Available online at www.sciencedirect.com
SCIENCE DIRECT®
ELSEVIER
Artificial Intelligence 167 (2005) 31–61
Artificial Intelligence
www.elsevier.com/locate/artint

Learning to talk about events from narrated video in a construction grammar framework

Peter Ford Dominey *, Jean-David Boucher

Institut des Sciences Cognitives, CNRS 67 Blvd. Pinel, 69675 Bron Cedex, France

Received 7 July 2004; received in revised form 11 May 2005; accepted 2 June 2005

Available online 22 August 2005

Abstract

The current research presents a system that learns to understand object names, spatial relation terms and event descriptions from observing narrated action sequences. The system extracts meaning from observed visual scenes by exploiting perceptual primitives related to motion and contact in order to represent events and spatial relations as predicate-argument structures. Learning the mapping between sentences and the predicate-argument representations of the situations they describe results in the development of a small lexicon, and a structured set of sentence form-to-meaning mappings, or simplified grammatical constructions. The acquired grammatical construction knowledge generalizes, allowing the system to correctly understand new sentences not used in training. In the context of discourse, the grammatical constructions are used in the inverse sense to generate sentences from meanings, allowing the system to describe visual scenes that it perceives. In question and answer dialogs with naïve users the system exploits pragmatic cues in order to select grammatical constructions that are most relevant in the discourse structure. While the system embodies a number of limitations that are discussed, this research demonstrates how concepts borrowed from the construction grammar framework can aid in taking initial steps towards building systems that can acquire and produce event language through interaction with the world.

Keywords: Grammatical construction; Language acquisition; Event recognition; Language technology

* Corresponding author.
E-mail addresses: dominey@isc.cnrs.fr (P.F. Dominey), boucher@isc.cnrs.fr (J.-D. Boucher).
URL: http://www.isc.cnrs.fr/dom/dommenu-en.htm.

0004-3702/$ – see front matter
doi:10.1016/j.artint.2005.06.007

1. Introduction

As robotic systems become increasingly capable of complex sensory and motor functions, the ability to interact with them in an ergonomic, real-time and adaptive manner becomes an increasingly pressing concern. The goal of the current research is to test the hypothesis that simplified grammatical constructions—template-like mappings from sentence form to meaning that are part of the human progression to a full language capability [8,24,47]—can be learned by artificial systems in order to begin to address the problem of human-robot interaction in a limited domain. We first review results from a system that can adaptively acquire grammatical constructions based on training with human-narrated video events [13]. We then demonstrate how the system can then use these grammatical constructions to communicate with humans in a relevant and pragmatic manner.

1.1. An overview of the system

We begin with an overview of the physical setup of our vision-language platform, and will then present the information processing system and its motivation from studies of cognitive development [11–13]. The cognitive development issues are of potential interest because they provide clues as to how these information processing mechanisms have successfully been implemented in humans. Fig. 1A illustrates the physical setup in which the human operator performs physical events with toy blocks in the field of view of a color CCD camera. Fig. 1B illustrates a snapshot of the visual scene as observed by the image processing system. Fig. 1C illustrates the overall processing architecture.

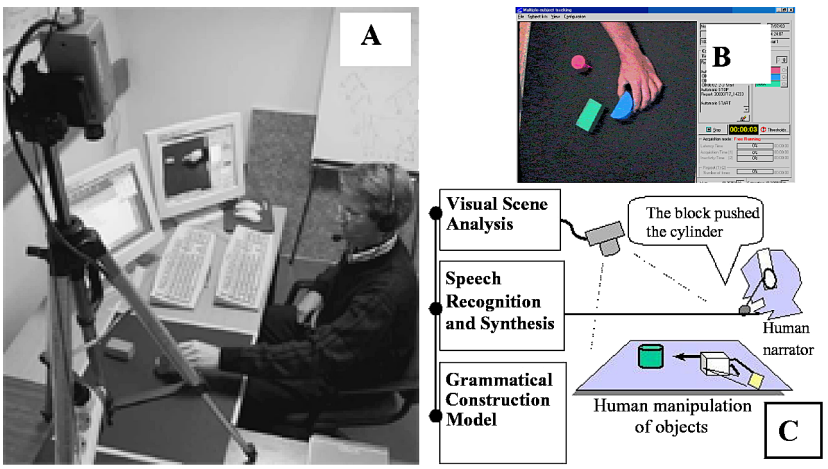

Fig. 1. Overview of human-robot interaction platform. A. Human user interacting with the blocks, narrating events, and listening to system-generated narrations. B. Snapshot of visual scene viewed by the CCD camera of the visual event processing system. C. Architecture Overview: Three distinct processors for vision processing, speech processing and dialog management, and grammatical construction learning and use in comprehension and production.

During the training of the system, the human operator performs physical events and narrates these events. An image processing algorithm extracts the meaning of the events in terms of action(agent, object, recipient) predicate-argument meaning descriptors. The event extraction algorithm detects physical contacts between objects, and then uses the temporal profile of contact sequences in order to categorize the events, based on a set of temporal event templates described below. The visual scene processing system is similar to related event extraction systems that rely on the characterization of complex physical events (e.g., give, take, stack) in terms of composition of physical primitives such as contact, support and attachment (e.g., [41,43]). Together with the event extraction system, a commercial speech to text system (IBM ViaVoice^{TM}) was used, such that each narrated event generated a well-formed ⟨sentence, meaning⟩ pair.

The ⟨sentence, meaning⟩ pairs were provided as training input to the learning model whose architecture is depicted in Fig. 2. We briefly introduce the model here, with a more detailed description and example of processing in Section 3. The model integrates two functional aspects borrowed from construction grammar [24] in order to yield a

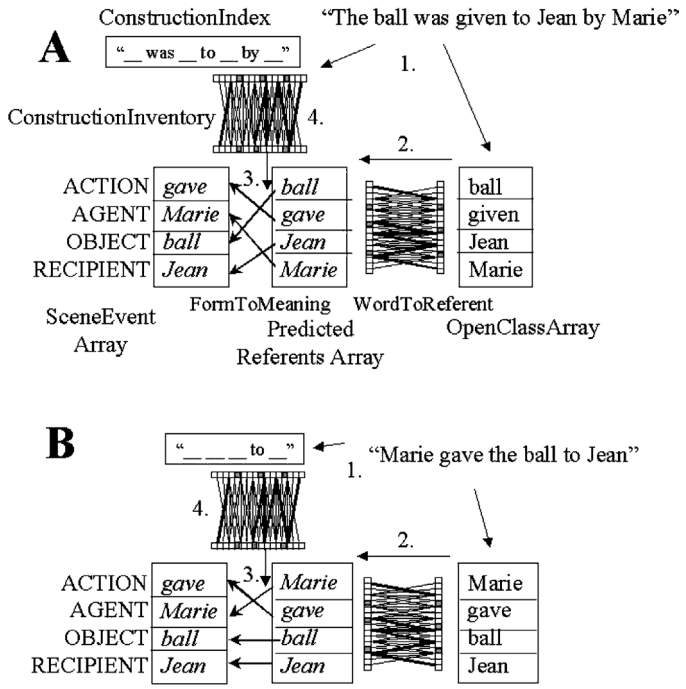

Fig. 2. Grammatical construction architecture. Processing of active and passive sentence types in A, B, respectively. 1. On input, Open-class words populate the Open-class Array (OCA), and closed-class words populate the Construction Index. Visual Scene Analysis populates the Scene Event Array (SEA) with the predicate-argument representation of the action. 2. Words in OCA are translated to Predicted Referents via the WordToReferent mapping to populate the Predicted Referents Array (PRA). 3. PRA elements are mapped onto their roles in the Scene Event Array (SEA) by the FormToMeaning mapping, specific to each sentence type. 4. This mapping is retrieved from Construction Inventory, via the ConstructionIndex that encodes the closed-class words that characterize each sentence type.

construction-based language learning capability. The essential problem the model is designed to address is that of mapping grammatical structure of sentences onto the semantic structure of their meanings. As illustrated in Fig. 2A and B, the problem of this mapping is not trivial, because a given language consists of a large ensemble of possible mappings. The first principle inherent in the model is that instead of representing ⟨sentence, meaning⟩ mappings in terms of a generative grammar, they are represented directly in a structured inventory of grammatical constructions that are nothing more than these mappings [24,47]. Growing evidence both from studies of human language development [46,47], and adult processing [21,32,38] indicate that a substantial component of language behavior can be accounted for in this manner. That is, a large part of language production and comprehension is based on the re-use (including recombination) of existing templates, in a context in which the templates (i.e., grammatical constructions) can be learned by straightforward mechanisms as illustrated in Fig. 2. This does not exclude the existence of truly generative mechanisms for construction and decoding new grammatical forms. However, for our purposes, in the domain of human-robot interaction, the ability to rapidly acquire relevant constructions in relatively restricted domains should prove quite useful. In addition, it is crucial to note that the use of these template-based constructions represents a clearly specified phase in the development of the human language capability [47].

