Grounding Language for Physical Systems
AAAI Technical Report WS-12-07

Situated Comprehension of Imperative Sentences in Embodied, Cognitive Agents

Shiwali Mohan and John E. Laird
Computer Science and Engineering
University of Michigan, Ann Arbor, MI 48105
{shiwali, laird}@umich.edu

Abstract

Linguistic communication relies on non-linguistic context to convey meaning. That context might include, for instance, recent or long-term experience, semantic knowledge of the world, or objects and events in the immediate environment. In this paper, we describe embodied agents instantiated in Soar cognitive architecture that use context derived from their linguistic, perceptual, procedural and semantic knowledge for comprehending imperative sentences.

Introduction

All linguistic communication occurs in a context. Contemporary theories of natural language comprehension in artificial systems have been limited to the examination of syntactic and semantic properties of words and sentences considered in isolation. Such systems range from naive systems that adopt the 'bag of words' approach and define the context of a word as its neighborhood, to formal systems that relate language to formal notations such as symbolic logic and the context is derived from logical reasoning and inference.

However, evolution of language did not occur in isolation from the environmental context. Although the theories of language evolution differ in important ways, they generally assume that sophisticated language developed to facilitate social coordination in situated tasks and collaborative learning. Research in human language processing has shown that people regularly leverage non-linguistic context to convey meaning. This non-linguistic context not only includes the objects and events in the immediate environment, but might also include a person's recent or long-term experiences and semantic facts about the world. Indeed, non-linguistic context crucially provides the means through which our communication is situated in both the world around us and our experiences. Use of language to refer to non-linguistic context allows humans to establish shared beliefs about the environment and to learn and generalize from others experiences.

An embodied, artificial agent that can effectively coordinate with humans and learn from such collaborative interactions should be able to comprehend language by connecting the linguistic symbols to its perceptions, actions, experiences and learning. We are interested in developing and studying embodied, cognitive agents that can associate linguistic symbols to aspects of cognition that originate outside of the linguistic system.

The main focus of this paper is to analyze the utility of a theory of situated language comprehension - the Indexical Hypothesis (Glenberg and Robertson 1999) in designing cognitive agents that can interpret natural language sentences using non-linguistic context derived from the environment and agent's experiences and demonstrate and discuss their linguistic capabilities.

The Indexical Hypothesis for Comprehension

(Glenberg and Robertson 1999) propose the Indexical Hypothesis that describes how sentences become meaningful through grounding their interpretation in situated action. The hypothesis asserts that comprehending a sentence requires three processes: first, *indexing* words and phrases to referents that establishes the contents of the linguistic input, second, deriving *affordances* from these referents, and third, *meshing* these affordances under the guidance of physical constraints along with constraints provided by the syntax of the sentence. The evidence from experimental studies (Kaschak and Glenberg 2000) supports the Indexical Hypothesis by suggesting a specific type of interaction between syntax and semantics that leads to understanding. The linguistic information specifies a general scene, and the affordances of objects are used to specify the scene in detail sufficient to take action.

Language can *index* a situation in various ways (Barsalou, 1999); in *immediate indexing*, the participants of the conversation are simultaneously embodied in a physical situation and use language to refer to objects and events in the current environment; in *displaced indexing*, the participants use language to refer to objects and events from prior experiences with the environment. These referents may not be present in the current perceptions.

The results of comprehension of a sentence are influenced by the intentions behind that utterance. For imperative sentences, such as "put that book in the shelf", the comprehension results in an action by the listener that leads to the intended goal. The speaker intends the listener to perform an action. However, comprehension of utterances such as assertions - "there is a blue couch in the living room" results in the establishment of shared belief. Interpretation of

an interrogation - "where is the blue couch?" results in a speech act that provides the requested information. Other utterances might result in perceptual simulation. Recognizing the intention behind an utterance is a significantly complex challenge and an open area of research. For the purposes of this paper, we will only study imperative sentences and assume that their comprehension results in an action in the environment.

