down into finer constituents. For instance, with referring expressions, we annotate the object attributes (e.g., size, color, decoration) and break down actions into their semantic roles (e.g., theme, location, angle).

We now present the four principles guiding our annotation scheme in Table 2. Though it is certainly possible and useful to mark labels at the phoneme level, we chose the word level for annotation as a good approximation to incremental annotation as principle 1 states. Principle 2 is applied by reducing speech acts to their minimal units. In our scheme we have object anchors, locations, relation types, core action labels, and all arguments types (e.g., color, angle).

To ensure incremental annotation, labels should be marked exactly at the point where they become unambigious. The appropiate place to do this is at the point when enough information has been gathered to know the label semantics. Also, even though the transcript contains future sentences, they should not be used for labelling as principle 3 describes. Last, when the speaker uses vocabulary outside the domain, as principle 4 states, we annotate the intended meaning of the word. For instance the speaker may say "tomato" or "apple" both to refer to the same object, or use "move" or "put" both to refer to the same action.

1. Annotation is done at the word level (e.g., not the phonological or sentence level).
2. Annotation is done in minimal semantic increments (e.g., identifying anchors, relation types, arguments).
3. Semantic content is marked at the point it is disambiguated without looking ahead.
4. Reference is annotated according to speaker's intention.

Table 2: Principles of Annotation.

.

To exemplify how the annotation principles work, let us examine the annotation of a simple NP "The small box in Morningside" in Table 3. The first word that the annotator considers, "the", introduces a noun phrase. However, we do not yet know the type, color, or size of the object. At this point, the annotator can only introduce an anchor for the object. Later in the speech, the annotator will label object features and link them back to the anchor. In this manner, principle 1 is followed by having the anchor be aligned to the word "the". Principle 2 is observed when the minimal unit at this point is simply the anchor. In order to follow principle 3, object features are not annotated by using later information (i.e. linking to an entity in the upcoming stream by looking ahead in the transcript or video).

In time step 2, the word "small" is under consideration. The word elaborates one feature of the object which is introduced with anchor A1. The annotator marks the role type (e.g., *size*), role value (e.g., *small*), and role anchor (e.g., *A1*). At time step 3, the object type is introduced by identifying the role type and value in relation to the anchor *A1*. However, the word "box" was marked as *square* in order to follow principle 4.

5 Description of Domain Actions

Speaker can request the actor to perform a certain actions on an object or objects. Domain objects can be selected, moved, rotated, and painted. In addition to these, there are actions that involve mouse movement. For example a Grab action requires the actor to point to an object, select it, and yet do not move it. Table 4 shows some of the actions in the Fruit Carts domain along with their

Time	Word	Annotation
1	"The"	anchor(A1), definite(A1)
2	"small"	size(A1, small)
3	"box"	objectType(A1, square)
4	"in"	Anchor(A2), spatialRelation(A2, inside), location(A1,A2)
5	"morningside"	anchor(A3), Name(A3), ObjectReferent(A3,MorningsideRegion3), Ground(A2, A3)

Table 3: Detail annotation of "The small box in morningside"

semantic roles.

Action	Semantic Roles
Select	obj
Move	obj, location, distance, heading
Rotate	obj, angular distance, heading
Paint	obj, color

Table 4: Actions in the Fruit Carts Domain.

6 Annotation Layers

Speaker utters actions to be performed, domain objects, locations in the map, distances, etc, while the actor is executing as response to these utterances. Speaker may then correct, refine, reject or accept such executions. To annotate this rich amount of information we developed six layers of annotation that convey the dialogue underway focusing on the incremental interpretations of both referential expressions and actions. These layers are Object, Location, Atomic, Speaker, Actor, and Transaction layer.

The first three layers encode values for the action semantic roles. In this way noun phrases (Object Layer), spatial relations (Location Layer) and atomic values (Atomic Layer) are ready for the second three layers to refer to. Our annotation scheme makes use of bottom three layers (see Figure 1), Speaker, Actor and Transaction, to encode the dialogue occurring between the speaker language and the actor execution.