If the language capability consists of a structured inventory of grammatical constructions, then the problem remains concerning how this inventory is managed. This is where the second important principle of developmental linguistics comes in: the cue competition hypothesis of Bates et al. [1]. They propose that across languages, there is a limited set of possible cues including word ordering regularities and the use of grammatical function words (e.g., to, by, from, that, was), that code the argument structure of sentences, that allows the determination of "who did what to whom". Thus, as illustrated in Fig. 2, the ensemble of closed-class words together form the "Construction Index" that serves as an index into an associative memory that stores the appropriate transformations. This memory store is referred to as the ConstructionInventory in Fig. 2.

Once a set of constructions has been learned in the comprehension context, it can then—with some additional processing—be used in the inverse sense for expressing meaning. In this context, the relevance of distinct constructions that describe the same event structure becomes apparent. As we will see below, the use of the active and passive as illustrated in Fig. 2 becomes relevant depending on whether it is the agent or the object of the event that is in the focus of the discourse.

1.2. Assumptions, scope of the model and limitations

The principal assumption behind the model is that, simply stated, a certain degree of language learning performance can be derived from a construction grammar framework that uses a template-based system in which lexical word orderings in sentences are mapped to predicate/argument orderings in meanings. These mappings are stored in a memory (the ConstructionInventory) that uses the configuration of function words as an index (the ConstructionIndex) into that memory in order to store and retrieve these mappings. In this context, the system is based on two aspects of the functionalist construction grammar paradigm. First, these templates are related to (though do not completely capture the essence

of) grammatical constructions, as "form to meaning" mappings (e.g., [24]). Second, the use of the function word (closed-class word) configuration as an index into a memory of these stored constructions is an adaptation of the "cue competition" theory developed by Bates and MacWhinney (e.g., [1]). In this theory, across languages, different cues including lexical category, word order, prosody and grammatical marking as either free (i.e., function words) or bound morphemes are used in different combinations in order to code grammatical structure.

The benefit of this approach is that constructions can be easily learned and used in order to generate and understand novel sentences that employ one of the previously learned construction types. That is, after a construction has been learned, it can provide a basis for systematic generalization. Thus, once the system can understand "John pushed Bill" and "John gave the ball to Bill" it can also understand "Bill pushed John", and "Bill pushed the ball to John". The principal limitation is that the system does not generalize in a *compositional* manner, i.e. in order to accommodate a new construction type, it must first have the opportunity to learn the form to meaning mapping given a well-formed ⟨sentence, meaning⟩ input.

Interestingly, while this is a severe limitation in the long term, it appears to be a developmental step for human children (of about 2 years of age) on their way to more adult-like generative performance [8,47]. Likewise, providing a formal explanation of how constructions are combined in a generative manner remains an open issue within the construction grammar community [8,47]. But the idea is that (1) children first imitatively learn concrete linguistic expressions, then (2) learn to use these as abstract constructions that can take arguments, and finally (3) learn to combine these structures in a generative manner [46,47]. The current model corresponds to the second phase, and thus, part of the goal of the current study is to see how far can we get with this approach, and what are the next steps beyond it.

Given this introduction, the following sections provide more detail on the extraction of meaning, mapping grammatical constructions to meaning, and then the use of this knowledge in human-machine interaction.

2. Extraction of meaning

In a developmental context, Mandler [29] 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, attachment [45] the infant could construct progressively more elaborate representations of visuospatial meaning. Thus, the physical event "collision" is a form of the perceptual primitive "contact". Kotovsky and Baillargeon [27] 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 [41] has demonstrated that force dynamic primitives of contact, support, 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.

2.1. Visual scenes and event analysis

In the current study we take an approach that is similar to that of Siskind [41], in which event categorization is performed based on detection of perceptual primitives. Based on the temporal sequence of contacts extracted from a given video sequence the system generates the corresponding event description in the format *event*(*agent*, *object*, *recipient*).

Single event labeling. Events are defined in terms of contacts between elements. A contact 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. Based on these parameters of contact, scene events are recognized as depicted in Fig. 3, and described below.

Touch(*agent*, *object*): 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.

Push(agent, object): Similar to touch, in which the contact duration is superior or equal to *touch_duration* and inferior to *take_duration* (5 sec), and (b) the object is displaced.

Take(*agent, object*): 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.

Take(*agent, object, source*): Multiple contacts, as the agent takes the object from the source. Similar to take(agent, object), with an optional second contact between agent and source in which (a) the duration of the contact is inferior to

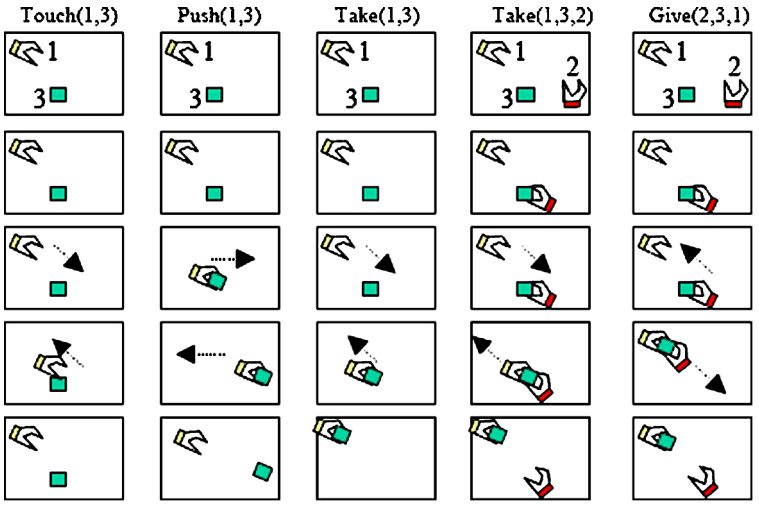

Fig. 3. Temporal profile of contacts defining different event types.

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.

Give(*agent, object, recipient*): Multiple contacts as agent takes object, then initiates contact between object and recipient, and finally terminates the agent-object contact.

These event labeling templates form the basis for a template matching algorithm that labels events based on the contact list, similar to the spanning interval and event logic of Siskind [41].

Complex "hierarchical" event labeling. The events described above are simple in the sense that there have no hierarchical structure. This imposes serious limitations on the syntactic complexity of the corresponding sentences [19,30]. The sentence "The block that pushed the moon was touched by the triangle" illustrates a complex event that exemplifies this issue. The corresponding compound event will be recognized and represented as a pair of temporally successive simple event descriptions, in this case: *push*(*block*, *moon*), and *touch*(*triangle*, *block*). The "block" serves as the link that connects these two simple events in order to form a complex hierarchical event.

2.2. Attention, relevance and spatial relations

Part of our implicit assumption is that certain perceptual primitives, e.g. physical contact, will generate an attentional drive for the perceptual event processing system. In this section, we consider the analogous question of perceptual primitive processing in the domain of spatial relation perception. The point is that the use of perceptual primitives to generate predicate-argument semantic representations should extend to spatial relations, thus allowing spatial relations and sentences that describe them to enter into the ⟨sentence, meaning⟩ format of the grammatical construction model. Thus, Quinn et al. [34] and Quinn [33] have demonstrated that by the age of 6–7 months, infants can learn binary spatial relations such as left, right, above, below in a generalized manner, as revealed by their ability to discriminate in familiarization-test experiments. That is, they can apply this relational knowledge to scenes with new objects in the appropriate spatial relations.

In theory, the predicate-argument representation for event structure that we have described above can provide the basis for representing spatial relations in the form Left(X,Y), Above(X,Y) etc. where X is the target object that holds the spatial relation with the referent object Y. That is, Left(X,Y) corresponds to "X is left of Y". In order to extract spatial relations from vision we return to the visual processing system described above. Based on the observations of Quinn [33] we can consider that by 6–7 months, the perceptual primitives of Relation(X,Y) are available, where Relation corresponds to Left, Right, Above and Below.