Imperative sentences can rely on both immediate and displaced indexing. Consider the sentence - "put the red box on the table.". The speaker refers to objects in the current environment that fit the description of 'the red box' and 'the table' and assumes that such descriptions will allow the listener to resolve the intended objects. If this description is not sufficient to disambiguate the intended objects from other objects present or if the listener cannot find objects that fit these descriptions, the listener will resolve the disambiguity by further interaction. However, a sentence such as - "put the red box in the kitchen", refers to a location that fits the description 'kitchen' even though it might not be currently perceptible. The speaker assumes that the listener knows the referent from prior experience with the environment and can resolve the description correctly. If the listener fails in resolution, further interactions will occur.

In this paper, we propose a scheme for indexical comprehension in agents instantiated in Soar cognitive architecture, define indexing and meshing within the architectural constraints of Soar, demonstrate the utility of Soar cognitive mechanisms for immediate and displaced indexing, and analyze and discuss the linguistic capabilities of the agents. A novel contribution of this work is the use of various knowledge sources - perceptual, linguistic, procedural and semantic, for situated comprehension for cognitive agents.

Environment Overview

To study situated comprehension of imperative sentences, we chose a simple, simulated robotic domain shown in Figure 1. The domain simulates a toy kitchen; it includes three locations with simulated functions - a *stove*, a *dishwasher* and a *pantry*. It also consists of a variety of movable objects of different colors, sizes and shapes.

- Perceptions: The agent perceives its world as a set of objects and locations. Each object is associated with a set of perceptual symbols that describe its shape, size, color, and pose data. The locations are also augmented with symbols that represent their functional state, for example, the agent can perceive if the stove is on.
- Actions: The actions encoded in the domain are deterministic and include locomotion (`goto (<x>, <y>, <z>)`), object manipulation (`pick-up <id>`, `put-down <id>`) and functional (`set-value(stove,on)`).
- Interaction: A human can communicate with agent by typing natural language action commands through a chat interface. The human-agent communication is embedded in a mixed-initiative, interaction framework (Mohan and Laird 2012) that enables the agent to learn new, composite actions through interactions. Currently, the grammar is constrained to allow for parsing of simple imperative sentences (Table 1). Typically, an imperative sentence in English is composed of a verb followed by its argument structure that can refer to various physical entities (objects and locations in our environment).

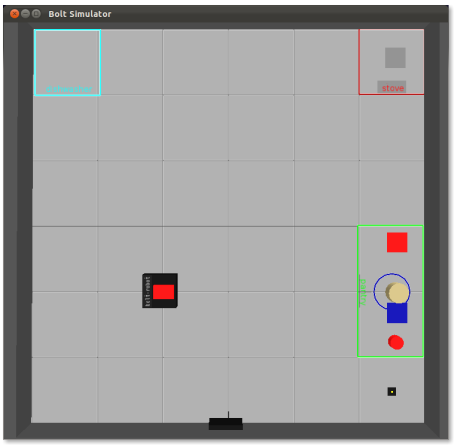

Figure 1: The Kitchen World

Table 1: Supported Grammar

$S \rightarrow VP$
$VP \rightarrow VB \mid VB\ NP \mid VB\ NP\ PP\ NP \mid VB\ PP\ NP$
$NP \rightarrow DET?\ ADJ^*\ Nominal$
$VB \rightarrow go, put, pick$
$DET \rightarrow a, the$
$PP \rightarrow on, to, in$
$ADJ \rightarrow red, blue, \ldots\ small, large, \ldots$
$Nominal \rightarrow cube, cylinder, pantry, dishwasher, stove$

Agent Design

We now describe the our agent design including a brief overview of the underlying cognitive architecture, the formal state representation convention, and the encoded action implementation knowledge.

Cognitive Architecture

The agents we describe have been instantiated in Soar (Laird 2012), a cognitive architecture based on the *problem space* hypothesis. It has been used extensively in designing AI systems and cognitive models. Through its various long term memories, a Soar agent can represent different forms of knowledge which can be used to inform agent's learning.

Perception Perceptual symbol grounding in high-dimensional data from sensors is an important research challenge. Prior work in grounded language acquisition (Roy 2002; Gupta and Davis 2008) has tackled this challenge and made some promising advances. In this paper, we assume that underlying sub-symbolic sensing mechanisms can reliably generate relevant perceptual symbols such

as color, size, shape and pose for various objects. Our agents are *grounded* in the simulated world (described in previous sections). This simplifying assumption allows us to investigate the association of language with other higher cognitive abilities of the agent (procedural and semantic knowledge). Future work will involve experimenting with robotic agents embedded in the real world.

