# Building Common Ground and Interacting through Natural Language

**Arthi Murugesan[1], Wende K. Frost[2], Derek Brock[2], and Dennis Perzanowski[2]**

[1]NRC/NRL Postdoctoral Fellow

[2]Naval Research Laboratory, 4555 Overlook Ave, S. W.,

Washington, DC 20375 USA

{Arthi.Murugesan.ctr, Wende.Frost, Derek.Brock, Dennis.Perzanowski}@nrl navy mil

**Abstract**

Natural language is a uniquely convenient means of communication due to, among its other properties, its flexibility and its openness to interpretation. These properties of natural language are largely made possible by its heavy dependence on context and common ground. Drawing on elements of Clark's account of language use, we view natural language interactions as a coordination problem involving agents who work together to convey and thus coordinate their interaction goals.

In the modeling work presented here, a sequence of interrelated modules developed in the Polyscheme cognitive architecture is used to implement several stages of reasoning the user of a simple video application would expect an addressee—ultimately, the application—to work through, if the interaction goal was to locate a scene they had previously viewed together.

## Introduction

Natural language can be viewed as a collaborative means for expressing and understanding intentions by using a body of widely shared conventions. The challenge of conveying an intention from one agent to another, in this case from a speaker to an addressee, can be characterized as a coordination problem that participants must work together to solve. People rely on a procedural convention for collaborating with each other (Clark 1996) that can be summarized as follows: in posing a coordination problem for an addressee to solve, the speaker is expected to construct the problem so that the effort needed to work out the intended solution is minimized. Doing this entails the use of a system of straight-forward practices that include: 1) making the focus of the coordination problem explicit or salient, 2) posing a problem one expects the addressee will be able to solve and 3) framing the problem in a manner that makes it easy for the addressee to solve. All three steps require the speaker to take into account the common ground he/she shares with the addressee.

In the modeling work presented here, a sequence of interrelated modules is developed with the Polyscheme cognitive architecture (Cassimatis 2006). These modules simulate the stages of reasoning a video application might execute, if the goal were to find a frame or scene in the video that the user and the application had both seen at an earlier time. The user's verbal description of the scene is treated as a coordination problem that the application must try to solve. Accordingly, each word and the higher-order semantics of the description—its conceptual references to objects, places, and events—are matched against a body of domain-specific lexical and schematic representations held by the application. Each stage may result in a failure prompting either the application or the user to initiate repairs. In the case of a successful interaction, the application is able to infer which scene among those the user has previously inspected is the one the user intends for the application to find.

## Natural Language Interactions as Joint Actions

Natural language can be viewed as a collaborative joint action with the aim of expressing and interpreting intentions (Allen and Perrault 1980). Clark (1996) characterizes the challenge of conveying an intention from one agent to another—for example, from a speaker to an addressee—as a coordination problem that participants must work together to solve. To get to the intended solution, or a solution that is the best possible given the

other parameters, individuals routinely proceed in a conventional collaborative way. In particular, they rely on certain heuristic presumptions regarding a set of actions they expect to carry out together, which includes posing and grasping the problem and working out and acting on the result. Instantiations of a few of these presumptions are modeled in this work from the point of view of the addressee. They are 1) salience—the words and actions the speaker uses are expected to make identification of the intention behind them obvious to the addressee and 2) solvability—the speaker is expected to have an intended result in mind and to have framed the expression of the intention so the addressee can readily "solve" or work out what the intended result must be and act on it.

## Application Setup and Tools Used

The heuristics cited above are implemented to work with a simple application that is loosely based on an experimental test bed known as InterTrack (Pless et al. 2009) used for research on advanced traffic monitoring and user-interaction techniques. In this variant of InterTrack, a scene of interest is "shared" with the application by clicking somewhere in a video frame. The resulting world information derived from the video at that time is recorded and is then referred to as a "card." Automated identification of objects and events in shared scenes has not been implemented, so what the application "knows" about a specific scene is currently coded by hand. Once a shared card has been created, the user can then access it at a later time with a written sentence.

Computational implementation of natural language interactions is a complex undertaking that requires both cognitive modeling and linguistic tools. The Polyscheme cognitive architecture (Cassimatis 2006) is used in the present effort because of its role as the substrate for recent modeling work in sentence comprehension (Murugesan and Cassimatis 2006) and intention-based reasoning (Bello and Cassimatis 2006), both of which are needed to model linguistic communication as a collaborative activity. Head-driven phrase structure grammar (HPSG) (Sag, Wasow, and Bender 2003) is the syntactic theory used in the modeling work because of its lexical integration of syntax and semantic constraints and the computational advantages of its framework.