One interesting problem presents itself however, related to referential ambiguity. Fig. 4 illustrates the spatial configuration of objects after a human user has placed the cylinder in its current position and said "The cylinder is below the triangle". Given this image, any one of the four objects could be the subject of the relation, and any one of the remaining three could be the referent, thus yielding 12 possible relations. The problem then is one of referential uncertainty, or "what is the speaker talking about?"

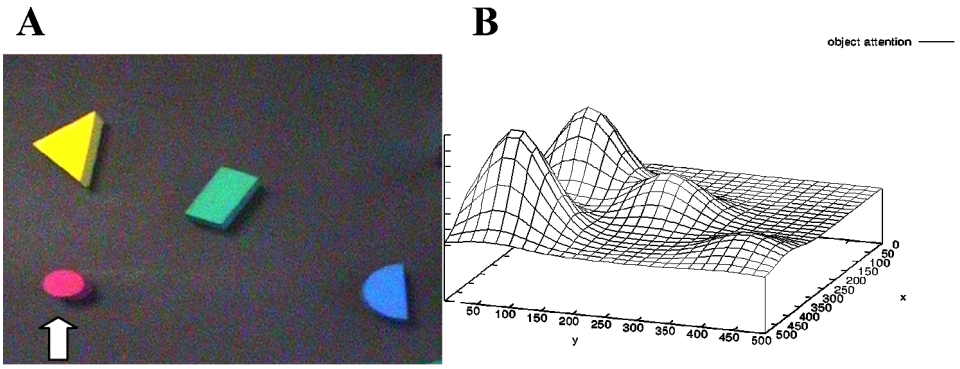

Fig. 4. Spatial Attention for Relation Selection. The human user shows the robot a spatial relation and describes it. How does the robot know which of the multiple relations is the relevant one? A. The cylinder (lower left) has been moved into its current position, and now holds spatial relations with the three other objects. B. Attentional selection based on parameters of (1) minimal distance from the target object and (2) minimal angular distance from the four principal directions (above, below, left, right). In this case, the most relevant relation (indicated by the height of the two highest peaks) is Below(Cylinder, Triangle). The horizontal and vertical axes correspond to the spatial horizontal and vertical axes in A, and the height axis of the 3D surface corresponds to the value of the attention function at that spatial location.

Tomasello [47] clearly emphasizes the crucial role of shared attention between the speaker and listener in solving this referential uncertainty. One of the most primitive forms of attention is related to the detection of movement—and the act of "showing" something almost always involves either pointing to picking up and moving the object. In this context Kellman et al. [26] demonstrated that as early as 16 weeks, infants are sensitive to object motion that can provide the basis for object identification.

Thus, Dominey and Boucher [13] employed a simple attention mechanism based on motion to select the last object in motion (cylinder in the example of Fig. 4) as the target object, but the intended referent for the "below" relation could be any one of the multiple other objects, and so the problem of referential ambiguity must still be resolved. We hypothesize that this redundancy is resolved based on two perceptual parameters. First, spatial proximity, or distance from the target will be used. That is, the observer will give more attentional preference to relations involving the target object and other objects that are closest to it. The second parameter is the angular "relevance" of the relations, quantified in terms of the angular distance from the cardinal positions *above*, *below*, *left* and *right*. Fig. 4B represents the application of this perceptual attention mechanism that selects the relation Below(Cylinder, Triangle) as the most relevant, revealed by the height of the peak for the triangle in Fig. 4B. Below we confront these embodied hypotheses with the behavior of human subjects who "teach" the system.

3. Sentence to meaning mapping

Once meaning is extracted from the scene, the significant problem of mapping sentences to meanings remains. The nativist perspective on this problem holds that the ⟨sentence,

meaning⟩ data to which the child is exposed is highly indeterminate, and under specifies the mapping to be learned. This "poverty of the stimulus" claim is a central argument for the existence of a genetically specified universal grammar (UG), such that language acquisition consists of configuring the UG for the appropriate target language [5]. In this framework, once a given parameter is set, its use should apply to new constructions in a generalized, generative manner.

An alternative functionalist perspective on learning holds that learning and general cognitive mechanisms play a much more central role in language acquisition. The infant develops an inventory of grammatical constructions as mappings from form to meaning [24]. These constructions are initially rather fixed and specific, and later become generalized into a more abstract compositional form employed by the adult [46,47]. In this context, construction of the relation between perceptual and cognitive representations and grammatical form plays a central role in learning language (e.g., [18,19,28,29,45]).

These issues of learnability and innateness have provided a rich motivation for simulation studies that have taken a number of different forms. Elman [17] 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., [44]). From the "grounding of language in meaning" perspective (e.g., [18,19,24,28]), Chang and Maia [7] exploited the relations between action representation and simple verb frames in a construction grammar approach, and Cottrel et al. [9] associated sequences of words with simple image sequences. In 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 processing of the next embedded clause in a recursive manner.

In a more generalized approach, Dominey [10] exploited the regularity that sentence to meaning mapping is encoded in all languages by a small set of cues including word order and grammatical marking (bound or free) [1]. That model was based on the functional neurophysiology of cognitive sequence and language processing and an associated neural network model that has been demonstrated to simulate interesting aspects of infant [16] and adult language processing [14].

Our approach is thus based on the cross-linguistic observation that open-class words (e.g. nouns, verbs, adjectives and adverbs) are assigned to their thematic roles based on word order and/or grammatical function words or morphemes [1]. The mapping of sentence form onto meaning [24] takes place at two distinct levels: Open-class words are associated with individual components of event descriptions, and their functional roles within scene events are determined as a function of grammatical structure based on the constellation of closed-class words in the sentence (Fig. 2).

With respect to open- and closed-class words, newborn infants are sensitive to the perceptual properties that distinguish these two categories [39], and in adults, these categories are processed by dissociable neurophysiological systems [3]. Similarly, artificial neural networks can also learn to make this function/content distinction [2,31]. Thus, for the speech input that is provided to the learning model, open- and closed-class words are directed to separate processing streams that preserve their order and identity, as indicated in

Fig. 2. Future research should exploit the distributional statistics of open vs. closed categories so that this distinction could be learned from the input.

The first level of the lexical mapping of words to meaning has been addressed by Siskind [40], Roy and Pentland [37] and Steels [42] and we treat it here in a relatively simple but effective manner. Our principle interest lies more in the second level of mapping between scene and sentence structure, in an approach related to that of the DESCRIBER system of Roy [36] that we will address in the discussion.

3.1. Model overview

We first present an overview of the model, and define the representations and functions of each component of the model using the example sentence "The ball was given to Jean by Marie," and the corresponding meaning "gave(Marie, Ball, John)" in Fig. 2A. Words in sentences, and elements in the scene are coded as single bits in respective 25-element vectors, and sentences length is constrained such that they contain 6 or less open-class words. On input, open-class words (e.g., ball, given, Jean, Marie) are stored in the Open-class Array (OCA), which is thus an array of 6×25 element vectors, corresponding to a capacity to encode up to 6 open-class words per sentence. Open-class words correspond to single-word noun or verb phrases, and determiners do not count as closed-class words. Thus in Fig. 2 the elements in the OpenClassArray actually correspond to 25 element vectors with a single bit on, each bit corresponding to the appropriate open-class word that it encodes. Closed-class words (e.g., was, to, by, and non-lexicalized begin-sentence and end-sentence markers) are encoded in the Construction Index, a 25 element vector, by an algorithm described below that preserves the identity and input order of the closed-class elements.

The meaning component of the ⟨sentence, meaning⟩ pair is encoded in a predicate-argument format in the Scene Event Array (SEA). The SEA is also a 6×25 array. In this example the predicate is *gave*, and the arguments corresponding to agent, object and recipient are *Marie, Ball, John*. The SEA thus encodes one predicate and up to 5 arguments, each as a 25 element vector. During learning, complete ⟨sentence, meaning⟩ pairs are provided as input. In subsequent testing, given a novel sentence as input, the system can generate the corresponding meaning.