Working Memory A Soar agent's beliefs about the current state are held in its working memory. The agent's beliefs are derived from its perceptions of the world (*immediate grounding*) and from its experiential knowledge of the world (*displaced grounding*). The data in working memory is represented as a symbolic, labeled graph of working memory elements (WME).

Semantic Memory Semantic memory (Derbinsky, Laird, and Smith 2010) provides the ability to store and retrieve declarative facts about the world and structural regularities of the environment. It is context independent; it contains knowledge that is not related to when and where it was acquired. The agent can *deliberately* store parts of its working memory into semantic memory as concepts. A concept can be retrieved from the semantic memory by placing a *cue* into a special buffer in working memory. The *cue* is then used to search semantic memory for a match biased by recency and frequency. The result is then retrieved into the working memory.

Episodic Memory Soar's episodic memory (Derbinsky and Laird 2009) is a context dependent memory that records the agent's experience during its lifetime. It effectively takes snapshots of working memory which are then stored in chronological fashion, providing the agent the ability to remember the context of past experiences as well as temporal relationships between experiences. A specific episode can be retrieved by deliberately creating a cue in an episodic memory buffer in the working memory. The episodic memory searches through past episodes for the best partial match biased by recency and retrieves the episode into the episodic memory buffer. Episodic memory also provides an ability to step through episodes once an episode is retrieved. The agents described in this work do not use episodic memory, however, in future we will extend the definition of *displaced indexing* to include agent's prior experiences.

Procedural Memory Behaviors and actions in Soar are represented as production rules in procedural memory. Whenever a conditions match the contents of working memory, the rule fires changing the state of the working memory. An operator is the unit of deliberation in Soar which when applied changes the internal state of the agent and may initiate changes in the environment through agent's actuators.

Decision-making in Soar is goal-directed. Deliberate goals in Soar take the form of operators in working memory, a distinction from other cognitive architectures where goals are often represented in declarative memory. The state of working memory causes rules to propose relevant operators. A selection mechanism makes a decision between proposed operators based on agent's selection knowledge. An operator is applied by rules that test for a set of WMEs and modify them.

If the operator selection or implementation knowledge is missing, an impasse results and a subgoal is created to correspondingly select an operator or implement the selected operator. In this goal, knowledge from other sources such as episodic memory, semantic memory, task-decomposition and/or look-ahead search can be used to inform the decision at the superstate. Through chunking, Soar compiles the reasoning performed in the substate into a new rule, effectively learning new rules that are applicable in future/analogous situations.

Environment State Representation

The agent maintains an *object-oriented* representation of the environment such that each object is associated with a set of perceptual attributes such as color, size, shape, pose etc.

Formally, let A be a set of perceptual attributes $\{a1, a2, ...\}$ like color, size, shape, pose etc. Every attribute has a domain $Dom(a_i)$ and for the sake of simplicity, we assume that these domains are non-overlapping sets. A set of classes, $C = \{C_1, ..., C_c\}$ is defined such that a class C_i is a set of attributes $\{C_i.a | C_i.a \in A\}$. Let O be a set of objects $\{o_1, ..., o_o\}$. Each object o_i is defined as an instance of one class C_j and is obtained by a complete value assignment to its attributes, $o_i = \{(C_j.a_k, val_{a_k}), ...(C_j.a_m, val_{a_m})\}$ where $val_{a_r} \in Dom(a_r)$. We define a function $Val(o_i)$ that returns the set of value assignment to the object's attributes.

A set of predicates, $P = \{P_1...P_p\}$ is defined over instantiated objects, $P_i(o_m, ..., o_n)$. These predicates include predicates that represent spatial relationships between objects and if an object is currently perceptible. The state of the environment is the set of true predicates, $S = \{P_k, ..., P_l\}$. The agent's belief about the environment state is represented by the presence/absence of WMEs that correspond to the predicates and objects in the current environment state.

The state of the environment is partially observable, i.e. there might be certain objects that are not perceptible to the agent. If an object is perceptible, the complete value assignment to its attributes is known. One can imagine a more complex scenario where the complete value assignment is not known for a perceptible object. However, in this work we concentrate on the simpler situation.