6.1 Object Layer

The first layer of annotation is the Object Layer. An object is fully described when its type (e.g., triangle, square, flag, etc), size (e.g., small, big), color (e.g., blue), location (link to Location Layer entry), decoration type (e.g., heart, diamond), and decoration location (e.g., corner, side) attributes are all instantiated. Our approach is to annotate NP's incrementally by identifying an anchor to which each object attribute is linked. The first word of an NP will be marked as an anchor (usually "the" or "a". To relate attributes to the anchor we use a construct named *Id-role* in order to provide an exact trace of incremental interpretations.

[*Id-role*]: Id-role is a speech act that identifies a particular relationship (the role) between an object (the anchor) and an attribute (the value). It is used for incrementally defining the content of referring expression and action descriptions

Table 5: Annotation of incremental interpretations with Id-role.

Anchor labels are assinged semantic roles of object features. Anchor types include pronouns, definites, indefinites, names, demonstratives, etc. If speaker uses a pronoun, an anchor of type *pronoun* will be marked. Then the role ObjectReferent assigns the domain unique-id to the anchor. If on the other hand, the speaker uses a complex NP such as that one in example 3, an anchor is entered at the first word (e.g.,, "the", "a"). All other object features are marked and linked to the anchor as they are elaborated by the speaker.

For example, the NP "the triangle with a star on the hypotenuse" has an anchor at "the" of type definite. At the time we hear the word "triangle" we do not know certain semantic roles such as decoration type (whose value is "star") nor the decoration location (whose value is "hypotenuse"). Furthermore, even though the speaker is thinking of a particular object, an hence using a definite article, it is not clear if they are referring a small or big triangle.

To evidence this ambiguity and annotate incrementally we mark the anchor which will then be elaborated by identifying role values in later in the speech. Another type of referring expressions consists of a group of objects over which an action is distributed, as in "Paint all objects blue". The annotation of this example follows from the construction of Id-role which can have a list of values instantiating a role. Thus we would link all objects to the theme role of Paint action.

This annotation scheme is quite versatile allowing any objects with partial descriptions be annotated. The interpretation trace of NP will play an important role in seeing how the actor execution of actions are triggered suggesting how early in the command the actor can disambiguate information.

6.2 Location Layer

Entries in this layer encode of location descriptions for objects (e.g., "the box in morningside"), spatial relation of objects or the semantic role of Move and Rotate actions. Spatial relations contains three attributes a Relation, Relation Modifier and Ground. A relation can be the following: *inside of, on top of, right of, left of,* and others. The Ground attribute links to an entry in the Object Layer which serves as frame of reference. Thus an entry in this layer is equivalent to the expression *RELATION (x, ground)* where x is the object holding the relation with the ground. The Relation Modifier has three values, *a little more*, *a little less* and *touching*. The modifier handles cases where speaker gives commands incrementally as in "keep going" or "a little more" making heavy use of ellipsis constructions and is particularly used in refinement of the Location semantic role.

As example of this layer, consider the phrase "into the snake river delta" in Figure 1. Here we want to express that the final location of the object in the Move action. Since "snake river delta" is a referring expression, it is used as the ground for the spatial relation that the object "it" refers should be *inside of* the region *Snake River Delta*. The label entry in the Location Layer is similar to the expression *INSIDE(x, Snake_River_DeltaRegion5)* where x is the object whose location we are describing. The Id-role entry identifies the Location Layer entry as the instance of the location semantic role for the Move action (see Figure 1).

Another utterance from data is the following: "In Morningside Heights, there needs to be a triangle with a star on its hypotenuse". Notice that the location of the Move action is specified first, before any other argument of the action. Even that we are dealing with a Move action does not follow directly from the copula verb. In other examples such as "the color of the square is blue" also shows that the underlying action is not always evident from the verb choice, but rather the argument types.

Our scheme handles these cases nicely due to the versatility of the id-role constructions. For instance, at the time the phrase "In Morningside Heights" is uttered we can not be certain that the speaker is intending a Move action. Thus we are unable to mark it as a location semantic role. Such label only happens at a point after the copula verb when the object "a triangle" is specified.

Nevertheless a spatial relation can still be constructed before the location role. The word "in" can be marked as both an anchor for a spatial expression (in the same fashion as NP), and also a *inside of* spatial relation with "Morningside Heights" as *ground*.

