

Title: Miniature Language Learning via Mapping of Grammatical Structure to Visual Scene  
Structure in English and Japanese

Authors : Peter Ford Dominey(1) and Toshio Inui (2)

1. Institut des Sciences Cognitives, CNRS UMR 5015,  
67 Blvd. Pinel, 69675 Bron Cedex,  
France  
phone: (33) 437911266 ; fax: (33) 437911210;  
e-mail: dominey@isc.cnrs.fr

2. Graduate School of Informatics,  
Kyoto University,  
Yoshida-honmachi, Sakyo-ku, 606-8501,  
Kyoto, Japan  
Phone : +81-75-753-3146  
Fax : +81-75-753-3292  
e-mail : inui@kyoto-u.ac.jp

*Abstract*— The current research demonstrates a system that employs relatively simple learning mechanisms to construct mappings between structure in visual scenes, and the grammatical structure of sentences that describe those scenes. These learned mappings allow the system to process natural language sentences in order to reconstruct complex internal representations of the visual scenes that those sentences describe. Initial representations are extracted using speech-to-text and color-based visual object recognition and tracking functions. Structure is then further extracted from these representations, and a novel associative learning technique is employed to establish the mappings between different grammatical structures and the corresponding event structure of the paired visual scene. During post-learning sentence interpretation, the appropriate mapping of grammatical structure to scene structure is retrieved based on grammatical markers inherent to the sentence. The system demonstrates error free performance and systematic generalization for a rich subset of English constructions that includes complex hierarchical grammatical structure. Further testing demonstrates (1) graceful degradation of performance in the presence of noise, (2) the capability to accommodate a significantly extended set of constructions, and (3) extension to Japanese, a free word order language that is structurally quite different from English, thus demonstrating the extensibility of the structure mapping model. The development of flexible natural language human-machine interfaces remains a highly complex and open area of investigation. The current research represents an initial investigation of how the construction grammar framework may provide an adaptable capability in this context.

*Index Terms*—perceptual scene analysis, language acquisition, model , neural network

## I. INTRODUCTION

Feldman et al. [1] posed the problem of "miniature" language acquisition based on <sentence, image> pairs as a "touchstone" for cognitive science. In this task, an artificial system is confronted with a reduced version of the problem of language acquisition faced by the child, that involves both the extraction of meaning from the image, and the mapping of the paired sentence onto this meaning. From a developmental perspective, Mandler [2] suggested that the infant begins to construct meaning from the scene based on the extraction of perceptual primitives. From simple representations such as contact, support, and attachment [3] the infant could construct progressively more elaborate representations of visuospatial meaning. In this context, the physical event "collision" is a form of the perceptual primitive "contact". Kotovsky & Baillargeon [4] observed that at 6 months, infants demonstrate sensitivity to the parameters of objects involved in a collision, and the resulting effect on the collision, suggesting indeed that infants can represent contact as an event predicate with agent and patient arguments. Siskind [5] has demonstrated that force dynamic primitives of contact, support, and attachment can be extracted from video event sequences and used to recognize events including pick-up, put-down, and stack based on their characterization in an event logic. The use of these intermediate representations renders the system robust to variability in motion and view parameters. Most importantly, Siskind demonstrated that the lexical semantics for a number of verbs could be established by automatic image processing. Likewise, similar results have been obtained for extracting simple events from video sequences by Steels and Baillie [6] in the context of robot communication. The common point is that extraction of events from video, invariant to parameters including orientation, motion profiles and viewpoint is advancing, but remains a wide open research domain.

Assuming that meaning can be extracted from the scene, the problem of language acquisition can be posed in the following manner: given a set of <sentence, meaning> pairs, the child should learn which

sentences are associated with which meanings, and should be able to generalize this knowledge to new sentences (e.g. [1], [7: 56]). The child comes to this task equipped with some innate learning capabilities that are often referred to as the "initial state" or the language acquisition device (LAD). The <sentence, meaning> pairs are referred to as the primary linguistic data (PLD), and the result of learning is the adult state. One school of thought, associated with Chomsky [8] holds that the PLD is highly indeterminate and underspecifies the mapping to be learned. Thus, they propose that the LAD embodies a genetically pre-specified syntactic system or universal grammar (UG), and that language acquisition consists of setting the UG parameters to correspond to those for the target language. This implies the "continuity hypothesis," which holds that via UG, children have an adult-like syntactic system that they bring to the problem of language acquisition ([9], and see discussion in [10]). This school thus argues for a UG in which the essential structure of the grammar is innate, and they propose that what is learned are the values of parameters that identify the target language grammar within the UG framework. Partially because of this endowment of UG, this school advocates the characterization of grammars in terms of formal syntactic regularities that can be characterized largely independently of semantics and pragmatics.

A separate "functionalist" school associated with authors including but not limited to Talmy [3], Feldman & Lackoff et al. [1], Langacker [11], Goldberg [12-14], Tomasello [10, 15-17] and others (see Newmeyer [18], and papers from Tomasello [15]) holds that the LAD does not contain a parameterized "universal grammar" but is rather a mechanism that learns the mapping between grammatical forms and meanings, (grammatical constructions) emphasizing the importance of communicative and social functions in language acquisition.

As form - meaning pairs, constructions include words, idioms and argument structure constructions such as active and passive which provide the basic means for clausal expression in a language [13]. In this context, Goldberg [12,13] proposes a tight correspondence between the structure of perceptual events that are basic to human experience, and the constructions for the corresponding basic sentence types. Children's acquisition of constructions would then require the matching of patterns extracted from the basic scenarios

of their human experience, and patterns extracted from sentential forms in their language. This type of pattern matching has been advocated by Fisher [19] who suggests that sentence structure can be mapped by analogy onto the child's conceptual representation of an observed event.

In contrast with the continuity hypothesis that suggests dormant adult-like syntactic capabilities in children, this framework is based in part on observations that the first grammatical constructions employed by infants appear more appropriately considered in terms of idiom-like linguistic gestalts that are initially fixed (e.g. "Gimme the ball"), and that through a progressive usage-based analysis become more open and productive (reviewed in [10,17]). In this context, the competence of the speaker is characterized as a structured inventory of grammatical constructions, rather than an abstract generative grammar [12-14, 20, 21]. This school diminishes the independent significance of abstract generative syntactic rules and places a much greater emphasis on the concrete relation between grammatical forms and meaning. This is consistent with the proposition that the formal systematic and productive properties of mental representations exist independent of these same properties in language [22, 23].

In this context of language acquisition, issues of learnability and innateness have provided a rich motivation for simulation studies that have taken a number of different forms. Elman [24] demonstrated that recurrent networks are sensitive to predictable structure in grammatical sequences. Subsequent studies of grammar induction demonstrate how syntactic structure can be recovered from sentences (e.g. [25]). From the "grounding of language in meaning" perspective (e.g. [1,11,12,26]), Chang & Maia [27] exploited the relations between action representation and simple verb frames in a construction grammar approach, and Cottrell et al. [28] associated sequences of words with simple image sequences. Steels and colleagues [6,29] have also extensively explored the interaction between meaning and language in the development of communication between agents. In a related effort to consider more complex grammatical forms, Miikkulainen [30] demonstrated a system that learned the mapping between relative phrase constructions and multiple event representations, based on the use of a stack for maintaining state information during the parsing of the next embedded clause in a recursive manner.

In the current approach, the visual scenes are made up of actions corresponding to *touch*, *push*, *take* and *give* that occur with colored toy blocks on an experimental workspace. The objects are manipulated by the experimenter who at the same time narrates the ongoing events. Image analysis yields a time-ordered list of the physical contacts between objects, and associated parameters including the relative velocities of the objects involved in a contact, and the duration of the contact. Based on this perceptual primitive information, a higher level representation is constructed in terms of specific events (touch, push, take, give) and the associated agent, object and recipient for each event, and the use of complex events corresponding to sentences with relative clauses. In parallel, the speech narrative of the ongoing events is processed to generate a text file of the narrative. Together, the set of perceptual events, and the corresponding set of event narratives are provided as input to an associative structure mapping algorithm that learns the mapping between words and their referents in the scene, and between grammatical structures and their event-level interpretations in the scene.