The kitchen domain has two classes; movable objects (cubes, cylinders etc) and locations (dishwasher, stove, pantry). Objects are perceptible only if they are within a certain distance from the robot.

Action Implementation Knowledge

The agent has procedural knowledge that allows it to perform certain *primitive actions* in the environment. The actions are implemented through operators in Soar. An action is defined by its *availability conditions* (a set of predicates that have to be true for the action to be applicable), *execution knowledge* - rules that execute locomotion and manipulation commands in the environment and *termination conditions*, a set predicates that signify that the goal of the action is achieved.

A primitive action, $pa_i(o_j...o_k)$ is instantiated for a set of objects $\{o_j...o_k\}$ when its availability conditions are met in

agent's working memory. The agent uses its domain knowledge to select between the available actions and applies one. The action is terminated when its goal is achieved.

The agent constantly maintains a list of available primitive actions PA. This list incorporates all the actions the agent can take given the current physical constraints, object affordances and its domain knowledge.

Linguistic Knowledge

For comprehension of imperative sentences by incorporating different cognitive knowledge sources, the agent should be able to associate linguistic symbols with perceptual symbols (*noun/adjective-perceptual symbol* mapping) and action control knowledge (*verb-operator* mapping). Additionally, the agent begins with some *semantic knowledge* about the domain.

Noun/Adjective-Perceptual Symbol Mapping

The agent can map its perceptual symbols to linguistic symbols such as *red*, *large*, *cube* etc. Conceptually, these can be understood as adjectives that qualify an object. The agent also can map a noun such as *dishwasher* or *cube* to an object. Acquisition of these mappings has been explored by prior work on concept labeling.

Verb-Operator Indexical Mapping

The verb-operator indexical mapping is encoded declaratively in the agent's semantic memory. Given the argument structure of the verb in the imperative sentence, this mapping allows the agent to access the related action operator and associtate the objects indicated by the noun phrases in the argument sturcture of the verb to the physical objects in the environment. Essentially, this mapping serves to associate the procedural knowledge of actions to lexical structure of imperative sentences.

Consider the example shown in Figure 2. It maps the verb 'put' with an argument structure consisting of a 'direct-object' and the object connected to the verb via the preposition 'in' with the operator 'op_put-down-object-location'. This allows the agent to associate the sentence - "put a red, large block in the dishwasher" with an appropriate operator which will achieve the intended goal.

The mapping (A5) contains two child nodes that represent the linguistic (A4) and the procedural (A6) structure. A4 points to an action operator that takes two physical objects as arguments. A6 connects these arguments (L1, O2) to the lexical structure of the corresponding imperative sentences.

Domain Semantic Knowledge

In Soar, semantic memory holds long-term, persistent declarative structures that generally correspond to facts about objects or concepts. These general facts about the environment can be used by the agent to *augment* its sensing of the environment state to reason about objects that are not currently perceptible, but are known to exist in the environment.

For this task, we encoded the objects belonging to the class *locations* in agent's semantic memory (Figure 3).

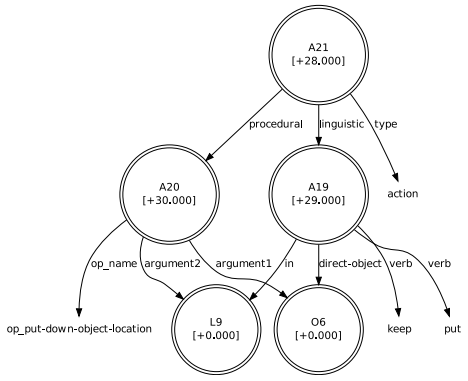

Figure 2: Verb-Action Indexical Mapping

These objects cannot be moved and their pose data does not change across situations. Encoding these objects in semantic memory allows to the agent to reason about action that can be instantiated for these object when the speaker talks about them even if they are not currently perceptible. It has been shown previously (Laird, Derbinsky, and Voigt 2011) that this knowledge can acquired autonomously from experiences in the environment.