## Modeling Expectations in Language Use

The models described in this section are conceived as a set of interrelated modules. Reasoning about the user's sentence input is performed at different levels in stages that roughly correspond to the two heuristics outlined above. The salience heuristic is applied to the utterance level of the user's sentence, while the solvability heuristic is modeled in two stages 1) the linguistic implications and 2) the practical implications. Both salience and solvability assume common ground—it is expected that the speaker has taken into account the knowledge believed to be shared with the addressee as a basis for the words and actions that are used to convey the intention.

### Common Ground

Common ground is roughly defined as mutual, common or joint knowledge, beliefs and suppositions between two people (Clark 1996). In this paper we do not attempt to model common ground in its true complexity or entirety; instead, we make two simplifying assumptions which allow for future extensions: first, we assume that the application interacts with only one speaker at all times and second, we only model the effects that common ground has on the application's interpretation of a sentence. We do not attempt to understand the influence that common ground has on the speaker's generation of language.

In this simplified cognitive model, the application's knowledge is limited to the vehicular traffic domain– to words, concepts, common sense knowledge and experiences shared with the user. Example words listed in the model's lexicon include such words as grammatical determiners *the* and *a*; pronouns, such as *you* and *me*; descriptors or adjectives, such as *red, white* and *black*; common nouns, such as *car, truck, street* and *intersection*; and domain-specific verbs, such as *show, pass, passing, stalled* and *stopped*. Each of these words has an HPSG feature structure composed of both syntactic and semantic elements, which are encoded within the Polyscheme model, figure 1 shows an example.

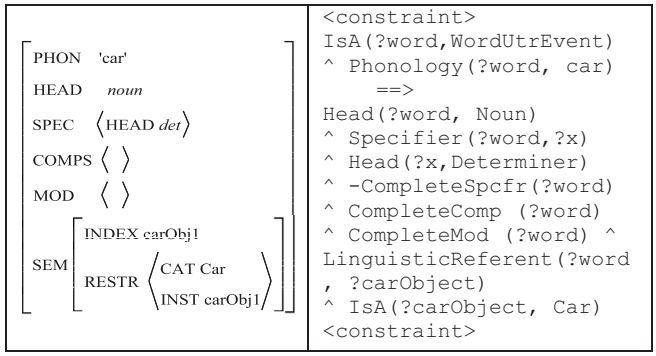


**Figure 1.** The HPSG feature structure of the word "car" is shown on the left. On the right is its Polyscheme constraint in XML. Propositions on the right of the arrow ==> (consequents) are inferred from propositions on the left (antecedents). Note, "?" used as prefix denotes a variable.