Within this context of learning grammatical constructions that correspond to distinct form-meaning pairs, there must be some reliable method for the child to identify and distinguish between different sentence forms or constructions, and their corresponding mappings to meaning. Bates et al. [31] identified four surface level cues that can serve in this role of mapping sentence to meaning including lexical items, word order, grammatical morphology (including free and bound morphemes) and prosody. Based on this proposition, Dominey [32-33] developed a construction-based model of lexical and phrasal semantics that demonstrated capabilities for argument structure satisfaction, or thematic role assignment, with a relatively restricted set of active and passive grammatical constructions. The effort in that study was to examine the importance of the interaction between meaning and grammatical structure. In the current study, the exploration of the construction model is extended and demonstrated to account for multiple aspects of phrasal semantics in English and Japanese.

The goals of the current study are fourfold: First to test the hypothesis that meaning can be extracted from visual scenes based on the detection of contact and its parameters in an approach similar to but

significantly simplified from Siskind [5]; Second to determine whether the model of Dominey [32,33] can be extended to handle embedded relative clauses and the corresponding hierarchical complexity of meaning representations; Third to demonstrate that these two systems can be combined to perform miniature language acquisition; and finally to demonstrate that the combined system can provide insight into the developmental progression in human language acquisition without the necessity of a pre-wired parameterized grammar system [8].

\*\*\*\* Insert Figure 1 About Here \*\*\*\*

## II. STRUCTURE MAPPING FOR LANGUAGE LEARNING

The model architecture is presented in Figure 1. From a behavioral perspective, during learning, the model is presented with <sentence, meaning> pairs, and it should learn the word meanings, and the set of grammatical constructions that define the sentence to meaning mappings in the input training set. During testing, the model should demonstrate that it can use this knowledge to understand new sentences that use the same lexicon, and the same set of grammatical constructions, but that were not presented in the training set. In particular the model should demonstrate systematicity, such that words that have only been experienced in particular syntactic roles (e.g. subject in an active transitive sentence) will be correctly processed when they appear in new legal syntactic positions (e.g. the same word now as an object in an active transitive sentence).

The functional organization of the model is based on the following principles: (1) Language acquisition can be characterized as learning the mappings from grammatical form to meaning (i.e. grammatical constructions) that should allow productive generalization with the learned constructions. That is, once a given grammatical construction has been learned, it can be instantiated with novel configurations of open class words in new sentences. (2) Within the sentence, the construction is encoded or identified by the relative configuration of open and closed class elements, that can thus be used as an index by which the corresponding construction for that sentence type can be learned and retrieved. This is

a specific re-statement of the slightly more general claim by Bates and MacWhinney [31] that includes prosody as an additional factor. These concepts are presented in an overview in Figure 1. The following sections then describe the model in detail.

### A. Word Meaning

In the initial learning phases, the association between a word and its corresponding scene item is learned by a simple associative memory, and is stored in the WordToReferent matrix (Eqn 1)<sup>1</sup>. In this initial configuration the term  $\alpha$  is 1, and this learning simply associates every word with every element in the current scene. This exploits a form of cross situational learning, in which the correct word-scene item associations will emerge as that which remains common across multiple sentence-scene situations [34]. In this manner the system can extract the cross-situational regularity that a given word will have a higher coincidence with the scene object to which it refers than with other objects. This allows initial word learning to occur, which contributes to learning the mapping between sentence and scene structure (Eqn. 4, 5 & 6 below). Once this learning has occurred, knowledge of the syntactic structure, encoded in SentenceToScene can be used to identify the appropriate referent (in the SEA) for a given word (in the OCA), corresponding to a zero value of  $\alpha$  in Eqn. 1. This corresponds to a form of “syntactic bootstrapping” in word learning. Thus, for the new word “gugle”, syntactic knowledge of the sentence “John pushed the gugle” can be used to assign “gugle” to the object of push.

$$\begin{aligned} \text{WordToReferent}(i,j) = & \text{WordToReferent}(i,j) + \\ & \text{OCA}(k,i) * \text{SEA}(m,j) * \text{LRSem} * \\ & (1-\alpha) * \text{SentenceToScene}(m,k) \end{aligned} \quad (1)$$

<sup>1</sup> In Eqn 1, the index  $k = 1$  to 6, corresponding to the maximum number of words in the open class array (OCA). Index  $m = 1$  to 6, corresponding to the maximum number of elements in the scene event array (SEA). Indices  $i$  and  $j = 1$  to 25, corresponding to the word and scene item vector sizes, respectively. LRSem is a learning rate parameter.

### B. Mapping Sentence to Scene

Grammatical construction learning consists in learning the mapping between open class element “slots” in a given sentence type, and the corresponding event/argument “slots” in the paired meaning representation. In terms of the architecture in Figure 1, this can be restated in the following successive steps. First, words in the OpenClassArray are decoded into their corresponding scene referents (via the WordToReferent mapping) to yield the PredictedReferentsArray (Eqn 2)<sup>2</sup> that contains the translated words while preserving their original order from the OCA.

$$PRA(k,j) = \sum_{i=1}^n OCA(k,i) * WordToReferent(i,j) \quad (2)$$

Next, each grammatical construction will correspond to a specific mapping between the PRA and the SEA. Distinct instances of these mappings are encoded in the SentenceToScene matrix for the different grammatical constructions. The problem will be to retrieve for each grammatical construction, the appropriate SentenceToScene mapping. To solve this problem, we rely on the Bates and MacWhinney coding principle such that each grammatical construction will have a unique ConstructionIndex in terms of the constellation of closed class elements with respect to open class element. Thus, the appropriate SentenceToScene mapping for each grammatical form can be indexed by its corresponding ConstructionIndex.

The ConstructionIndex (Eqn.3) encodes the closed class function words of a sentence, preserving their order of arrival and their relative position with respect to open class words in the sentence. Since each grammatical construction has a unique configuration of function words, the ConstructionIndex will thus uniquely identify each distinct grammatical construction. The ConstructionIndex is a 25 element vector.

Each function word is encoded as a single bit in a 25 element FunctionWord vector. When a function word is encountered during sentence processing, the current contents of ConstructionIndex are shifted (with wrap-around) by  $n + m$  bits where  $n$  corresponds to the bit that is on in the FunctionWord, and  $m$  corresponds to the number of open class words that have been encountered since the previous function word (or the beginning of the sentence). Finally, a vector addition is performed on this result and the FunctionWord vector.

$$\begin{aligned} \text{ConstructionIndex} &= \text{shift}(\text{ConstructionIndex}, (n + m)) \\ &+ \text{FunctionWord} \end{aligned} \quad (3)$$

ConstructionIndex thus encodes the function words, their relative order and their relative position with respect to the content words. The link between the ConstructionIndex and the corresponding SentenceToScene mapping is established as follows. As each new sentence is processed, we first reconstruct the specific SentenceToScene mapping for that sentence (Eqn 4)<sup>3</sup>. The resulting, SentenceToSceneCurrent encodes the correspondence between open class word order (that is preserved in the PRA) and thematic roles in the SEA for the given construction. Note that the quality of SentenceToSceneCurrent will depend on the quality of acquired word meanings as reflected in the PRA. Thus, syntactic learning requires a minimum baseline of semantic knowledge. Given the SentenceToSceneCurrent mapping for the current sentence, we can now associate it with the corresponding function word configuration for that sentence, expressed in the ConstructionIndex (Eqn 5)<sup>4</sup>.

---

<sup>2</sup> Index  $k = 1$  to 6, corresponding to the maximum number of scene items in the predicted references array (PRA). Indices  $i$  and  $j = 1$  to 25, corresponding to the word and scene item vector sizes, respectively.

<sup>3</sup> Index  $m = 1$  to 6, corresponding to the maximum number of elements in the scene event array (SEA). Index  $k = 1$  to 6, corresponding to the maximum number of words in the predicted references array (PRA). Index  $i = 1$  to 25, corresponding to the word and scene item vector sizes.