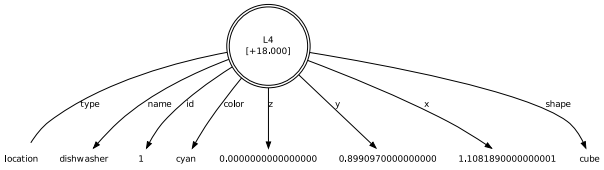

Figure 3: A Semantic Object

Indexing Imperative Sentences

To access and apply the corresponding situated action the agent has to (1) scan its perceptions and long-term memories for a physical referent corresponding to noun phrases in the sentence (*indexing arguments*), (2) associate the verb with the intended action (*indexing verb*) and (3) instantiate the action with the physical referents under the constraints action definition (*meshing*). The process is shown in Figure 4 and described in following sections. Consider the imperative sentence (and Figure 4) - "put a blue cube in the dishwasher", to comprehend this sentence the agent goes through the following processes.

Indexing Arguments

The goal of indexing arguments is to associate the linguistic description of an objects (the noun phrase) to a perceptible or a semantic object. The noun phrase in imperative sentences such as 'a blue cube', 'a large cylinder' or 'the dishwasher' is a linguistic description of an object present in the environment. Given a mapping between linguistic and perceptual symbols, the noun phrase can be mapped to a perceptual description of the object.

A noun phrase is translated to a set Q_o of perceptual symbols $\{p|p \in \{Dom(C_i.a_j)\}\}$. The agent tries to resolve this noun phrase to either perceptible objects (immediate indexing) or semantic objects (displaced indexing).

- Immediate Indexing: From the perceptible objects, the agent builds a candidate set of all objects such the value assignment to its attributes is a superset of the required perceptual description Q_o. Formally, a candidate set CO is a set of objects, $\{o_i|Q_o \in Val(o_i)\}$. If $CO = \{\phi\}$, the object being described by the noun phrase is not perceptible. The agent tries to resolve the phrase to the referent through displaced indexing.
- Displaced Indexing: The agent queries its semantic memory for an object such that the value assignment to its attributes is a superset of the required perceptual description ($\{o_i|Q_o \in Val(o_i)\}$). If the memory query returns in a failure, the *index arguments* phase fails and further interactions with the speaker are required to determine the correct physical referent. If the memory query is successful, the retrieved semantic object is added to the candidate set CO.

If the noun phrases are successfully indexed to perceptible objects or to objects from semantic memory (i.e. $CO \neq \{\phi\}$), the agent tries to index the verb in the imperative sentence. A distinct candidate set is created for every noun phrase in the sentence.

The example sentence "put a blue cube in the dishwasher", contains two noun phrases, - 'a blue cube' and 'the dishwasher'. 'A blue cube' can be indexed to a perceptible object. However, the object described by the phrase 'the dishwasher' is not perceptible, and therefore the agent tries displaced indexing. A candidate set will be created for each (CO_{do}, CO_{in}) of these noun phrases.

Indexing Verbs

The next phase is to map the verb in the imperative sentence to the correct operator (action). The argument structure of the verb in the imperative sentence is used to create a query, Q_v for semantic memory. For example, for the sentence "put up a blue cube in the dishwasher" that consists of the verb 'put' with one direct-object ('a blue cube') and a prepositional object ('dishwasher' connected to the verb by the preposition 'in'), the agent will query its semantic memory for a mapping that allows the agent to access the related action and associated arguments.

The retrieved operators are augmented with objects from the candidate set (CO_{do}, CO_{in}). Note that if the candidate set for an argument contains multiple elements, augmentation will result in multiple operators. These operators are added to a set of candidate operators CA.

If verb indexing results in a failure, either the agent does not know a related action or it knows a relevant action but lacks the mapping structure. Both of these situations are resolved by further interactions with the speaker.

Meshing

The candidate operator set CA can be understood as a set of different meanings of the imperative sentences intended by the speaker. CA can have multiple elements arising from ambiguous description of objects (CO_i has multiple elements). The set of available primitive actions PA consists of all the actions that are currently applicable given the current physical and spatial relationships between the objects. The intersection of both these sets, $CA \cap PA$ gives a set of intended actions that are applicable under current physical constraints and agent's domain knowledge. If this set contains a single element, that action operator is applied. If it contains multiple elements, further interaction or internal reasoning is necessary for resolution.