The first step in the sentence-meaning mapping process is to extract the meaning of the open-class words and store them in the Predicted Referents Array (PRA). The word meanings are extracted from the real-valued WordToReferent associative memory matrix that encodes learned mappings from input word vectors to output meaning vectors. The second step is to determine the appropriate mapping of the separate items in the PredictedReferentsArray onto the predicate and argument positions of the SceneEventArray. This is the "form to meaning" mapping component of the grammatical construction. Up to 6 PRA items are thus mapped onto their 6 corresponding roles in the Scene Event Array (SEA) by the FormToMeaning mapping, specific to each construction type. FormToMeaning is thus a 6×6 real-valued matrix. This mapping is retrieved from ConstructionInventory, based on the ConstructionIndex that encodes the closed-class words that characterize each sentence type. The ConstructionIndex is a 25 element vector, and the FormToMeaning mapping is a 6×6 real-valued matrix, corresponding to 36 real values. Thus the Con-

structionInventory is a 25×36 real-valued matrix that defines the learned mappings from ConstructionIndex vectors onto 6×6 FormToMeaning matrices. Note that in 2A and 2B the ConstructionIndices are different, thus allowing the corresponding FormToMeaning mappings to be handled separately. Given this model overview we now address the detailed processing.

3.2. Word meaning—Eq. (1)

Eq. (1) describes the update procedure for the real-valued WordToReferent matrix that defines the mapping from word vectors in the OpenClassArray to referent vectors in the PredictedReferentsArray and SceneEventArray. For all k, m, $1 \leqslant k \leqslant 6$, corresponding to the maximum number of words in the open-class array (OCA), and $1 \leqslant m \leqslant 6$, corresponding to the maximum number of elements in the scene event array (SEA). For all i and j, $1 \leqslant i, j \leqslant 25$, corresponding to the word and scene item vector sizes, respectively.

The WordToReferent matrix values are initialized with a set of uniform weights, and after each update, the weights are normalized to preserve the total weight, thus avoiding a learning-related weight saturation, and providing a form of competition. Initially, the update procedure associates every open-class word with every referent in the current scene. This exploits the cross-situational regularity [40] that a given word will have a higher co-occurrence with the referent to which it refers than with other referents. However, for a given ⟨sentence, meaning⟩ pair, a particular element in the OpenClassArray is grammatically associated with only one specific element in the SceneEventArray. For example, in Fig. 2A, the first element of the OCA (ball) is associated only with the third (object) element of the SEA. This grammatical information is encoded in the FormToMeaning matrix. Thus, in Eq. (1), the "*Max(α, FormToMeaning(m, k))" term allows this FormToMeaning information to be used so that only grammatically appropriate associations are learned, corresponding to a zero value of α in Eq. (1). Thus, initially, α is set to 1 to allow cross-situational learning. That is, all the open class words are associated with all of the possible referents in the scene. This provides the basis for acquisition of limited FormToMeaning mapping. Once this learning has occurred, we can use a more appropriate association strategy, associating a given open class element (i.e., the contents of a given element of the OpenClassArray) with the contents of the PredictedReferentsArray element indicated by the FormToMeaning mapping. Again, to do this, α is set to 0 to exploit this "syntactic" bootstrapping described above in the example for "ball". Dominey [10] provides a detailed analysis of the interaction between acquisition of lexical and grammatical knowledge in this context.

$$\begin{aligned}\text{WordToReferent}(i, j) &= \text{WordToReferent}(i, j) \\ &+ \text{OCA}(k, i) * \text{SEA}(m, j) * \text{Max}\big(\alpha, \text{FormToMeaning}(m, k)\big). \end{aligned} \tag{1}$$

3.3. Learning the mapping from sentence to meaning

Learning this mapping can be characterized in two successive steps that involve determining this mapping for the current sentence, and then storing this mapping for future use, respectively. Eq. (2) describes the debut of the first step, which consists in retrieving

for each word in the Open-class Array the corresponding meaning that is encoded in the WordToReferent mapping, and then storing these meaning vectors in the Predicted Referents Array, preserving their original order from the OCA.

$$\mathrm{PRA}(m, j) = \sum_{i=1}^{n} \mathrm{OCA}(m, i) * \mathrm{WordToReferent}(i, j). \tag{2}$$

Now, given PRA and the input meaning coded in the SceneEventArray, we can determine the correspondence between them. That is, the FormToMeaning mapping for the corresponding ⟨sentence, meaning⟩ input can be extracted simply by matching elements in the SceneEventArray with their correspondent in the PredictedReferentsArray as can be seen in Fig. 2. Eq. (3) describes the calculation of the 6×6 matrix FormToMeaningCurrent, which corresponds to this mapping from meaning slots in the PredictedReferentsArray onto event-role slots in the SceneEventArray. In the example of Fig. 2A this 6×6 matrix corresponds to the following mappings PRA(1) to SEA(3) for *ball*, PRA(2) to SEA(1) for *gave*, PRA(3) to SEA(4) for *Jean*, and PRA(4) to SEA(2) for *Marie*.

$$\mathrm{FormToMeaningCurrent}(m, k) = \sum_{i=1}^{n} \mathrm{PRA}(k, i) * \mathrm{SEA}(m, i). \tag{3}$$

Given this FormToMeaningCurrent provided by the first step, the system should associate this mapping with the corresponding grammatical construction type in the second step so that it can later be retrieved and used. Recall that each construction type will have a unique constellation of closed-class words and/or bound morphemes [1] that can be coded in a ConstructionIndex, illustrated in Fig. 2A and B. Eq. (4) describes how this coding takes place. The ConstructionIndex is a 25 element vector that should provide a unique identifier for each distinct grammatical construction. When a function word is encountered during sentence processing, it is encoded as a single bit in a 25 element FunctionWord vector. 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. This algorithm was developed to meet the requirement that the ConstructionIndex should uniquely identify each distinct construction type, and sentences of the same construction type should have the same ConstructionIndex.

$$\begin{aligned}\mathrm{ConstructionIndex} = {} & f_{\mathrm{Shift}}(\mathrm{ConstructionIndex}, \mathrm{FunctionWord}) \\ & + \mathrm{FunctionWord}.\end{aligned} \tag{4}$$

Finally, the system must establish the link between the ConstructionIndex and the corresponding FormToMeaning mapping. Given the FormToMeaningCurrent mapping for the current sentence, we can now associate it in the ConstructionInventory with the corresponding ConstructionIndex for that sentence. This process is expressed in Eq. (5) that describes how the ConstructionInventory matrix is updated during learning. This is a real-valued matrix that is initialized with a uniform weight distribution that is normalized after each update. Note that the quality of FormToMeaningCurrent will depend on the quality of acquired word meanings in WordToReferent. Thus, learning these constructions requires a

minimum baseline of semantic knowledge. In parallel, as noted during the description of Eq. (1), the semantic knowledge can be influenced by the quality of grammatical knowledge in a synergistic manner.

$$\begin{aligned}\text{ConstructionInventory}(i,j) &= \text{ConstructionInventory}(i,j) \\ &+ \text{ConstructionIndex}(i) * \text{FormToMeaningCurrent}(j). \end{aligned} \tag{5}$$

Now that this two-step learning process of (1) extracting FormToMeaningCurrent, and (2) associating it with the ConstructionIndex in the ConstructionInventory has occurred, how does this learning allow the interpretation of new sentences? Given a new sentence we calculate its ConstructionIndex and use this to extract the FormToMeaning mapping from the learned ConstructionInventory as illustrated in Eq. (6). Note that in Eqs. (5) and (6) we have linearized FormToMeaningCurrent and FormToMeaning from 2 to 1 dimensions to make the matrix multiplication more transparent. Thus index j varies from 1 to 36 corresponding to the 6×6 dimensions of FormToMeaningCurrent.

$$\text{FormToMeaning}(j) = \sum_{i=1}^{n} \text{ConstructionInventory}(i,j) * \text{ConstructionIndex}(i). \tag{6}$$

To accommodate the dual scenes for complex events corresponding to "The block that pushed the moon was touched by the triangle" as described in Section 2.1, Eqs. (3) and (5)–(7) are instantiated twice each along with the corresponding data structures, to represent the two components of the dual scene. In the case of simple scenes, the second component of the dual scene representation is null.