<sup>4</sup> Note that we have linearized SentenceToSceneCurrent from 2 to 1 dimensions to make the matrix multiplication more transparent. Thus index  $j$  varies from 1 to 36 corresponding to the 6x6 dimensions of SentenceToSceneCurrent.

$$\text{SentenceToSceneCurrent}(m,k) = \sum_{i=1}^n \text{PRA}(k,i) * \text{SEA}(m,i) \quad (4)$$

$$\begin{aligned} \text{ConstructionInventory}(i,j) &= (\text{ConstructionInventory}(i,j) \\ &+ \text{ConstructionIndex}(i) \\ &* \text{SentenceToSceneCurrent}(j) \\ &* \text{LRSyn} / \text{Sum}(\text{ConstructionInventory}) \end{aligned} \quad (5)$$

Finally, once this learning has occurred, for new sentences we can now extract the SentenceToScene mapping from the learned ConstructionInventory by using the ConstructionIndex as an index into this associative memory, illustrated in Eqn. 6<sup>5</sup>.

$$\text{SentenceToScene}(i) = \sum_{j=1}^n \text{ConstructionInventory}(i,j) * \text{ConstructionIndex}(j) \quad (6)$$

Note that the learning in (5) encodes the association between ConstructionIndex and the desired SentenceToScene mapping in ConstructionInventory. We can also assure that false associations are not encoded, by using the error between the “correct” SentenceToSceneCurrent and estimated SentenceToScene from (6). This error can then be used to weaken the association between ConstructionIndex and appropriate SentenceToScene elements.

An important issue concerns processing of sentences with embedded relative clauses such as “The cylinder that pushed the block touched the moon.” Such sentences encode two distinct events, and thus place additional requirements on the system. To accommodate the dual scenes for complex events, Eqns. 4-7 and the associated data structures are instantiated twice each, to represent the two components of the

dual scene. Thus, for these two-verb (e.g. relativised) sentences, the construction encodes the mapping from the sentence onto the meaning in the form of two distinct events. In the case of simple scenes, the second component of the dual scene representation is null. Miikkulainen [30] addressed this issue in a more general manner, by processing the clauses as they were completed, and storing ongoing clauses while awaiting their completion. While clearly more general, this approach is also significantly more expensive in terms of computational space and time, requiring the use of discrete parsing and stack management capabilities. The potential link between these approaches will be addressed in the discussion.

We evaluate performance by using the WordToReferent and SentenceToScene knowledge to construct for a given input sentence the “predicted scene”. That is, the model will construct an internal representation of the scene that corresponds to the input sentence. This is achieved by first converting the OpenClassArray into its corresponding scene items in the PredictedReferentsArray as specified in Eqn. 2. The referents are then re-ordered into the proper scene representation via application of the SentenceToScene transformation as described in Eqn. 7<sup>6</sup>.

$$\text{PSA}(m,i) = \text{PRA}(k,i) * \text{SentenceToScene}(m,k) \quad (7)$$

When learning has proceeded correctly, the predicted scene array (PSA) contents should match those of the scene event array (SEA) that is directly derived from input to the model. We then quantify performance error in terms of the number of mismatches between PSA and SEA.

### III. VISUAL SCENES AND ANALYSIS

---

<sup>5</sup> Again to simplify the matrix multiplication, Sentence-to-World has been linearized to one dimension, based on the original 6x6 matrix. Thus, index  $i = 1$  to 36, and index  $j = 1$  to 25 corresponding to the dimension of the ConstructionIndex.

<sup>6</sup> In Eqn 7, index  $i = 1$  to 25 corresponding to the size of the scene and word vectors. Indices  $m$  and  $k = 1$  to 6, corresponding to the dimension of the predicted scene array, and the predicted references array, respectively.

This section describes how “meaning” is extracted from visual scenes. From the perspective of human cognitive development, it is well known that among the most early acquired perceptual capacities is that of detecting physical contact between objects [4]. Based on this finding, contact will be treated as a perceptual primitive upon which more complex event representations will be built in visual scene analysis.

The visual environment consists of three objects on a black matte surface. The objects are a red cylinder, a green block and a blue semicircle or “moon.” Visual scenes are made up of events that occur between these objects, physically generated by the experimenter. The simplest events, *touch*, *push*, and *take*, involve the causal agent, and the object. The event *give* involves a third role corresponding to the recipient, and *take* can also involve a third argument in the case that the object is being taken from a source.

Within this context, the objective of the visual scene analysis is, for a given video sequence, to generate the corresponding event description in the format  $event_i(agent, object, recipient)$ , where event corresponds to touch, push, take or give, and  $i = 1$  for simple events (e.g. corresponding to sentence types 1-5 in Appendix 1), and  $i = (1, 2)$  for compound events (e.g. corresponding to sentence types 6-10).

#### A. Single Event Labeling

Scene events are defined in terms of contacts between elements. A contact between two physical elements is defined in terms of the time at which it occurred, the agent, object, and duration of the contact. The agent is determined as the element that had a larger relative velocity towards the other element involved in the contact. Interestingly, this parameter of movement is also one of the most perceptually salient visuo-spatial properties used by human infants in scene analysis [2-4]. All of this information can be directly extracted from the video images via on color-based object recognition and tracking. This allows for a system that can recognize a useful set of event categories, with computational complexity that is reduced with respect to related systems [5, 6]. Based on these parameters of contact, scene events are recognized using an event logic parser as follows:

1) *Touch(agent, object)*

This event corresponds to a single contact, in which (a) the duration of the contact is inferior to *touch\_duration* (1.5 seconds), and (b) the *object* is not displaced during the duration of the contact.

2) *Push(agent, object)*

This event corresponds to a single contact in which (a) the duration of the contact is superior or equal to *touch\_duration* and inferior to *take\_duration* (5 sec), (b) the object is displaced during the duration of the contact, and (c) the agent and object are not in contact at the end of the event.

3) *Take(agent, object)*

This event corresponds to a single contact in which (a) the duration of contact is superior or equal to *take\_duration*, (b) the object is displaced during the contact, and (c) the agent and object remain in contact.

4) *Take(agent, object, source)*

In this event, the agent takes the object from the source. This is a compound event in that it is made up of multiple contacts. For the first contact between the agent and the object (a) the duration of contact is superior or equal to *take\_duration*, (b) the object is displaced during the contact, and (c) the agent and object remain in contact. For the optional second contact between the agent and the source (a) the duration of the contact is inferior to *take\_duration*, and (b) the agent and source do not remain in contact. Finally, contact between the object and source is broken during the event.

5) *Give(agent, object, recipient)*

In this event, the agent first takes the object, and then gives the object to the recipient. This is a compound event, made up of multiple contacts. For the first contact between the agent and the object (a) the duration of contact is inferior to *take\_duration*, (b) the object is displaced during the contact, and (c) the agent and object do not remain in contact. For the second contact between the object and the recipient (a) the duration of the contact is superior to *take\_duration*, and (b) the object and recipient remain in contact. For the third (optional) contact between the agent and the recipient (a) the duration of the contact is inferior to *take\_duration* and thus the elements do not remain in contact.

While this decomposition of events may seem obvious or trivial, these event labeling templates form the basis for a effective event labeling algorithm based on contact template matching. Initial studies identified events based on the agency (determined as the element with the maximum relative velocity), and the contact durations. Displacement parameters can provide a more robust characterization for event discrimination.

### *B. Complex “Hierarchical” Events*

The events described above are simple in the sense that there have no recursive or hierarchical structure. This imposes serious limitations on the syntactic complexity of the corresponding sentences [26, 30]. Consider the sentence “The block that pushed the moon was touched by the triangle”. The hierarchical structure of this sentence corresponds to the hierarchical structure of its meaning that can be expressed by the event descriptions: *push(block, moon)*, and *touch(triangle, block)*. In this “dual event” representation of meaning, the “block” serves as the link that connects these two simple events in order to form a complex hierarchical event. In this manner, the main and embedded phrases of the sentence are mapped onto their respective event meaning representations. As mentioned above, the SceneEventArray is duplicated so that two events can be encoded, with corresponding duplication of the SentenceToScene, and ConstructionInventory mappings. A related method of using a multiple event representation for the meaning component of sentences with embedded clauses is developed in detail in [30].