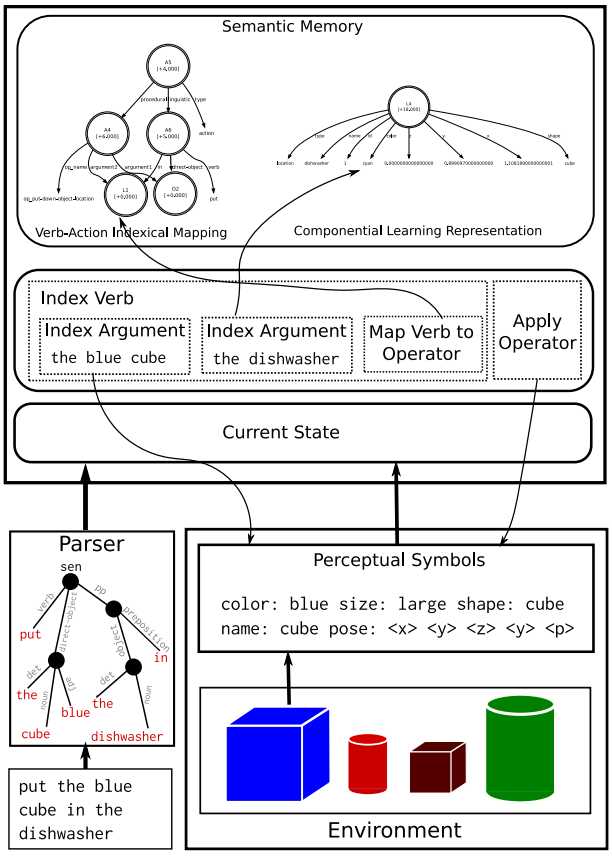

Figure 4: Indexing Imperative Sentences

Discussion

We have described the knowledge and processes encoded in the agent to allow for situated comprehension. We now discuss the linguistic and cognitive capabilities of the agents.

Linguistic Capabilities

Situated Referent Resolution The agent is able to integrate knowledge from different sources for resolving the physical referent indicated by the noun phrases.

- Perceptual Knowledge: *Immediate indexing* uses perceptual features of objects that are currently being sensed for resolution. Consider the perceptual situation shown in Figure 4 and assume that the speaker intends the agent

to pick up the blue cube. The object can be referred to in various ways, the simplest being 'the cube'. This information is not enough to completely resolve the referent since there are two objects that satisfy this description. If, instead, the speaker uses the noun phrase 'the large cube' or the 'the blue cube', the agent can successfully determine the correct object. However, if only one cube (blue) was perceptible, any noun phrase from 'the cube', 'the large cube', 'the blue cube' or 'the large blue cube' would have resulted in the agent picking up the blue cube.

- Semantic Knowledge: *Displaced indexing* uses the knowledge of the semantic facts about the environment for resolution. Consider again the situation shown in the Figure 4 in which the location dishwasher is not perceptible. Assume that the speaker intends the agent to go to the dishwasher. Since, the agent cannot perceive the dishwasher, it cannot reason about the actions associated with it. However, semantic knowledge about the dishwasher (which includes its position) allows the agent to instantiate the intended action and successfully apply it.
- Procedural Knowledge: *Meshing* allows for the use of procedural knowledge in resolving the intended referent. Consider a slight diversion from situation in Figure 4, in which the agent has picked up the blue cube. The speaker intends the agent to put down the blue cube and issues a "put down the cube" sentence. Although the noun phrase incompletely specifies the intended object given the agent's perceptions (the agent can also perceive the brown cube), the agent correctly resolves the referent since the 'put-down' action is only specified for objects the agent is holding.

Situated Action Resolution The meaning of verbs is greatly influenced by its argument structure which includes the direct objects and objects occuring in its prepositional phrases. In case of imperative sentences, the argument structure of a verb allows the speaker to specify the intended goal in terms of the relationships between objects. This suggests that a verb should index to different actions depending on its argument structure. The action corresponding to "put the small cylinder on the blue cube" is a sequence of object manipulations, which is different from "put the small cylinder in the dishwasher" which involves a sequence of locomotive and manipulative actions. Our indexical mapping scheme allows for mapping of verb argument structure to different operators which in turn allows for different interpretation of the same verb given its argument structure.