3.4. Evaluating performance after learning

We evaluate performance by using the WordToReferent and FormToMeaning knowledge to construct for a given input sentence the "predicted scene". That is, the model will construct an internal representation of the scene that should correspond to the input sentence. This is achieved by first converting the OpenClassArray into its corresponding scene items in the PredictedReferentsArray as specified in Eq. (2). The mapping from PredictedReferentsArray to SceneEventArray must then occur. This involves first the computation of the ConstructionIndex as specified in Eq. (3). Next, the FormToMeaning mapping is extracted from the ConstructionInventory with this ConstructionIndex as defined in Eq. (6). The referents are then re-ordered into the proper scene representation via application of the FormToMeaning transformation as described in Eq. (7).

$$\text{PSA}(m,i) = \text{PRA}(k,i) * \text{FormToMeaning}(m,k). \tag{7}$$

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. It is important to note that the FormToMeaning mapping is independent of the values of meanings in the PredictedReferentsArray, and thus for this reason a learned construction can generalize to new sentences, allowing learned nouns to occur in roles not used during learning, and learned verbs to take different argument structures from those used in learning.

4. Experimental results for event processing in ideal conditions

The results of the experiments reviewed in this section document robustness of the construction model under highly controlled conditions as observed by Dominey and Boucher [13].

4.1. The training data

To generate data for training 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 video camera above the surface provides a video image that 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 that is subsequently processed for event analysis (described above). The simultaneous narration of the ongoing events is processed by a commercial speech-to-text system (IBM ViaVoice^{TM}). 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 tested in the following experiments.

4.1.1. Learning of active forms for simple events

In this experiment, sentences were generated in a "scripted" manner, using the active transitive and active dative forms (terminology from [4]) as illustrated.

1. Active: The block pushed the triangle.
2. Dative: The block gave the triangle to the moon.

Seventeen ⟨sentence, meaning⟩ were generated that employed the 5 different events (touch, push, take, take-from, give), and narrations in the active voice, corresponding to the grammatical forms 1 and 2. The model was trained for 32 passes through the 17 ⟨sentence, scene⟩ pairs for a total of 544 ⟨sentence, scene⟩ pairs. During the first 200 ⟨sentence, scene⟩ pair trials, α in Eq. (1) was 1 (i.e., no syntactic bootstrapping before syntax is acquired), and thereafter it was 0. This was necessary in order to avoid the random effect of syntactic knowledge on semantic learning in the initial learning stages. The trained system displayed error free performance for all 17 sentences. In a subsequent generalization test using sentences that had not been used in training generated from the same constructions, the learned capability transferred to these new sentences with no errors.

4.1.2. Passive forms

This experiment examined learning active and passive grammatical forms, employing grammatical forms 1–4. Word meanings were used from Experiment A (i.e., the WordToReferent matrix was retained), so only the structural FormToMeaning mappings were learned.

3. Passive: The triangle was pushed by the block.

4. Dative Passive: The moon was given to the triangle by the block.

Seventeen new ⟨sentence, meaning⟩ pairs were generated with active and passive grammatical forms for the narration. Within 3 training passes through the 17 sentences (51 ⟨sentence, scene⟩ pairs), error free performance was achieved, with confirmation of error free generalization to new untrained sentences of these types. The rapid learning indicates the importance of lexicon in establishing the form to meaning mapping for the grammatical constructions.

4.1.3. Relative forms for complex events

Here we considered complex scenes narrated by relative clause sentences. Eleven complex ⟨sentence, scene⟩ pairs were generated with narration corresponding to the grammatical forms indicated in 5–10:

5. The block that pushed the triangle touched the moon.
6. The block pushed the triangle that touched the moon.
7. The block that pushed the triangle was touched by the moon.
8. The block pushed the triangle that was touched by the moon.
9. The block that was pushed by the triangle touched the moon.
10. The block was pushed by the triangle that touched the moon.

After presentation of 88 ⟨sentence, scene⟩ pairs, the model performed without error for these 6 grammatical forms, and displayed error-free generalization to new sentences that had not been used during the training for all six grammatical forms.

4.1.4. Combined test with and without lexicon

A total of 27 ⟨sentence, scene⟩ pairs, extracted from those used in Experiments B and C, were employed that exercised the ensemble of grammatical forms 1–10 using the learned WordToReferent mappings. After six training epochs (162 ⟨sentence, scene⟩ pairs) the model performed and generalized without error. When this combined test was performed without the pre-learned lexical mappings in WordToReferent, the system failed to converge, illustrating the advantage of following the developmental progression from lexicon to simple to complex grammatical structure.

4.1.5. Robustness to noise

In the above experiments, the ⟨sentence, scene⟩ pairs employed were restricted to be well-formed, such that each sentence accurately described the accompanying scene. The rapid learning is due in part to this simplification. Here we consider the introduction of noise. 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 [39] 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 cat-

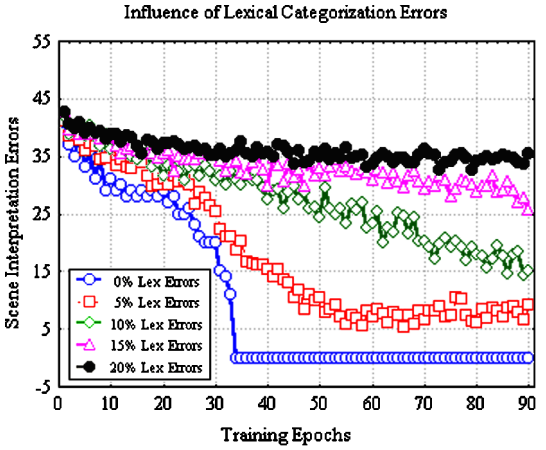

Fig. 5. The effects of Lexical Categorization Errors (miscategorization of an open-class word as a closed-class word or vice-versa) on performance (Scene Interpretation Errors) over Training Epochs. The 0% trace indicates performance in the absence of noise, with a rapid elimination of errors. The successive introduction of categorization errors yields a corresponding progressive impairment in learning. While sensitive to the errors, the system demonstrates a desired graceful degradation.

egory and processed as such (e.g., an open-class word being processed as a closed-class word, or vice-versa). Fig. 5 illustrates the performance of the model with random errors of this type introduced at levels of 0 to 20 percent errors.

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 asses 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 ($\sim$ 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 noisy input. This will be further illustrated in the experiments under less constrained conditions described below.

4.1.6. Learning an 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. We have subsequently demonstrated that the model can accommodate 38 different grammatical constructions that combine verbs with two or three arguments, active and passive forms and relativisation, along with additional sentence types as exemplified by 1–3 below [13,15]

(1) John took the key and opened the door.
(2) The boy said that the dog was chased by the cat.
(3) The block said that it pushed the cylinder.

The consideration of these sentence types requires us to address how their meanings are represented. Sentences corresponding to (1) are represented by the two corresponding events, e.g., *took*(*John*, *key*), *open*(*John*, *door*) for the example above. Sentences corresponding to (2) are represented, for example, as *said*(*boy*), *chased*(*cat*, *dog*). This assumes indeed, for verbs that take sentential arguments (e.g., said, saw), that the meaning representation includes the second event as an argument to the first. Finally, for sentences of type (3), 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". Thus, in the current implementation semantic differences between different types of multiple-event meanings are not represented. For example the meaning representations of relative clauses do not differ from those of conjoined sentences. Thus "The ball that broke the window fell on the floor" and "The ball broke the window and fell on the floor" would have the same meaning representation. Ideally there should be an additional component within the meaning representation that would capture the semantic difference and be part of the construction, but for the current studies this was not necessary.

For this testing, the ConstructionInventory is implemented as a lookup table in which the ConstructionIndex is paired with the corresponding FormToMeaning mapping during a single learning trial. In the initial experiments the ConstructionInventory was an associative memory in the form of a real-valued matrix that maps ConstructionIndex vectors to FormToMeaning matrices. This allows a more realistic study of the interaction of this grammatical learning and lexical learning in the WordToReferent matrix. However, the use of the lookup table is computationally much more direct and rapid.

Based on the tenets of the construction grammar framework [24], 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, and demonstrated generalization to new sentences that it was not trained on. A sample of the construction types is presented in Table 1.

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.