## IV. IMPLEMENTATION ISSUES

In order to generate <sentence, meaning> pairs for training and testing the model, the human experimenter enacts and simultaneously narrates visual scenes made up of events that occur between a red cylinder, a green block and a blue semicircle or “moon” on a black matte table surface. A Sony CCD camera is located 85 cm above the surface with a field of view of 1 meter in diameter. The video image is processed by a color-based recognition and tracking system (Smart – Panlab, Barcelona Spain) that

generates a time ordered sequence of the contacts that occur between objects in the field of view of the camera. Based on this contact sequence, the higher level event description is extracted as described above. This description is encoded in the Scene Event Array (SEA) that consists of two sub-arrays so as to accommodate complex events (see IIIB). Each sub-array contains fields corresponding to *action*, *agent*, *object*, *recipient/source*. Each field is a 25-element vector with a single bit-on encoding. The SEA thus allows representation of the five simple event types, as well as their combinations in hierarchical events.

The simultaneous narration of the ongoing events is processed by a commercial speech-to-text system (IBM ViaVoice™). Content words are coded with single bit-on in the range 1-16, and function words in 17-25. The content (or open class) words will be encoded in the Open Class Array (OCA) that contains 6 fields, each a 25-element vector with single bin-on encoding. The function (or closed class) words are encoded in a linear array called the ConstructionIndex described above. Speech and vision data were acquired and then processed off-line yielding a data set of matched <sentence, scene> pairs that were provided as input to the structure mapping model. A total of ~300 <sentence, scene> pairs were generated in this manner.

For “dual-events” corresponding to embedded clauses, the events are encoded in the order in which they occur by the event recognition system, and for our current purposes, we specify sentence structure such that the relativised phrase refers to the event that occurs first in time so that the main and embedded phrases can be aligned with their meaning representations. Alternatively this can be left ambiguous such that the temporal ordering is unspecified. The main point is that in the event representation "push(block, triangle), touch(block, moon)" retains the linking element "block" in both events, independent of their temporal order.

For these <sentence, scene> pairs tested in Experiment A and B, the vocabulary was restricted to *cylinder*, *moon* and *block* for nouns, and *touch*, *push*, *give* and *take* for verbs. The closed class words were *to*, *by*, *from*, *that*, and *was*. In Experiment C the vocabulary was extended to include closed class words *and*, *it*, *itself*, and open class nouns *dog*, *cat* and *ball*; and the reflexive verb *said*. The set of constructions

learned by the model is presented in Appendix 1. In Experiment D, the model was tested with Japanese in order to determine if the approach would generalize to a free word-order language. These constructions are presented in Appendix 2.

## V. EXPERIMENTAL RESULTS

Four sets of results will be presented. First the demonstration of the model for scene analysis and sentence to meaning mapping under ideal, noise free conditions will be presented as a proof of concept. The effects of errors/noise in lexical categorization will then be examined, followed by a test of generalization to new grammatical constructions. Finally, in order to validate the cross-linguistic validity of the underlying principals, the model is tested with Japanese, a free word-order language that is qualitatively quite distinct from English.

### A. *Ideal conditions*

In the following experiments the human operator manipulated colored toy blocks in the CCD visual field, and simultaneously narrated his actions. Speech and vision data were acquired and then processed off-line. The two data streams were temporally aligned so that corresponding scene and narration were paired for each event. Each experiment thus yielded a data set of matched sentence – scene pairs that were provided as input to the structure mapping model.

#### 1) *Initial Learning of Active Forms for Simple Events*

The first experiment examined learning with the five event types, and corresponding narrations only in the active voice, corresponding to the grammatical forms 1 and 3 in Appendix 1.

For this experiment, 17 <sentence, scene> pairs were generated that employed the 5 different events. The model was trained for 32 passes through the 17 <sentence, scene> pairs for a total of 544 <sentence, scene> pairs (Fig 2, Exp 1). During the first 200 trials (scene/sentence pairs), value  $\alpha$  in Eqn. 1 was 1 and thereafter it was 0. This was necessary in order to avoid the effect of erroneous (random) syntactic knowledge on semantic learning in the initial learning stages. Evaluation of the performance of the model

after this training indicated that for all 17 sentences, there was error-free performance. That is, the PredictedScene generated from each sentence corresponded to the actual scene paired with that sentence. An important test of language learning is the ability to generalize to new sentences that have not previously been tested. Generalization in this form also yielded error free performance. In this experiment, only 2 grammatical constructions were learned, and the lexical mapping of words to their scene referents was learned. As stated in Section IIB, word meaning provides the basis for extracting more complex syntactic structure. Thus, these word meanings are fixed and used for the subsequent experiments.

\*\*\*\* Insert Figure 2 About Here \*\*\*\*

## 2) *Passive forms*

The second experiment examined learning with the five event types and the introduction of passive grammatical forms, thus employing grammatical forms 1-4 in Appendix 1. Seventeen new <sentence, scene> pairs were generated that employed the different event types, with two- and three- arguments, and active and passive grammatical forms for the narration. Word meanings learned in Experiment 1 were used, so only the structural mapping from grammatical to scene structure was learned. As indicated in Figure 2 (Exp 2), within 3 training passes through the 17 sentences, error free performance was achieved. Note that only the WordToReferent mappings were retained from Experiment 1. Thus, the 4 grammatical forms were learned from the initial naive state. In the generalization test, the learned values were fixed, and the model demonstrated error-free performance on new sentences for all four grammatical forms that had not been used during the training.

## 3) *Relative forms for Complex Events*

The complexity of the scenes and corresponding grammatical forms in the previous experiments were quite limited. Here we consider complex <sentence, scene> mappings that involve relativised sentences

and dual event scenes. Eleven complex scene/sentence pairs were generated with narration corresponding to the grammatical construction types 6-10 and 13 in Appendix 1.

After 8 presentations of the 11 scene/sentence training sentences, the model performed without error for these 6 grammatical forms. In the generalization test, the learned values were fixed, and the model demonstrated error-free performance on new sentences for all six grammatical forms that had not been used during the training.

#### *4) Combined Test*

The objective of the final experiment was to verify that the model was capable of learning the 10 grammatical forms together in a single learning session. A total of 27 scene/sentence pairs, used in Experiments B and C, were employed that exercised the ensemble of 10 grammatical forms. After exposure to 6 presentations of the 27 scene/sentence trials, the model performed without error. Likewise, in the generalization test the learned values were fixed, and the model demonstrated error-free performance on new sentences for all ten grammatical forms that had not been used during the training.

#### *5) Effects of WordToReferent Knowledge*

The rapid acquisition of the grammatical constructions in the presence of pre-learned WordToReferent knowledge is quite striking, and indicates the power of semantic bootstrapping that uses knowledge of word meaning to understand grammatical structure [35]. To further examine this effect, we re-ran these experiments A2-A4 without using the WordToReferent knowledge (i.e. word meanings) that had been acquired in Experiment A1. In this case the results were equally striking. The active and passive forms in A2 required more than 90 Training Passes to achieve error free performance, vs. 3 Training Passes when word meanings are provided, and 32 Training Passes when only the active forms were employed in A1. Training with the relativised constructions in A3 without pre-learned WordToReferent knowledge failed to converge, as did the combined test in A4. This indicates the importance of acquiring an initial

lexicon in the context of simple grammatical constructions, or even single word utterances [36] in order to provide the basis for acquisition of more complex grammatical constructions. This is consistent with the developmental observation that infants initially acquire a restricted set of concrete nouns from which they can bootstrap grammar, and further vocabulary [9, 35].

This set of experiments in ideal conditions demonstrates a proof of concept for the system, though several open questions can be posed based on these results. First, it appears that the ability to perform the lexical categorization of open and closed class elements is crucial to the functioning of the system, but it is likely that infants do not perform this categorization with perfect accuracy. Thus, it is important to determine how the model will perform in the face of partial errors in this categorization. Second, while the demonstration with 10 grammatical constructions is interesting, we can ask if the model will generalize to an extended set of constructions. Finally, we know that the English language is quite restricted with respect to its word order, and thus we can ask whether the theoretical framework of the model will generalize to free word order languages such as Japanese. Each of these three questions are addressed in the following three sections.