Verb Synonymy Different verbs can be used to refer to same or similar actions. Verb-operator indexical mapping (Figure 2) allows for verb synonymy. Using the map, the agent can resolve different verbs (*put* and *keep* in the example) to the same action of putting an object in a location.

Desiderata for Cognitive Language Comprehension

(Mayberry, Crocker, and Knoeferle 2009) characterize cognitive spoken language comprehension as - **incremental**, **anticipatory**, **integrative**, **adaptive** and **coordinated**. Although, our system does not include a speech understanding component, we find that it demonstrates the following desirable cognitive properties.

1. **Integrative**: *Multiple sources bear simultaneously on comprehension.* We have demonstrated an integration of knowledge from various cognitive sources (linguistic, procedural, semantic, perceptual) for comprehension.
2. **Adaptive**: *Comprehension robustly exploits relevant information whenever it is available.* Our system is adaptive and is able to perform correctly when desired object is not present in the current perceptions by relying on the semantic knowledge of the domain. We have also identified situations where further interaction might be necessary for correct resolution. Potentially, the agent can engage the speaker in a dialog for resolution.
3. **Coordinated**: *Sources of information may temporally depend on each other.* We have discussed ways in which ambiguities in sentence comprehension can lead to further interactions between the listener and speaker. Further research is required to understand how temporally distant information can be combined for effective comprehension. Soar's episodic memory can be useful source for this information.

Related Work

Researchers have studied the problem of grounding language from many different perspectives. There has been extensive research on grounded acquisition of nouns from labeled pictures (Barnard, Forsyth, and Jordan 2003; Gupta and Davis 2008) and computer-generated visual scenes (Roy 2002), associating linguistic descriptions with spatial relationships (Kollar et al. 2010) and ground verbs in visual perception (Siskind 2001). However, little work has been done in situated interpretation of sentences. Recent work by (Tellex et al. 2011) attempts to ground an action command using its compositional structure, to objects, events and locations.

Our work is best understood as a close kin to Winograd's SHRDLU, a well known system that could understand and generate natural language referring to objects and actions in a simple blocks world (Winograd 1972). Like our system it performs semantic interpretation during parsing by attaching short procedures to lexical units. Unlike our system, it does not combine constraints dervied from various cognitive components (linguistic, perceptual, procedural and semantic in our case) for effective linguistic comprehension. Our agents are designed within a well constrained cognitive architecture that has been shown to demonstrate learning and reasoning in a variety of domains. Our motivations are very closely aligned to Winograd's work, although, with the eventual goal of grounded in the real world via perceptual symbol grounding.

A recent work (Goertzel et al. 2010) describes software architecture which enables a virtual agent in an online virtual world to carry out simple English language interactions grounded in its perceptions and actions. This system uses knowledge from external sources such as FrameNet and other similar sources to associate semantic meaning to linguistic utterances. This work shares some conceptual ideas

with our research, however, we are interested in association of linguistic symbols to experiential knowledge acquired by the agent rather than an external knowledge base.

(Cantrell et al. 2010) demonstrate a natural language understanding architecture for human-robot interaction that integrates speech recognition, incremental parsing, incremental semantic analysis and situated reference resolution. The semantic interpretation of sentences is based on lambda representations and combinatorial categorial grammar. Our work is significantly different from this work in deriving semantic meaning from action representations.

Future Work

There several potential directions for further study. An immediate concern is devising formal evaluations for situated language processing in embodied agents. More specifically, we are interested in quantifying the information contained in an utterance and understanding how the information content in language affects disambiguation.

To be able to comprehend natural language sentences and associate them with perception and other cognitive abilities, the agent requires some background knowledge such as the indexical mapping structures shown in Figure 2. We are interested in investigating how this knowledge can be acquired by experience in the environment combined with interaction with a language expert (a human) allowing the agent to autonomously acquire language. A related research goal is to investigate if indexical mapping also aids in generation of natural language where the contents of agent's utterances are provided by its experience in the world.