In summary, the results reviewed in this section demonstrate that the model can learn ⟨sentence, meaning⟩ mappings based on a given input corpus, and can then use these learned constructions to understand new sentences made from the same vocabulary, but using sentences not seen in the training corpus. The model can operate on input data derived from the vision processing platform, as well as extended data that uses a much larger

Table 1
Sample sentences with their meanings (left column) and the corresponding abstract grammatical constructions (right column)

Example sentences and meanings	*Grammatical constructions*
1. The block pushed the cylinder. Push(block, cylinder)	1. Agent verb object. (Active) Verb(agent, object)
2. The cylinder was pushed by the block. Push(block, cylinder)	2. Object was verbed by agent. (Passive) Verb(agent, object).
3. The block gave the cylinder to the moon. Give(block, cylinder, moon)	3. Agent verbed object to recipient. (Dative) Verb(agent, object, recipient)
4. The cylinder was given to the moon by the block. Give(block, cylinder, moon)	4. Object was verbed to recipient by agent. (Dative passive) Action1(agent1, object2, recipient3).
Dual-event relative constructions	
6. The block that pushed the cylinder touched the moon. push(block, cylinder), Touch(block, moon)	6. Agent1 that verb1ed object2 verb2ed object3. Action1(agent1,object2), Action2 (agent1, object3)
7. The block was pushed by the moon that touched the cylinder. Touch(moon, cylinder), Push(moon, block)	7. Object3 was action2ed by agent1 that action1ed object2. Action1(agent1,object2), Action2 (agent1, object3)
17. The cat was given from the dog to the block that pushed the cylinder. Push(block, cylinder), Give(dog, cat, block)	17. Ag3 act2ed obj4 to recip1 that act1ed obj2 Action1(agent1,object2), Action2 (agent3,object4,recipient1)
18. The cylinder that was pushed by the block gave the cat to the dog. Push(block, cylinder), give(cylinder, cat, dog).	18. Obj4 was act2ed from ag3 to recip1 that act1ed obj2 Action1(agent1,object2), Action2 (agent3, object4,recipient1)
Dual-event conjoined constructions	
27. The block pushed the cylinder and the moon. Push(block, cylinder), Push(block, moon)	27. Agent1 action1 object1 and object. Action1(agent1, object1), Action1(agent1, object2)
28. The block and the cylinder pushed the moon. Push(block, moon), Push(cylinder, moon)	28. Agent1 and agent3 action1ed object2. Action1(agent1, object2), Action1(agent3, object2)
29. The block pushed the cylinder and touched the moon. Push(block, cylinder), Touch(block, moon).	29. Agent1 action1ed object2 and action2 object3. Action1(agent1, object2), Action2(agent1, object3)
30. The moon and the block were given to the cylinder by the cat. Give(cat, moon, cylinder), Give(cat, block, cylinder).	30. Object2 and object3 were action1ed to recipient4 by agent1. Action1(agent1, object2, recipient4), Action1(agent1, object3, recipient4)

variety of construction types. Finally, the system demonstrates the ability to cope with a certain degree of noise in the input. The ability to accommodate different semantic domains (spatial relations) and input derived from less constrained situations with naïve users will now be explored.

5. Experimental results for spatial relations

The results reviewed in this section demonstrate the ability of the grammatical construction concept to extend to learning simple spatial relations [13]. Fisher [22] suggested that once a mechanism for mapping grammatical structure to predicate-argument representations for verbs exists, it should generalize to other such mappings, e.g., spatial relations. In

this context, and in order to validate the attentional strategy for extracting spatial relations described above, we collected data from 4 human subjects who were instructed to "teach" the robot by demonstrating and narrating spatial relations with the four colored blocks. The resulting data were 74 training examples, each consisting of a short video sequence in which the subject "showed" or demonstrated a spatial relation, and provided the corresponding sentence description (e.g., "The block is below the triangle") of the demonstrated relation. The spatial attention mechanism determined the most relevant spatial relation for the video sequence in each case in order to extract the "meaning" in terms of a spatial relation. Of the resulting 74 meanings, 67 (90%) corresponded to the meaning described by the subject, i.e., to the intended meaning.

Fig. 6 illustrates the robustness of the two underlying assumptions with respect to human performance. In Fig. 6A we see that the human subjects reliably demonstrated relations in a pertinent manner, adhering closely to the four principal axes. Likewise, Fig. 6B illustrates that in the large majority of the examples, subjects placed the target object closer to the referent object than to the other objects in the scene. This demonstrates that perceptual primitives of motion, distance and angle can be reliably used in order to construct a higher-level attention capability.

The 74 resulting ⟨sentence, relation-meaning⟩ pairs were then used as input to the grammatical construction learning model. After 5 exposures to the data set, the model converges to a stable performance. Of the 74 input ⟨sentence, meaning⟩ pairs, 67 are well-formed, and 7 are not well-formed, i.e., the extracted relation does not correspond to the described meaning. After training, the model correctly identifies the 7 non-well-formed ⟨sentence, meaning⟩ pairs (i.e., it detects that the relation described in the sentence does not correspond to the actual relation), and performs at 98% correct (66/67) for the remaining correct pairs. This demonstrates the robustness of learning with real data. We also verified

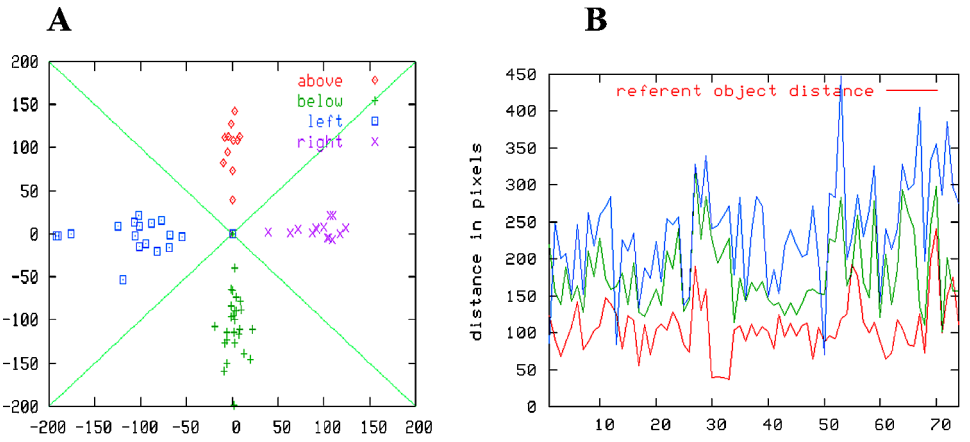

Fig. 6. A. Location of the target with respect to referent object in the Relation(target, referent) relations. Note that the experimental subjects place the target object closely aligned with appropriate direction (left, right, above, below), and not ambiguously, as hypothesized. B. Distance between target and other objects. Lowest curve is for the intended referent, extracted from the verbal descriptions. As predicted, subjects almost invariably place the target closest to the intended referent, as hypothesized.

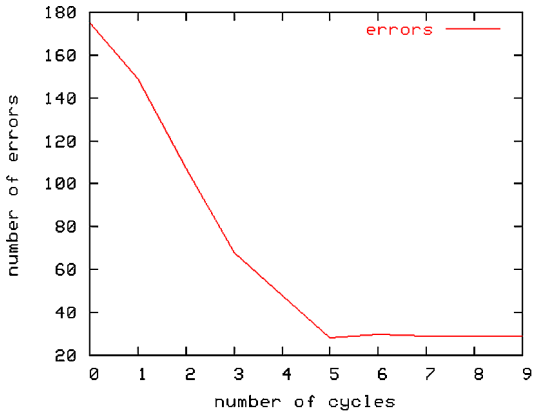

Fig. 7. Learning performance of grammatical construction learning model with the 74 relational training examples. Each cycle corresponds to a full pass through the 74 ⟨sentence, relation⟩ pairs. Final errors are due to incorrect ⟨sentence, meaning⟩ data in the input.

that based on training with correct examples, the model could generalize this knowledge to a new ⟨sentence, relation-meaning⟩ generalization data set, obtained by generating hand-coded ⟨sentence, relation-meaning⟩ pairs outside of the training set. This demonstrates the extension of the construction grammar framework to encompass spatial relations. Gorniak and Roy [25] have demonstrated a system capable of learning to understand much more complex expression of spatial relations, using data obtained from subject's in an object specification task (see also [35]). It will be of interest to determine to what extent these expressions can be captured in the construction framework.

6. Experimental results in unconstrained conditions

The results reviewed in Sections 4 and 5 indicate that when well-formed ⟨sentence, meaning⟩ pairs are provided to the model for learning under well-controlled experimental conditions, the grammatical construction model is quite robust in its ability to extract the underlying structural relations between sentences and meanings, and that this mechanism extends to meanings in the form of spatial relations. The current experiment attempts to determine if the system can exhibit similar performance under less constrained conditions.