### *B. Learning with Lexical Categorization Errors*

The model relies on lexical categorization of open vs. closed class words both for learning lexical semantics, and for building the ConstructionIndex for phrasal semantics. While we can cite strong evidence that this capability is expressed early in development [37, 38] it is still likely that there will be errors in lexical categorization. The performance of the model for learning lexical and phrasal semantics for active transitive and ditransitive structures is thus examined under different conditions of lexical categorization errors. A lexical categorization error consists of a given word being assigned to the wrong category and processed as such (e.g. an open class word being processed as a closed class word, or vice-versa). Figure 3

illustrates the performance of the model with random errors of this type introduced at levels of 0 to 20 percent errors.

\*\*\*\* Insert Figure 3 About Here \*\*\*\*

We can observe that there is a graceful degradation, with interpretation errors progressively increasing as categorization errors rise to 20 percent. In order to further assess the learning that was able to occur in the presence of noise, after training with noise, we then tested performance on noise-free input. The interpretation error values in these conditions were 0.0, 0.4, 2.3, 20.7 and 33.6 out of a maximum of 44 for training with 0, 5, 10, 15 and 20 percent lexical categorization errors, respectively. This indicates that up to 10 percent input lexical categorization errors allows almost error free learning. At 15 percent input errors the model has still significantly improved with respect to the random behavior (~45 interpretation errors per epoch). Other than reducing the lexical and phrasal learning rates, no efforts were made to optimize the performance for these degraded conditions, thus there remains a certain degree of freedom for improvement. The main point is that the model does not demonstrate a catastrophic failure in the presence of lexical categorization errors.

### *C. Generalization to Extended Construction Set*

As illustrated above the model can accommodate 10 distinct form-meaning mappings or grammatical constructions, including constructions involving "dual" events in the meaning representation that correspond to relative clauses. Still, this is a relatively limited size for the construction inventory. The current experiment demonstrates how the model generalizes to a number of new and different relative phrases (the remaining 15 Types in 1-26 in Appendix 1), as well as additional sentence types including: conjoined (John took the key and opened the door), reflexive (The boy said that the dog was chased by the cat), and reflexive pronoun (The block said that it pushed the cylinder) sentence types (Types 27-38). The

consideration of these sentence types requires us to address how their meanings are represented. Conjoined sentences are represented by the two corresponding events, e.g. *took(John, key)*, *open(John, door)* for the conjoined example above. Reflexives are represented, for example, as *said(boy)*, *chased(cat, dog)*. This assumes indeed, for reflexive verbs (e.g. *said*, *saw*), that the meaning representation includes the second event as an argument to the first. Finally, for the reflexive pronoun types, in the meaning representation the pronoun's referent is explicit, as in *said(block)*, *push(block, cylinder)* for "The block said that it pushed the cylinder."

For this testing, the *ConstructionInventory* is implemented as a lookup table in which the *ConstructionIndex* is paired with the corresponding *SentenceToScene* mapping during a single learning trial. Based on the tenets of the construction grammar framework [12], if a sentence is encountered that has a form (i.e. *ConstructionIndex*) that does not have a corresponding entry in the *ConstructionInventory*, then a new construction is defined. Thus, one exposure to a sentence of a new construction type allows the model to generalize to any new sentence of that type. In this sense, developing the capacity to handle a simple initial set of constructions leads to a highly extensible system. Using the training procedures as described above, with a pre-learned lexicon (*WordToReferent*), the model successfully learned all of the constructions in Appendix 1, and demonstrated generalization to new sentences that it was not trained on.

That the model can accommodate these 38 different grammatical constructions with no modifications indicates its capability to generalize. This translates to a (partial) validation of the hypothesis that across languages, thematic role assignment is encoded by a limited set of parameters including word order and grammatical marking, and that distinct grammatical constructions will have distinct and identifying ensembles of these parameters. However, these results have been obtained with English that is a relatively fixed word-order language, and a more rigorous test of this hypothesis would involve testing with a free word-order language such as Japanese.

#### *D. Generalization to Japanese*

The current experiment will test the model with sentences in Japanese. Unlike English, Japanese allows extensive liberty in the ordering of words, with grammatical roles explicitly marked by postpositional function words *-ga*, *-ni*, *-wo*, *-yotte*. This word-order flexibility of Japanese with respect to English is illustrated here with the English active and passive di-transitive forms that each can be expressed in 4 different common manners in Japanese:

1. The block gave the circle to the triangle.
  - 1.1 Block-ga triangle-ni circle-wo watashita .
  - 1.2 Block-ga circle-wo triangle-ni watashita .
  - 1.3 Triangle-ni block-ga circle-wo watashita .
  - 1.4 Circle-wo block-ga triangle-ni watashita .
2. The circle was given to the triangle by the block.
  - 2.1 Circle-ga block-ni-yotte triangle-ni watasareta .
  - 2.2 Block-ni-yotte circle-ga triangle-ni watasareta .
  - 2.3 Block-ni-yotte triangle-ni circle-ga watasareta .
  - 2.4 Triangle-ni circle-ga block-ni-yotte watasareta .

In the “active” Japanese sentences, the postpositional function words *-ga*, *-ni* and *-wo* explicitly mark agent, recipient and, object whereas in the passive, these are marked respectively by *-ni-yotte*, *-ga*, and *-ni*. For both the active and passive forms, there are four different legal word-order permutations that preserve and rely on this marking. Japanese thus provides an interesting test of the model’s ability to accommodate such freedom in word order.

Employing the same method as described in the previous experiment, we thus expose the model to <sentence, meaning> pairs generated from the 26 Japanese constructions described in Appendix 2. We

predicted that by processing the -ga, -ni, -yotte and -wo markers as closed class elements, the model would be able to discriminate and identify the distinct grammatical constructions and learn the corresponding mappings. Indeed, the model successfully discriminates between all of the construction types in Appendix 2 based on the ConstructionIndex unique to each construction type, and associates the correct SentenceToScene mapping with each of them. As for the English constructions, once learned, a given construction could generalize to new untrained sentences.

This demonstration with Japanese is an important validation that at least for this subset of constructions, the construction-based model is applicable both to fixed word order languages such as English, as well as free word order languages such as Japanese. This also provides further validation for the proposal of Bates and MacWhinney [31] that thematic roles are indicated by a constellation of cues including grammatical markers and word order.

## VI. DISCUSSION

These results demonstrate that meaning can be extracted from visual scenes based on recognition of the perceptually salient primitive “contact”, and that grammatical sentence structure can be mapped onto these meanings in a generalized and productive manner. Part of the argument that is proposed is that by choosing a particular manner to address these problems, we solve them not with arbitrary brute force solutions that do not generalize, but rather with developmentally inspired solutions that are immediately generalizable, and that are suited for providing the basis for more productive extensions.

### A. *Extracting Meaning*

The problem of interpretation of visual scenes remains a major focus of computer vision research. In a novel approach to this problem, Siskind identified a small set of perceptual primitives including support, contact that could be used to provide abstract descriptions of physical events that are largely invariant to specific properties of view angle, velocity etc. [5]. In this same spirit, we looked to the developmental literature for perceptual primitives that could provide the basis for an invariant event recognition capability,

but with a significantly more simple computational approach. Immediately, the primitive “contact” became an obvious choice as an atom from which higher level events can be constructed, and indeed the developmental literature revealed that infants are sensitive to the parameters of physical contacts very early in development. Whereas Siskind employed a rather complex architecture based on an extended set of force dynamic primitives, our challenge was to generate an useful set of event categories using a single primitive and a very simple pattern matching method for recognizing them. Thus, we currently recognize 5 distinct categories of physical events, that is comparable with the performance of related systems (e.g. pickup, putdown, stack, unstack, move, assemble and disassemble [5]). While clearly this does not represent a final or “adult” state, it is of sufficient capacity to generate meanings that can then be integrated into grammatical constructions via the structure mapping model.