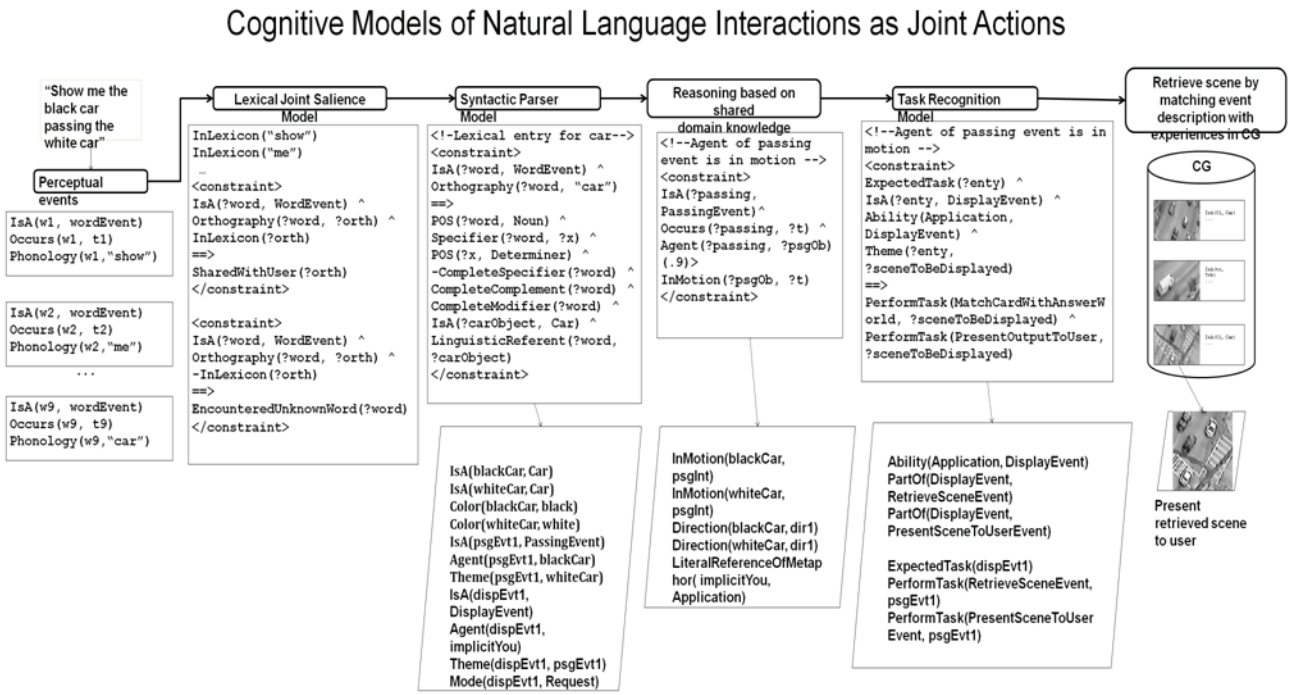


**Figure 2** shows the interaction of the different Polyscheme models that represent the stages involved in processing a natural language interaction. Sample output of processing the sentence "Show me the black car passing the white car" is also shown.

In this application, the person referred to as "you" by the speaker is always interpreted as the application, or the "addressee." Also, certain domain knowledge is employed, such as specifying that the agent doing a passing event is the object that is in motion. These are all represented as Polyscheme constraints in the appropriate models.

Common ground accrues as the interaction between the speaker and the application progresses. One of the most straightforward additions to the application's common ground is when the speaker clicks on a scene of interest in the video and the corresponding "card" is added to the application's internal knowledge. There are several other instances where common ground between the application and the speaker is further accumulated, as in the case of repairs, shown in the latter sections of this paper.

## Salience

Clark's principle of salience suggests, roughly, that the ideal solution to a coordination problem is one that is most prominent between the agents with respect to their common ground. Thus, for example, when the user enters "…the red car…," it is expected that these words are intended to make objects tied to this phrase more prominent than other objects in the knowledge and experiences the user shares with the addressee. In Polyscheme, attention is brought explicitly to a perceptual event by placing it as an input to the temporal perception specialist. The temporal perception specialist then utilizes Polyscheme's internal mechanisms (such as the focus of attention and attraction cues) to arrive at the most salient solution, which in Polyscheme is technically termed "the best answer world".

It is to be noted that when the user enters a word that the application does not know, for e.g. "… the ted car …" due to a typo of 't' instead of 'r,' the model recognizes that it is unable to identify the user's intention because the word "ted" is not in the common ground shared by the user and the application (see figure 3). The model responds by showing the user a message saying "I do not recognize the word ted."

```
<constraint>
IsA(?word, WordUtteranceEvent) ^
Orthography(?word, ?orth) ^
-IsA(?orth, LexicalEntry)
==>
EncounteredUnknownWord(?word) ^
-InSharedLexiconWithUser(?orth)
</constraint>
```

**Figure 3** shows a sample constraint from the model that identifies an unknown word.

The user now has the option of recovering by either rephrasing the utterance with words known to the system,

or in the case of advanced users, adding the specific unknown word and its syntactic, semantic and common sense implications to the common ground.

## Solvability

The first stage in solving the natural language utterance involves parsing it, forming its semantic interpretation and combining the semantic knowledge with relevant world knowledge in the common ground. In the second stage, the listener reasons further to identify the intention or goal behind the speaker's actions, the actions in this case being the speaker's words.

### Natural Language Understanding

Although an addressee's syntactic, semantic, and pragmatic processing may overlap in real-world collaborations, these levels are currently staged in separate models as shown in figure 2.

**Syntactic Parsing Model**

The output from the cognitive model that captures joint salience, which consists of words of the speaker's input the sentence marked in the appropriate temporal order, acts as the input for the first solvability model, which handles syntax. The syntax model assigns the lexical entries appropriate to these words (including probabilistic assignments for words with multiple senses or lexical entries), and invokes a HPSG parser to create the output tree structure. We have implemented a very basic HPSG parser in the current version of Polyscheme. This parser module can be replaced by existing state of the art HPSG parsers such as PET (Callmeier 2000) and Enju (Yusuke and Junichi 2005).

The HPSG output structure of a successfully parsed sentence includes the semantic components that build the sentence. The output box at the bottom of the Syntactic Parser Model in figure 2 shows the semantic components obtained as the result of parsing the sentence "Show me the black car passing the white car." The objects mentioned in the semantics include *blackCar*, *whiteCar, psgEvt1 and dispEvt1*.

Sentence processing may at times terminate abruptly due to any of several causes for failure, the most common being an inability to form a valid parse of the sentence. On failure, the model reports an error message to the user and requests the speaker to initiate a repair by using a simpler or more grammatically correct sentence.

**Reasoning based on shared domain knowledge**

As mentioned earlier, common ground also includes common sense knowledge and domain knowledge shared by the application and the speaker. Some examples of shared common sense knowledge are 1) the concept that the person the speaker refers to by "you" is the application and 2) The agent who is passing an object is in motion. An example of traffic domain specific knowledge is the concept that cars passing each other generally (with some probability higher than 50%) refers to a car overtaking another car going in the same direction. This information is encoded as Polyscheme constraints. The box under "reasoning based on shared domain knowledge" in figure 2 shows an example of a common sense constraint encoded in Polyscheme.