Another direction we are interested in pursuing is integrating situated language comprehension and generation with a mixed-initiative interaction framework in which the agent can pursue collaborative tasks with a human. It has been demonstrated previously that an agent can acquire new procedural knowledge through human instruction (Huffamn and Laird 1995; Mohan and Laird 2011). A natural extension to this work is to ask if agent can acquire useful semantic, and procedural knowledge from collaborative action with humans. A situated language framework will facilitate such learning.

Acknowledgment

The work described here was supported in part by the Defense Advanced Research Projects Agency under contract HR0011-11-C-0142. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressly or implied, of the Defense Advanced Research Projects Agency or the U.S. Government.

References

Barnard, K.; Forsyth, D.; and Jordan, M. I. 2003. Matching Words and Pictures. *Journal of Machine Learning Research* 3:1107–1135.

Cantrell, R.; Scheutz, M.; Schermerhorn, P.; and Wu, X. 2010. Robust Spoken Instruction Understanding for HRI. In *Proceedings of the 5th ACM/IEEE International Conference on Human-Robot Interaction*, 275–282.

Derbinsky, N., and Laird, J. 2009. Efficiently Implementing Episodic Memory. In *Proceedings of the 8th International Conference on Case-Based Reasoning*.

Derbinsky, N.; Laird, J.; and Smith, B. 2010. Towards Efficiently Supporting Large Symbolic Declarative Memories. In *Proceedings of the 9th International Conference on Cognitive Modelling*.

Glenberg, A. M., and Robertson, D. A. 1999. Indexical Understanding of Instructions. *Discourse Processes* 28(1):1–26.

Goertzel, B.; Pennachin, C.; Araujo, S.; Silva, F.; Queiroz, M.; Lian, R.; Silva, W.; Ross, M.; Vepstas, L.; and Senna, A. 2010. A General Intelligence Oriented Architecture for Embodied Natural Language Processing. In *Proceedings of the 3d Conference on Artificial General Intelligence*.

Gupta, A., and Davis, L. S. 2008. Beyond Nouns: Exploiting Prepositions and Comparative Adjectives for Learning Visual Classifiers. In *Proceedings of the 10th European Conference on Computer Vision*.

Huffamn, S., and Laird, J. E. 1995. Flexibly Instructable Agents. *Journal of Artificial Intelligence Research* 3.

Kaschak, M. P., and Glenberg, A. M. 2000. Constructing Meaning: The Role of Affordances and Grammatical Constructions in Sentence Comprehension. *Journal of Memory and Language* 43(3):508–529.

Kollar, T.; Tellex, S.; Roy, D.; and Roy, N. 2010. Toward Understanding Natural Language Directions. In *Proceeding of the 5th ACM/IEEE international conference on Human-robot interaction - HRI '10*, 259. New York, New York, USA: ACM Press.

Laird, J. E.; Derbinsky, N.; and Voigt, J. 2011. Performance Evaluation of Declarative Memory Systems in Soar. In *Proceedings of the 20th Behavior Representation in Modeling Simulation Conference*, 33–40.

Laird, J. E. 2012. *The Soar Cognitive Architecture*. MIT Press.

Mayberry, M. R.; Crocker, M. W.; and Knoeferle, P. 2009. Learning to Attend: a Connectionist Model of Situated Language Comprehension. *Cognitive science* 33(3):449–96.

Mohan, S., and Laird, J. E. 2011. Towards Situated, Interactive , Instructable Agents in a Cognitive Architecture. In *Artificial Intelligence*.

Mohan, S., and Laird, J. E. 2012. Exploring Mixed-Initiative Interaction for Learning with Situated Instruction in Cognitive Agents. In *AAAI (Student Abstract)*.

Roy, D. 2002. Learning Visually Grounded Words and Syntax for a Scene Description Task. *Computer Speech & Language*.

Siskind, J. 2001. Grounding the Lexical Semantics of Verbs in Visual Perception using Force Dynamics and Event Logic. *Journal of Artificial Intelligence Research* 15:31–90.

Tellex, S.; Kollar, T.; Dickerson, S.; and Walter, M. 2011. Understanding Natural Language Commands for Robotic Navigation and Mobile Manipulation. In *Proceedings of the National Conference on Artificial Intelligence (AAAI 2011)*.

Winograd, T. 1972. Understanding Natural Language. *Cognitive Psychology* 3(1):1—-191.