### *B. Structure Mapping*

The structure mapping approach is based on two principles from cognitive development theories that provide a significant generalization capability. First, the principle that language acquisition involves learning the mapping from sentences to meaning is a central principal of the functionalist construction grammar approach [12-17, 20, 21]. Second, the principle that the construction type for a given input sentence can be identified by the constellation of closed class elements in that sentence is a functional reformulation of the cue competition hypothesis of Bates and MacWhinney [31]. Implemented together in the current model, these principles provide the basis for a powerful generalization capability at three levels.

The first level of generalization is within constructions. Once a given construction has been learned, it can then be employed with an open set of new sentences. This provides for generalization that satisfies the systematicity requirement of Fodor and Pyshlyyn [39] such that a system that can represent “John loves Mary” can also represent “Mary loves John” without additional training. The second level of generalization concerns the learning of new constructions. As illustrated in Results Section C, the model is capable of acquiring new constructions with no additional modifications – simply through exposure to

<sentence, meaning> pairs that define the new construction. This indicates that the structure mapping model is not an ad hoc solution that works only for a fixed set of sentences and construction types, but rather that it is a more robust solution that generalizes. This form of generalization remains within the domain of the English language, thus introducing the third dimension of generalization. The third level of generalization is revealed by the demonstration that the model can also accommodate the free word-order language Japanese. This provides evidence that the principles underlying the model are applicable across different language types, and provide a potentially universal language acquisition capability. The verity of this claim remains to be tested with further languages, but we can at least conclude that the model is applicable to the miniature English and Japanese language domains explored in the current study. Part of the important message is that through the judicious use of a relatively simple construction mapping system, a great deal of communicative mileage can be gained. Concretely this corresponds to the ability to learn and use an inventory of grammatical constructions as sentence to meaning mappings. Interestingly, though we did not address this issue here, the system should also generalize to imperative (command), and interrogative (question) constructions, as long as the “meaning” component can be represented in a predicate-argument format [19]. These issues remain to be addressed in further research.

### *C. Limitations and Extensions*

In its current form, the model does fail to generalize in at least one important manner. That is, in order to perform the sentence to meaning mapping for sentences based on a new grammatical construction type, model requires a training exposure to a <sentence, meaning> pair in order to learn the corresponding mapping. Human language processing allows us to accommodate new construction types “on the fly,” and a model of language processing should be capable of addressing this issue. To consider a concrete example, a system should be able to accommodate relative phrases in novel sentence positions. Looking at construction types 6 and 9 in Appendix 1, we see in the Construction 6 that the agent of the sentence is a relativised noun phrase (The block that pushed the cylinder...), whereas in Construction 9 it is the object

that is a relativised noun phrase (... the moon that pushed the block). Ideally the system should accommodate these relativised noun phrases even when they appear in novel locations. Miikkulainen [30] has addressed this problem, by employing a pre-specified parser to identify relative phrase structure. An interesting extension of our current system would be to employ a pattern finding capability that could recognize that these relativised phrases are often encountered following the word “that” and thus to provide such a parsing capability via learning.

In this context, it is worth mentioning that in its current form, the model does not construct an explicit representation of syntactic structure, but rather performs a direct mapping from sentence to meaning. Interestingly, it appears that humans may operate in the same manner. Ferreira [40] has demonstrated that when exposed to sentences with complex or non-canonical syntax, subjects often make interpretation errors indicating they are performing heuristic “shallow parsing” rather than algorithmic construction of the complete syntactic representation. Related results providing evidence that such “underspecified” representations are used by humans during comprehension are reviewed in [41], suggesting indeed that human sentence processing often relies on pragmatic aspects of meaning rather than on the construction of deep representations of syntactic structure, and the interaction between these mechanisms is an important topic of future research. A model of language processing should thus take into account the possibility of both types of analysis. In conclusion, natural language interfaces between humans and machines will become increasingly present. The current research represents an initial investigation of how the construction grammar framework may provide an adaptable capability in this context, though much work remains to be done.

## VII. APPENDIX 1: ENGLISH GRAMMATICAL CONSTRUCTIONS

Each numbered sentence is an example of a specific abstract grammatical construction type whose meaning is provided in an event(argument) format following the sentence(s) corresponding to that meaning. Each construction can generalize to new sentences in which the open class elements are replaced.

**Single-Event Constructions**

1. The block pushed the cylinder.

Push(block, cylinder)

2. The cylinder was pushed by the block.

Push(block, cylinder)

3. The block gave the cylinder to the moon.

Give(block, cylinder, moon)

4. The cylinder was given to the moon by the block.

Give(block, cylinder, moon)

5. The block gave the moon the cylinder.

Give(block, cylinder, moon)

**Dual-Event Relative Constructions**

6. The block that pushed the cylinder touched the moon.

push(block, cylinder), Touch(block, moon)

7. The block was pushed by the moon that touched the cylinder.

Touch(moon, cylinder), Push(moon, block)

8. The moon that pushed the cylinder was touched by the block.

Push(moon, cylinder), Touch(block, cylinder)

9. The cylinder touched the moon that pushed the block.

Push(moon, block), Touch(cylinder, moon).

10. The block that was pushed by the moon touched the cylinder.

Push(moon, block), Touch(block, cylinder)

11. The cylinder was touched by the block that was pushed by the moon.

Push(moon, block), Touch(block, cylinder)

12. The moon that was touched by the cylinder was pushed by the block.

Touch(cylinder, moon), Push(block, moon)

13. The moon took the cylinder that was touched by the block.

Touch(block, cylinder), Take(moon, cylinder).

14. The cylinder was given to the moon by the block that touched the moon.

Touch(cylinder, moon), Give(block, cylinder, moon).

15. The block that touched the moon was given to the cylinder by the dog.

Touch(block, moon), Give(dog, block, cylinder).

16. The cat gave the dog to the cylinder that pushed the block.

Push(cylinder, block), Give(cat, dog, cylinder)

17. The cat was given from the dog to the block that pushed the cylinder.

Push(block, cylinder), Give(dog, cat, block)

18. The cylinder that was pushed by the block gave the cat to the dog.

Push(block, cylinder), give(cylinder, cat, dog).

19. The block gave the moon to the cylinder that was touched by he cat.

Touch(cat, cylinder), Gave(block, moon, cylinder)

20. The cat that gave the block to the moon pushed the cylinder.

Gave(cat, block, moon), Push(cat, cylinder)

21. The block was pushed by the moon that gave the cat to the cylinder.

Gave(moon, cat, cylinder), Push(moon, block)

22. The block pushed the moon that gave the cat to the cylinder.

Gave(moon, cat, cylinder), Push(block, moon)

23. The block that gave the moon to the cat was pushed by the cylinder.

Gave(block, moon, cat), Push(cylinder, block)

24. The block that was given to the moon by the cat pushed the cylinder.

Give(cat, block, moon), Push(block, cylinder)

25. The block pushed the moon that was given by the cat to the cylinder.

Give(cat, moon, cylinder), Push(block, moon)

26. The block that pushed the moon gave the cat to the cylinder.

Push(block, moon), Give(block, cat, cylinder)

### **Dual-Event Conjoined Constructions**

27. The block pushed the cylinder and the moon.

Push(block, cylinder), Push(block, moon)

28. The block and the cylinder pushed the moon.

Push(block, moon), Push(cylinder, moon)

29. The block pushed the cylinder and touched the moon.

Push(block, cylinder), Touch(block, moon).

30. The moon and the block were given to the cylinder by the cat.

Give(cat, moon, cylinder), Give(cat, block, cylinder).

### **Reflexive Constructions (Single- and Dual-Event)**

31. The block pushed itself.

Push(block, block)

32. The block said that the moon pushed itself.

Said(block), Push(moon, moon)

33. The block said that the moon pushed it.

Said(block), Push(moon, block).

34. The block said that it touched the cylinder.

Said(block), Touch(block, cylinder)

35. The block said that the cylinder was touched by the moon.

Said(block), Touch(moon, cylinder)

36. The block said that the cylinder gave the moon to the cat.

Said(block), Gave(cylinder, moon, cat)

37. The block said the cylinder gave the moon to the cat.

Said(block), Gave(cylinder, moon, cat)

38. The moon said that the block touched the cylinder.

Said(moon), Touch(block, cylinder).