6.3 Atomic Layer

The Atomic Layer contains any of the domain colors, numbers, and the two sizes (small, big) of objects. These are atomic values, as opposed to complex values (i.e. spatial relation) and hence Atomic Layer. These values instantiate distance, color, and size roles respectively.

As an example, if the speaker utters "rotate it 30 degrees", we can create an entry for number 30 on this layer. Then the Id-role triplets will relate this number as the angle semantic role for the Rotate action in the Domain Action Layer.

6.4 Speech Act

In this section we describe the Id-role, Domain Action and Speech Act layers. Given that objects, spatial relations and atomic values have been introduced, we can now identify what role these entries have in the action underway using the Id-role construct. Much in the same way of referential expressions, incremental interpretation is an important principle by which we annotate speaker's actions.

The Id-role construct which has been described in section 6.1 lays in the Id-role Layer (see Figure 1). Same as before the Id-role is a triplet that links the semantic roles to its respective value in any of the first three layers (Object, Location or Atomic). Different from before the anchor will not be an *object* being incrementally interpreted but rather an *action* being incrementally interpreted.

The following sublayer describes the domain actions the speaker can request. These have been explained in section 5. Next sublayer contains speech acts performed by the speaker. These, described in Table 6, include apologize, accept, reject, correct, apology, and others. In this section we are going to focus on the Refine action which is particular to our scheme.

Accept	Speaker can accept or confirm an action performed by the actor.
Request	Speaker can request the actor to perform any of the domain actions.
Correct	A Correct action can be divided into two: a self-correct (speaker) or actor-correct. Such action includes the new information that is being corrected.
Refine	Speaker wants to refine part of the information already given for another action previously stated.

Table 6: Speaker's Speech Acts

We are addressing data that shows incremental instructions to the actor. This occurs greatly due to the complex dialogue between speaker and actor that interleaves language and execution. Since speakers see that the actor is interpreting and executing their commands, they feel free to adjust parameters of their actions. Therefore utterances such as "a little bit more" after a move or rotate command are common (see dialogue 1).

These utterances present elliptical constructions where object, verb and even location are omitted. Usually these sentence will specify arguments given in previous utterances. Notice that the new utterance, either a "a little bit lower" or "keep going" are not contradictory with the previous actions. It is rather an elaboration or refinement of a previous semantic role value (or argument value) of the action. Thus to properly address these types of sentences we have develop an act called Refine that reissues the previous command and refines one of the action arguments. If the new piece of information were contradictory with the already stated actions, the speaker would be uttering a Correct speech act.

6.5 Actor Layer

This layer records the actor's part of the dialogue. It contains any of the domain actions (e.g., select, move) and any of their possible roles (e.g., object, color, distance). Here we take into account mouse pointing, movements, picking up objects without moving, and releasing objects.

6.6 Transaction Layer

The last layer of annotation is called Transaction Layer (see Figure 1). It summarizes the speaker-actor interaction by providing the state of the world at the end of all objects manipulations. The Transaction gives us information of what commitments the speaker-actor agree on and whether such commitments finish successfully or unsuccessfully.

At the moment we do not have overlapping transactions. This means that one has to finish before another one starts. Therefore transactions usually contain one domain action with possibly many other speech acts of correction, refinement, rejection, etc. Even though it is certainly possible to have an unfulfilled commitment before acquiring new ones, our current scheme does not allow that.

An utterance such as "move the square to the right and paint it blue" could be thought of a single commitment involving two actions or two overlapping commitments where the first one not yet fullfilled before the second one occurs.

7 Evaluation

An exploratory annotation exercise was performed by two individuals working independently on a same dialogue fragment in order to produce two annotation data sets. Although the annotators were not intensively trained for the task, they were provided with general guidelines.

The inter-annotator agreement, computed as simple percentage and not as kappa statistics (Carletta, 1996), was highest, between 80% and 96%, for labels such as Object Type, Action, Size, Distance, Spatial Relation Modifier, Color, Speech Act and Transaction. Lowest agreement, between 15% and 51%, occurred at labels such as Role Anchors, Role Values, and Speech Act Contents.