6.1. The training and testing data

To generate data for training and testing the model, four naive English speaking subjects were asked to enact and simultaneously narrate 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. The subjects were told that they should demonstrate and narrate their actions in order to teach the system the meaning of "touch, push, take and give". Each of the subjects demonstrated and narrated the different actions for 15 minutes, at a leisurely pace.

As before, a video camera above the surface provided a video image that was 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 that was subsequently processed for event analysis (as described above). The simultaneous narration of the ongoing events was recorded for subsequent transcription. Speech and vision data were thus acquired and then processed off-line yielding a data set of matched ⟨sentence, meaning⟩ pairs that were provided as input to the structure mapping model. Time-stamps in the video and auditory data streams allowed off-line realignment of the ⟨sentence, meaning⟩ data. From the auditory streams from the four subjects, a total of 289 sentences were transcribed. During the ⟨sentence, meaning⟩ pairing based on the time-stamp alignment, 7 sentences were isolated for which the corresponding event perception failed to generate a meaning, and so these sentences were removed from the corpus. The resulting data set from the four subjects consisted of 282 ⟨sentence, meaning⟩ pairs.

6.2. Evaluation

These sentences were divided into two data sets, one to be used for training (subjects S1 and S2) and the second used for testing (subjects S3 and S4) using the fixed parameters established during training. The training set consisted of 135 ⟨sentence, meaning⟩ pairs. Each sentence describes an event and the agent, object and potential recipient of the event, and thus for the 135 sentences there were 451 corresponding scene referent elements. After exposure to the training set, the grammatical construction model yielded a stable performance with a total error rate of 24% for subject S1 and 36% for subject S2 (Table 2), yielding a total error rate for the training set of 29%. Post-hoc analysis revealed two principal classes for the sources of the errors.

In a number of cases (accounting for 14% of the errors) we determined that the event or verb (i.e., touch, push, take or give) as determined by the vision system was incorrectly identified (with respect to the verb in the paired sentence), while the agent, object and recipients were correctly identified. A detailed analysis revealed that in these *verb-error* cases, the subjects' event demonstrations produced contact sequences with values that were not compatible with the physical parameters for distinguishing between touch, push and take that were used in the scene analysis system. The parameters used in the scene analysis system were pre-determined such that they provided the most robust performance on a "corpus" of event scenes that was collected prior to the current experiments. The failure of the system in the current *verb-errors* was thus due to variability in the spatio-temporal

Table 2
Error analysis for individual subjects

Subject	Sentences	Scene elements	Total errors	Verb errors	Expression and perceptual errors
S1 (Train)	81	263	64 (24%)	53 (20%)	11 (4%)
S2 (Train)	54	188	67 (36%)	11 (6%)	56 (30%)
S3 (Test)	88	271	55 (20%)	29 (11%)	26 (9%)
S4 (Test)	59	196	55 (28%)	16 (8%)	39 (20%)

aspects of the movements in the corpus used to determine these parameters, and the movements performed by the subjects in the current experiments.

A typical resulting error was that an event that the subject described as a *touch* was misrecognized as a *push*, while the agent and object were correctly identified. The remaining errors, accounting for 15% of the total errors were of mixed sources that included the speech and transcription errors, and other errors due to the perceptual system. The vast majority ($\geqslant 90\%$) of the descriptive strategies employed in the training set corresponded to use of the active voice transitive and ditransitive constructions, exemplified by "The block pushed the triangle" and "The moon gave the triangle to the block", respectively.

Once the learning system had thus been exposed to the training set, we disabled the learning mechanism, and then exposed the system to the test set. Under these generalization testing conditions, the model performed with a total of error rate of 24%, with 9% of the errors attributed to the verb error phenomena described above. This indicates that the learning achieved during exposure to the training set transferred with reasonable preservation of accuracy to the generalization test set.

It is of interest that the problems with verbs vs. nouns has also been observed in terms of an advantage for learning nouns first, over verbs, in English. Gillette et al. [23] discuss competing theories on why this might be the case, and the current results suggest that it may be related to the perceptual ambiguity of verbs vs. nouns. That is, perceptually, while it may be obvious who the agent and the object are, the correct choice of a verb to describe their interaction may be less obvious. Likewise it is likely that there will be variability between individuals concerning how they associate different words with different meanings (Reiter, this volume), and that this variability may be greater for event verbs than for concrete objects.

7. Spoken language interaction

These initial learning results for sentence understanding indicate that in this constrained environment of blocks on a table, the construction grammar based model is adequate for capturing the relations between language and the world, but of course an equally important test of utility is using this learned language capability in an interactive human-robot communication scenario. Technically there are several issues to be addressed, including (a) use of the learned grammatical constructions to generate sentences from visually perceived scenes, and to do so in a manner that is appropriate from a pragmatic discourse perspective; and (b) inserting this capability into an interactive environment coupled with speech synthesis and recognition.

7.1. Generating sentences from events

Each grammatical construction in the construction inventory corresponds to a mapping from sentence to meaning. This information can thus be used to perform the inverse transformation from meaning to sentence. For the initial sentence generation studies we concentrated on the 5 grammatical constructions illustrated in Table 4. These correspond to constructions with one verb and two or three arguments in which each of the different ar-

Table 3
Overall accuracy results for training and testing

⟨Sentence, meaning⟩ set	Accuracy training (S1 & S2)	Accuracy testing (S3 & S4)
All	71% (29% error)	76% (24% error)
All except Verb Errors	85% (15% error)	85% (15% error)

Table 4
Sentence and corresponding constructions for robot language generation

Sentence	Construction ⟨sentence, meaning⟩
1. The triangle pushed the moon.	⟨*Agent event object*, *event*(*agent*, *object*⟩.
2. The moon was pushed by the triangle.	⟨*Object* was *event* by *agent*, *event*(*agent*, *object*⟩
3. The block gave the moon to the triangle.	⟨*Agent event object* to *recipient*, *event*(*agent*, *object*, *recipient*)⟩
4. The moon was given to the triangle by the block.	⟨*Object* was *event* to *recipient* by *agent*, *event*(*agent*, *object*, *recipient*)⟩
5. The triangle was given the moon by the block.	⟨*Recipient* was *event object* by *agent*, *event*(*agent*, *object*, *recipient*)⟩

guments can take the focus position at the head of the sentence. The left columns of Table 3 illustrates example sentences, and on the right, the corresponding generic construction. In the representation of the construction, the element that will be at the pragmatic focus (i.e., at the beginning or head of the sentence) is underlined. This focus information will be of use in selecting the correct construction to use under different discourse requirements during question answering.

This construction set provides sufficient linguistic flexibility, so that for example when the system is interrogated about the block, the moon or the triangle after describing the event *give*(*block*, *moon*, *triangle*), the system can respond appropriately with constructions of type 3, 4 or 5, respectively allowing block, moon and triangle to take the head position of the sentence. The important point is that each of these different constructions places the pragmatic focus on a different argument by placing it at the head of the sentence.

The algorithm for selection of the construction type for sentence production takes as input a meaning coded in the form event(arg1, arg2, arg3), and an optional focus item (specification of one of the three arguments). Based on this input, the system will deterministically choose the appropriate two- or three-argument construction, with the appropriate focus structure, in a pragmatically relevant manner. Thus, in the dialog example below, the human user generates an event corresponding to *gave*(*block*, *cylinder*, *moon*) and then asks what happened to the moon (the recipient in this case). Based on these inputs, the system selects the three-argument construction in which the recipient is the focus element (Construction 5). The predicate and arguments from the meaning are inserted into their appropriate positions, and the system thus responds: *The moon was gave the cylinder by the block.*

7.2. Real-time interactive environment

The next task at hand is to integrate these pieces, including (a) scene processing for event recognition, (b) sentence generation from scene description and response to questions, (c) speech recognition for posing questions, and (d) speech synthesis for responding—into an interactive environment. The CSLU Speech Tools Rapid application Development (RAD) (http://cslu.cse.ogi.edu/toolkit/index.html) provides useful capability in this context. The system provides dialog management with a flexible and powerful graphical user interface, coupled with the global ability to link speech recognition and synthesis to the conditional execution of code on the same machine or on remote machines via ftp and socket protocols. This results in a hub architecture with RAD at the hub and the vision processing, language model, speech-to-text and voice synthesis at the periphery.