#### VIII. APPENDIX 2: JAPANESE CONSTRUCTIONS<sup>7</sup>

Each numbered sentence is an example of a specific abstract grammatical construction type in Japanese whose meaning is provided in an event(argument) format *following* the sentence(s) corresponding to that meaning. The corresponding English construction from Appendix 1 is indicated in ()'s. Each construction can generalize to new sentences in which the open class elements are replaced.

1(1). block-ga circle-wo tataita.

2(1). circle-wo block-ga tataita.

*The block hit the circle.*

<sup>7</sup> hit = tatakau, hit = tataita, be hit = tatarareru, was hit = tatakareta; give = ataeru, gave = watahita, be given = ataerareru, was given = watasareta; push = osu, pushed = tataita, be pushed = osareru, was pushed = osareta, believe = shinjiru, believed = shinjita; itself = jibun or jishin, it = sore.

*Hit(block, circle) active*

3(2). Circle-ga block-ni tatakareta.

4(2). Block-ni circle-ga tatakareta.

*The circle was hit by the block.*

*Hit(block, circle) passive*

5(3). Block-ga triangle-ni circle-wo watashita .

6(3). Block-ga circle-wo triangle-ni watashita .

7(3). Triangle-ni block-ga circle-wo watashita .

8(3). Circle-wo block-ga triangle-ni watashita .

*The block gave the circle to the triangle.*

*Gave(block, circle, triangle) active*

9(4). Circle-ga block-ni-yotte triangle-ni watasareta .

10(4). Block-ni-yotte circle-ga triangle-ni watasareta .

11(4). Block-ni-yotte triangle-ni circle-ga watasareta .

12(4). Triangle-ni circle-ga block-ni-yotte watasareta .

*The circle was given to the triangle by the block.*

*Gave(block, circle, triangle) passive*

13(6). Circle-wo tataita block-ga triangle-wo oshita.

*The block that hit the circle pushed the triangle.*

*Hit(block, circle), Pushed(block, triangle)*

14(7). Block-ga circle-wo oshita triangle-ni-yotte tatakareta.

15(7). Circle-wo oshita triangle-ni-yotte block-ga tatakareta.

*The block was hit by the triangle that pushed the circle.*

*Pushed(triangle, circle), Hit(triangle, block)*

16(8). Circle-wo tataita block-ga triangle-ni-yotte osareta.

17(8). Triangle-ni-yotte circle-wo tataita block-ga osareta.

*The block that hit the circle was pushed by the triangle.*

Hit(block, circle), Pushed(triangle, block)

18(9). Block-ga circle-wo oshita triangle-wo tataita.

19(9). Circle-wo oshita triangle-wo block-ga tataita.

*The block hit the triangle that pushed the circle.*

Pushed(triangle, circle), Hit(block, triangle)

20(10). Circle-ni-yotte tatakareta block-ga triangle-wo oshita.

*The block that was hit by the circle pushed the triangle.*

21(27). Block-ga circle-to triangle-wo tataita.

22(27). Circle-to triangle-wo block-ga tataita.

*The block hit the circle and the triangle.*

23(28). Block-to triangle-ga circle-wo tataita.

24(28). Circle-wo block-to triangle-ga tataita.

*The block and the triangle hit the circle.*

25(33). Block-ga sore-wo tataita triangle-wo oshita.

*The block pushed the triangle that hit it.*

26(34). Block-ga sore-ga tataita triangle-wo oshita.

*The block pushed the triangle that it hit.*

#### REFERENCES

- [1] Feldman JA, Lakoff G, Stolcke A, Weber SH (1990) Miniature language acquisition : A touchstone for cognitive science. Proc. 12 Ann. Conf. Cognitive Science Society, 686-693

- [2] Mandler J (1996) Preverbal representation and language. In Bloom et al. (Eds) *Language and Space*, 365-384.
- [3] Talmy L (1988) Force dynamics in language and cognition. *Cognitive Science*, 10(2) 117-149.
- [4] Kotovsky L, Baillargeon R (1998) The development of calibration-based reasoning about collision events in young infants., *Cognition*, 67, 311-351
- [5] Siskind JM (2001) Grounding the Lexical Semantics of Verbs in Visual Perception Using Force Dynamics and Event Logic, *Journal of Artificial Intelligence Research*, volume 15, pp. 31-90
- [6] Steels, L and Baillie J-C (2003). Shared Grounding of Event Descriptions by Autonomous Robots. *Robotics and Autonomous Systems*, 43(2-3):163-173
- [7] Crain S, Lillo-Martin D (1999) *An introduction to linguistic theory and language acquisition*. Blackwell, Malden MA, USA.
- [8] Chomsky N. (1995) *The Minimalist Program*. MIT
- [9] Pinker S (1984) *Language learnability and language development*, Cambridge, MA, Harvard University Press
- [10] Tomasello M (2000) Do young children have adult syntactic competence? *Cognition* 74 (2000) 209-253
- [11] Langacker, R. (1991). *Foundations of Cognitive Grammar. Practical Applications, Volume 2*. Stanford University Press, Stanford.
- [12] Goldberg, A. (1995) *Constructions*. University Chicago Press, Chicago and London.
- [13] Goldberg, A. (1998) Patterns of experience in patterns of language, In (Michael Tomasello Ed) *The new psychology of language*, Vol 1, 203-19
- [14] Goldberg, A. (2003) *Constructions: a new theoretical approach to language*, *Trends in Cognitive Science*, Volume 7, Issue 5 , May 2003, Pages 219-224
- [15] Tomasello, M. (1998). *The new psychology of language: Cognitive and functional approaches*, Mahwah, NJ: Erlbaum.

- [16] Tomasello M (1999) The item-based nature of children's early syntactic development, *Trends in Cognitive Science*, 4(4) :156-163
- [17] Tomasello, M. (2003) *Constructing a language: A usage-based theory of language acquisition*. Harvard University Press, Cambridge.
- [18] Newmeyer F (1998) *Language Form and Language Function*. MIT
- [19] Fisher C (1996) Structural limits on verb mapping: The role of analogy in children's interpretation of sentences. *Cognitive Psychology*, 31, 41-81
- [20] Croft W (2001) *Radical construction grammar: syntactic theory in typological perspective*. Oxford: Oxford University Press.
- [21] Croft (2003) Logical and typological arguments for Radical Construction Grammar, in *Construction Grammar(s) Cognitive and cross-language dimensions*, ed. Mirjam Fried and Jan-Ola Östman, Amsterdam, John Benjamins, In press.
- [22] Fodor JA (1975) *The Language of Thought*, Cambridge MA, Harvard U Press,
- [23] Jackendoff R (1999) Parallel constraint-based generative theories of language. *Trends Cogn Sci*. 1999 Oct;3(10):393-400.
- [24] Elman JL (2001) Connectionism and Language Acquisition. In M. Tomasello & E. Bates (Eds.), *Language Development*, Oxford, Blackwell
- [25] Stolcke A, Omohundro SM (1994) Inducing probabilistic grammars by Bayesian model merging, In *Grammatical Inference and Applications: Proc. 2<sup>nd</sup> Intl. Colloq. On Grammatical Inference*, Springer Verlag.
- [26] Feldman J., G. Lakoff, D. Bailey, S. Narayanan, T. Regier, A. Stolcke (1996). L0: The First Five Years. *Artificial Intelligence Review*, v10 103-129.
- [27] Chang NC, Maia TV (2001) Grounded learning of grammatical constructions, *AAAI Spring Symp. On Learning Grounded Representations*, Stanford CA.