These results can be explained as follows: 1) simple values such as color or action types are reliably annotated, well above chance since annotators are choosing from a set of options of around 10 items. 2) linking values that require annotators link to other labels (i.e. linking to different anchors). Since the annotators have not been intensively trained, we are developing a manual annotators can access on line to clarify these issues. Also the annotation scheme is still on a definition and refinement stage and some tagging conventions might be required. This agreement evaluation must be interpreted as a diagnosis tool and not as a final measurement of the scheme reliability. Discrepancies in annotation will be analyzed and discussed to refine the rules and it is expected that the agreement increases when using future, improved, versions of the scheme.

8 Future Directions

Since referential expressions, spatial relations and speech acts are annotated at the word level as opposed to the sentence level, we have rich information about when objects are brought into discourse, commands are issued by the speaker, actor actions occur, and the state of the world at the end of each transaction. This level of detail allows us to look closely at the relation between actor action and speaker's utterances.

This annotation will allow researchers to: Evaluate continuous understanding capabilities of conversational agents. Develop an intention recognition module that can identify action roles to interpret speech input so that a conversation agent can perform such actions. Identify the minimum set of action roles which are required for action recognition. Identify features that correlate a linguistic structure with a particular action role. Identify a typical structure of action roles that help recognize which action is underway. Identify action role based on most likely action to follow. Find features that would predict when a transaction is successful and when it is not.

References

G. Aist, J. Allen, E. Campana, L. Galescu, C. Gómez-Gallo, S. Stoness, M. Swift, and M. Tanenhaus. 2006. Software architectures for incremental understanding of human speech. In *Interspeech*.

S. Ait-Mokhtar and J.-P. Chanod. 1997. Incremental finite-state parsing. In *Proc. of the 5th Conf. on Applied Nat. Lang. Proc.*

G. Altmann and Y. Kamide. 1999. Incremental interpretation at verbs: Restricting the domain of subsequent reference. *Cognition*, 73:247–264.

J. Carletta. 1996. Assessing agreement on classification tasks: the kappa statistic. *Comp. Ling.*, 22(2):249–254.

M. Collins and B. Roark. 2004. Incremental parsing with the perceptron algorithm. In *Proc. Conf. of ACL*.

M. Core and J. Allen. 1997. Coding dialogues with the DAMSL annotation scheme. In D. Traum, editor, *Working Notes: AAAI Fall Symposium on Communicative Action in Humans and Machines*.

A. Jain and A. Waibel. 1990. Incremental parsing by modular recurrent connectionist networks. In *Proc. of the 2nd Conf. on Advances in Neural Information Proc. (NIPS)*.

M. Kipp. 2004. *Gesture Generation by Imitation - From Human Behavior to Computer Character Animation*. Ph.D. thesis, Saarland University.

D. Milward. 1992. Dynamics, dependency grammar and incremental interpretation. In *Proc. of the 14th Conf. on Comp. Ling.*

D. Molla. 2001. Towards incremental semantic annotation. In *Proc. of the 1st MMA*.

L. Pineda, H. Castellanos, S. Coria, V. Estrada, F. López, I. López, I. Meza, I. Moreno, P. Pérez, and C. Rodríguez. 2006. Balancing transactions in practical dialogues. In *CICLing, LNCS*. Springer Verlag.

P. Poller. 1994. Incremental parsing with ld/tlp-tags. *Computational Intelligence*, 10(4).

D. Reitter and M. Stede. 2003. Step by step: underspecified markup in incremental rhetorical analysis. In *Proc. of 4th LINC at EACL*, Budapest.

S. Stoness, J. Tetreault, and J. Allen. 2004. Incremental parsing with reference interaction. In *ACL Workshop on Incremental Parsing*.

M. Tanenhaus, M. Spivey-Knowlton, K. Eberhard, and J. Sedivy. 1995. Integration of visual and linguistic information in spoken language comprehension. *Science*, 268.

M. Traxler, M. Bybee, and M. Pickering. 1997. Influence of connectives on language comprehension: Eye- tracking evidence for incremental interpretation. *The Quarterly Journal of Experimental Psychology: Section A*, 50(3).

Hans van Halteren. 1998. The feasibility of incremental linguistic annotation. *Computers and the humanities*, 32(5).

M. Wiren. 1989. Interactive incremental chart parsing. In *Proc. of the 4th meeting of EACL*.