The semantic components of the parser output act as the input for the common sense and domain knowledge reasoning model. The model is able to infer that the *blackCar* must be in motion and that the *whiteCar* is possibly in motion as well and that they may both be travelling in the same direction, *dir1*. The output of the reasoning model is shown in the bottom output box under "reasoning based on shared domain knowledge" in figure 2.

The process of understanding the semantics of a sentence within the context of the domain knowledge may also result in inconsistencies. For example, when the semantic meaning of "the stalled car passed the truck" is combined with the domain knowledge that stalled objects are not in motion but that agents passing other objects must be in motion, it results in contradictory input as to whether or not the car is in motion. The model is able to report an error message saying that the car being in motion is in the state of contradiction, and allows the speaker to initiate a repair of either altering the input ("the silver car passed the truck") or making changes to the domain rules associated with this input (e.g. sometimes stalled cars are towed and can thus be in motion).

### Task Recognition

When one agent's intentions must be understood and acted upon by another, addressees presume the speaker has a practical task outcome in mind that they can recognize and help achieve. For example, when the speaker says "Show me the black car passing the white car," the monitoring application is able to recognize that "show" is a directive or request with its agent being the implicit "you" or the application, its recipient being the speaker and its theme the scene described by "the black car passing the white car" represented on a card. The application in turn reasons that "show" indicates the task expected by the speaker to be a display event, namely displaying the scene described by its theme - "the black car passing the white car".

Another part of the common ground shared by the speaker and the application is the information regarding the ability of the application. For example, this application is capable of retrieving one of the shared scenes based on its description and displaying that scene to the speaker. In this

simple set up, it is, however, incapable of performing any other tasks. When the application is capable of performing the expected task (in this case the displayEvent), it leads to a successful interaction.

Task recognition may fail in one of two ways: 1) the intended task may not be correctly recognized — when the user says "Show me the next stop of the bus", the literal meaning of the bus at a signal light is not intended (converstional implicatures) or 2) the application may not be able to perform the identified task—for example, the application, currently set up to display only one scene, is incapable of responding to "Show me all the left turns of the red car." The model is able to identify when it is incapable of performing the task and allows the user to revise or repair the command.

## Conclusion

Modeling coordinated activity between agents has the potential to offer more flexibility to users in terms of interacting by means of more natural, human-like language. The models outlined here focus on an addressee's heuristic expectations of a speaker's use of common ground, salience and solvability in the coordination of meaning and understanding.

When modeling these aspects of common ground, an advantage of modeling these individual stages of an addressee's processing is the ability to identify the precise nature of the problem when joint coordination failures arise. The rudimentary set of models shown in this paper demonstrate the capabilities of our system to understand and interpret a given task given in natural language, taking into account world knowledge, historical context between the participants, and language specific knowledge. We also present several of the various stages in which a natural language interaction can fail and introduces the notion that cognitive models can be created to accommodate error recovery initiated by an agent participating in the conversation.

## References


Allen, J., and Perrault, R. 1980. Analyzing Intentions in Utterances. *Artificial Intelligence* 15(3), 143-178.

Bello, P., and Cassimatis, N. L. 2006. Understanding Other Minds: A Cognitive Modeling Approach. In *Proceedings of the 7th International Conference on Cognitive Modeling*. Trieste.

Callmeier, U. 2000. PET-A Platform for Experimentation with Efficient HPSG Processing. *Natural Language Engineering* 6(1):99-107.

Cassimatis, N. L. 2006. A Cognitive Substrate for Achieving Human-Level Intelligence. *AI Magazine* 27(2): 45-56.

Clark, H. H. 1996. *Using Language*. Cambridge University Press, Cambridge, UK.

Murugesan, A., and Cassimatis, N. L. 2006. A Model of Syntactic Parsing Based on Domain-General Cognitive Mechanisms. In *Proceedings of the 28th Annual Conference of Cognitive Science Society*, 1850-1855. Vancouver.

Pless, R., Jacobs, N., Dixon, M., Hartley, R., Baker, P., Brock, D., Cassimatis, N., and Perzanowski, D. 2009. Persistence and Tracking: Putting Vehicles and Trajectories in Context. In *Proceeding of the 38th IEEE Applied Imagery Pattern Recognition Workshop*. Washington, DC.

Sag, I. A., Wasow, T., and Bender, E. 2003. *Syntactic Theory: A Formal Introduction*, 2nd Ed. University of Chicago Press, Chicago.

Yusuke, M. and Jun'ichi, T. 2005. Probabilistic Disambiguation Models for Wide-Coverage HPSG Parsing. In *Proceedings of ACL-2005*, 83-90.