- [28]Cottrel GW, Bartell B, Haupt C. (1990) Grounding Meaning in Perception. In Proc. GWAI90, 14<sup>th</sup> German Workshop on Artificial Intelligence, pages 307--321, Berlin, New York,. Springer Verlag
- [29]Steels, L. and Kaplan, F. AIBO's first words : The social learning of language and meaning. *Evolution of Communication*, 4(1), 2001
- [30]Miikkulainen R (1996) Subsymbolic case-role analysis of sentences with embedded clauses. *Cognitive Science*, 20: 47-73.
- [31]Bates E, McNew S, MacWhinney B, Devescovi A, Smith S (1982) Functional constraints on sentence processing: A cross-linguistic study. *Cognition*, 11, 245-299.
- [32]Dominey PF (2000) Conceptual Grounding in Simulation Studies of Language Acquisition, *Evolution of Communication*, 4(1), 57-85.
- [33]Dominey, P.F. (2003) Learning Grammatical Constructions in a Miniature Language from Narrated Video Events, Proceedings of the 25<sup>th</sup> Annual Meeting of the Cognitive Science Society, Boston
- [34]Siskind JM (1996) A computational study of cross-situational techniques for learning word-to-meaning mappings, *Cognition* (61) 39-91.
- [35]Pinker S (1987) The bootstrapping problem in language acquisition. In B. MacWhinney, ed., *Mechanisms of language acquisition*. Hillsdale, NJ: Lawrence Erlbaum Associates.
- [36]Brent MR, Siskind JM (2001) The role of exposure to isolated words in early vocabulary development. *Cognition*, 81, B33-44.
- [37]Shi R., Werker J.F., Morgan J.L. (1999) Newborn infants' sensitivity to perceptual cues to lexical and grammatical words, *Cognition*, Volume 72, Issue 2, B11-B21.
- [38]Höhle B, Weissenborn J (2003) German-learning infants' ability to detect unstressed closed-class elements in continuous speech, *Developmental Science*, 6 (2) 122-127.
- [39]Fodor JA, Pylyshyn Z (1988) "Connectionism and Cognitive Architecture: A Critical Analysis" in S Pinker and J. Mehler, eds. *Connections and Symbols*, Cambridge MA, MIT Press (A Cognition Special Issue).

[40]Ferreira F (2003) The misinterpretation of noncanonical sentences. *Cognitive Psychology*: 47, 164-203

[41]Sanford AJ, Sturt P (2002) Depth of processing in language comprehension: not noticing the evidence, *Trends in Cognitive Sciences*, Volume 6, Issue 9 , 1 September 2002, Pages 382-386

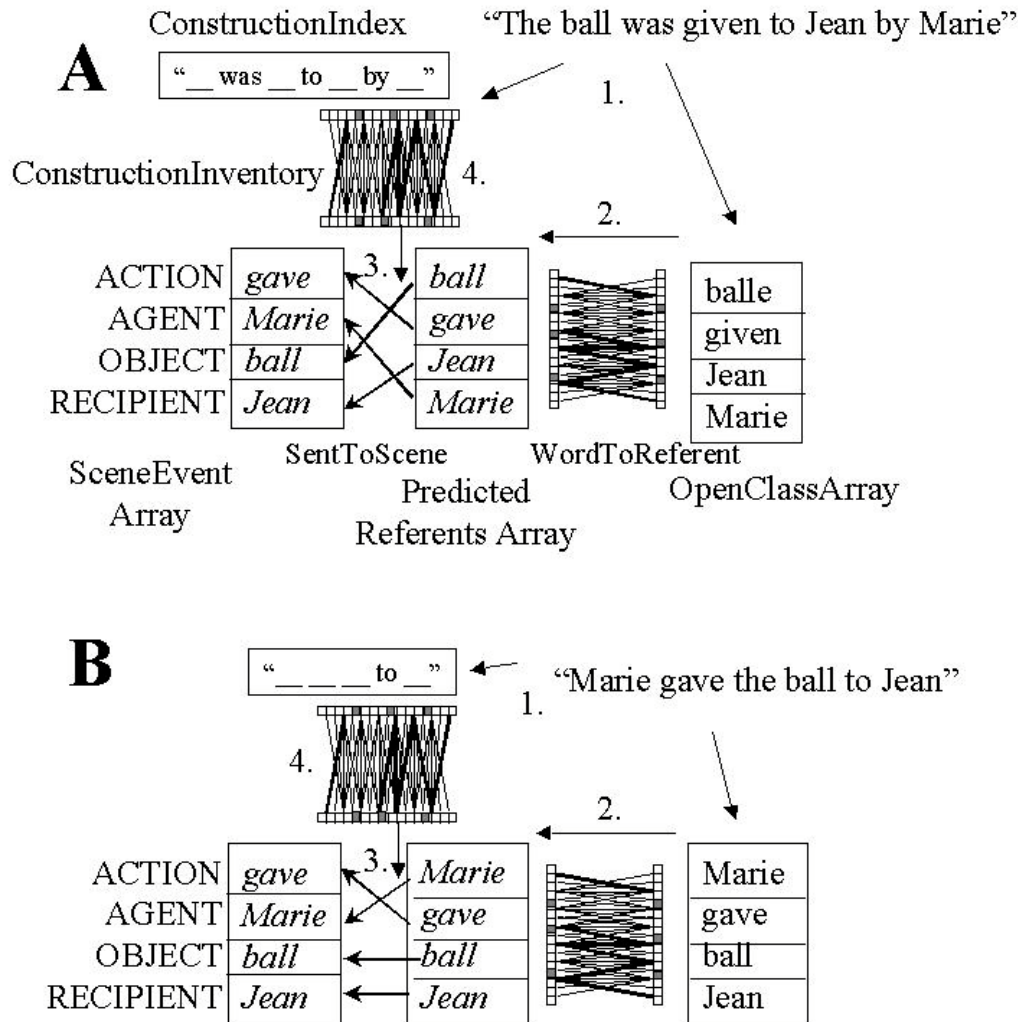


Figure 1. Structure-Mapping Architecture for language learning. **A.** (1) Open and closed class words respectively processed in the OpenClassArray (OCA) and ConstructionIndex. (2) Open class words in OCA are translated to their referents in the PRA via the WordToReferent mapping. (3) The PRA elements are then mapped onto their respective roles in the scene SEA by the SentenceToScene mapping, specific to each sentence type, retrieved from ConstructionInventory (4), via the ConstructionIndex that encodes the closed class function words that characterize each sentence type. **B.** Processing for the active grammatical construction. Note that the SentenceToScene mappings differ for the two grammatical construction types in A and B.

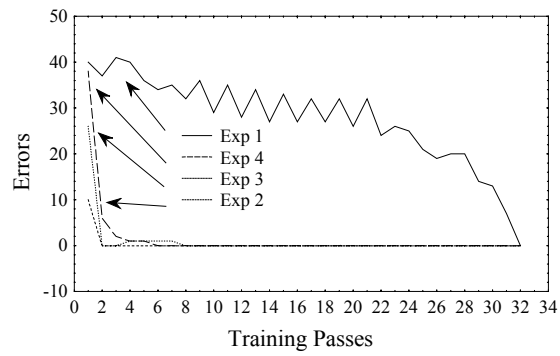


Figure 2. Evolution of interpretation errors during learning. Once the lexicon has been developed in Exp 1, interpretation errors are significantly reduced despite the increase of syntactic complexity in Exps 2-4. In these cases, error free performance is realized in 33, 3, 8 and 6 passes through the training corpus, respectively.

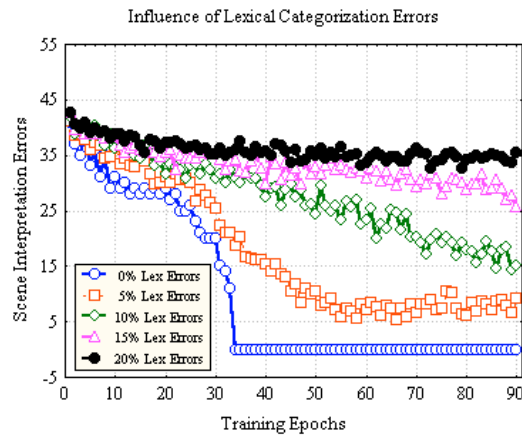


Figure 3. The effects of Lexical Categorization Errors (mis-categorization of an open-class word as a closed-class word or vice-versa) on performance (Scene Interpretation Errors) over Training Epochs. The 0% trace indicates performance in the absence of noise, with a rapid elimination of errors. The successive introduction of categorization errors yields a corresponding progressive impairment in learning. While sensitive to the errors, the system demonstrates a desired graceful degradation.