7.3. Example dialog

Here we provide an example of typical dialog. Speech is indicated in *italics*.

Robot: *Welcome to the talking robot demonstration. Show me an object.*
Human: (Picks up and moves the block)
Robot: *That is the block. Another object?*
Human: *Yes* (moves the moon)
Robot: *That is the moon. Another object?*
Human: *No.*
Robot: *Shall we stop or go on?*
Human: *Go on.*
Robot: *When you are ready, make a movement with the objects so that I can see what you are doing. I will then describe what you did.*
Human: (picks up the block and uses it to move the cylinder to the moon).
Robot: *The block gave the cylinder to the moon. Was I correct?*
Human: *Yes.*
Robot: *Good for me.* (stores results in database) *Do you have any questions?*
Human: *Yes.*
Robot: *Would you like to know about the block, the moon or the cylinder?*
Human: *The moon.*
Robot: *The moon was gave the cylinder by the block. Do you have any questions?*
Human: *Yes.*
Robot: *Would you like to know about the block, the moon or the cylinder?*
Human: *The cylinder.*
Robot: *The cylinder was gave to the moon by the block.*

The sample dialog illustrates how vision and speech processing are combined in an interactive manner. Two points are of particular interest. In the response to questions, the system uses the focus element in order to determine which construction to use in the response. This illustrates the utility of the different grammatical constructions. However, we note that the two passivized sentences have a grammatical error, as “gave” is used, rather than “given”. This is due to the indiscriminant binding of the words gives, gave, and given to the meaning *gave*. In production this morphological information is lost and cannot be

expressed. Such expression errors in the correct use of grammatical morphology are common in language acquisition [8]. Correcting such errors requires that the different tenses are correctly associated with the different construction types, and will be addressed in future research.

7.4. Evaluation

Once the interaction system was stable, we wanted to evaluate its performance with naïve users who were first shown a few trials with the system and then simply told to try to perform different actions with the system in order to see how it could perceive and describe those actions. We thus invited users to interact with the system and recorded their evaluation of the system's performance. That is, as illustrated in the dialog above, once the system describes the event, it asks the user whether the response was correct or not and encodes this information. Three response types are possible: Correct, Technical Error—corresponding to cases where the inter-process communication failed and there was no event described, and Description Error—corresponding to cases in which the description was incorrect. From a total of 241 interactions that were thus recorded, 173/241 of the trials (72%) were correct, 19/241 trials (8%) were incorrect due to Technical Errors and 50/241 trials (20%) were incorrect due to Description Errors. If the technical communication errors are eliminated, this yields an accuracy rate of 173/222 trials (77%) correct. This indicates that while the event perception and sentence generation capabilities can be improved upon, the system yields reasonable performance in actual-use conditions with naïve subjects.

8. Discussion

From the context of "connecting language to the world" this research has attempted to exploit knowledge of how infants extract meaning from the world, and how they learn the mappings between language and these meanings in order to become competent in language. In doing so, we chose to side with the "functionalist" or "usage based" school of cognitive development and language acquisition (e.g., [47]), vs. the "nativist" school that attributes a much greater importance to a highly pre-specified language-specific universal grammar capability (e.g., [5]). This choice was based on the progressive developmental trajectory that is proposed in the former and not the latter. This trajectory provides a useful set of technical milestones in the development of the system, with interesting results from the outset.

In this context, the learning results obtained with unbiased input from four naive subjects in the event description teaching task in Section 6 indicates that the grammatical construction model could adequately map sentences to meanings. Despite the fact the subjects were not biased, one could still argue that the task itself was biased as there are just so many ways that one can describe these actions. However, in response to this comment we can respond that we have clearly demonstrated that the system can accommodate a larger variety of different construction types (over 35), as well as constructions in typologically distinct languages including Japanese [15]. Indeed, as long as different construction types are identifiable by their closed-class signature (to date we have no exceptions), then the

model can accommodate them. The use of such constructions in the event description task of Section 7 illustrates the utility of this approach in human-machine interaction.

With respect to this type of mapping from sentence to meaning guided by regularities in the structure of closed-class elements and word order Chang [6] has taken a similar approach, in which a recurrent network serves essentially the same role as the ConstructionIndex, and guides the activation of variable bindings for meaning assignment in message-sentence pairs, analogous to the FormToMeaning mapping. The model displayed generalization allowing words to occur in novel locations, within learned sentence types, and explained a variety of data on normal and aphasic sentence production, though the constructions employed were relatively simple (e.g., no relativised sentences). In the domain of more complex constructions, Miikkulainen [30] demonstrated a system that learned the mapping between relative phrase constructions and multiple event representations. The architecture is based on a parser for extracting case role assignments, a stack for storing ongoing representations during recursion, a segmenter for segmenting the input into clauses, and control/synchronization of these components (e.g., for pushing and popping from the stack, etc). The system demonstrates impressive capabilities to generalize to new relativised constructions, at the expense of a significant quantity of processing capabilities specific for this task.

With respect to related research connecting language to the world, the current work can be situated in the context of similar systems developed by Roy [36] and Siskind [41]. Roy's DESCRIBER [36] learns scene semantics, words and sentence forms from ⟨scene, description⟩ input in order to describe spatial relations, including the use of relative phrases. DESCRIBER extracts probabilistic structures encoding regularities in the spatial scene and in the word category and sequential structure of the describing sentences. A planning algorithm then integrates the extracted semantic syntactic and contextual constraints to generate syntactically well-formed descriptions of novel scenes. Our approach differs from Roy's in three principal regards: (1) From the meaning perspective we concentrate on describing events in dynamic scenes rather than spatial relations in static scenes. In this context, Roy has taken on the problem of contextual ambiguity resolution via language in a more robust manner that we have. In order to unambiguously describe an object in a cluttered spatial array, DESCRIBER must use contextual information in order to generate the most relevant and unambiguous response. The overall ability of the system, and its particular ability to use these contextual constraints for disambiguation reveals the remarkable power of the extraction and reuse of probabilistic structure inherent in the input. (2) From the language model perspective, we employ a construction grammar based model, in which constructions are templates, as opposed to the statistical bigram based model employed by Roy [36]. (3) In the context of human-machine interaction, the human subject can interrogate our system about a specific object's involvement in an event. Depending on whether that object was the agent, the patient or the recipient of the action, the system will choose the appropriate grammatical construction so that the object term is at the head of the sentence. Thus, if the event is that "the block pushed the triangle", and the subject asks "What happened to the triangle?" the system will respond "The triangle was pushed by the block". An additional model related difference is that our construction grammar model was initially developed for comprehension, i.e., learning the mapping from sentence to meaning. In the current work we "inverted" these same grammatical constructions to allow mapping from

meaning to sentence in description tasks. In its current state, DESCRIBER describes but does not understand, though the same probabilistic structure mapping approach is clearly feasible for comprehension.

In the domain of event recognition, Siskind's system, LEONARD, extracts the perceptual primitives GROUNDED, RIGID, REVOLUTE, SAMELAYER, and TOUCHES in order to build up primitive force dynamic events including SUPPORTED, RIGIDATTACHMENT, SUPPORTS, CONTACTS and ATTACHED. This already involves some complexity in processing, as the determination that an object x supports an object y requires counterfactual reasoning. Using these force dynamic events, LEONARD can then recognize compound events PICKUP, PUTDOWN, STACK, UNSTACK, MOVE, ASSEMBLE, DISASSEMBLE based on temporal event logic formulation of these events in terms of the force dynamic events. Fern et al. [20] extended this work to demonstrate that the system could learn the event logic characterization of these different compound events.

The principal difference between our approach and that of Siskind is that our system is extremely simplified and minimalist, using only two primitives, i.e., physical contact, and motion that can be extracted with a minimum of computation, yet can still provide the basis for events including TOUCH, PUSH, TAKE, GIVE, TAKE-FROM. Thus, using the primitives of movement and contact that can be directly extracted from the image, we pass directly to "compound event" recognition, bypassing the intermediate force dynamic event stage. The advantage is the significant complexity reduction, and demonstration that even in this minimalist configuration the system can recognize events. The price is that the richness is reduced with respect to that of LEONARD. This recalls the idea of relevance as maximizing useful information while minimizing processing effort.

Our future objectives include the exporting of this system to a robot platform that allows human-robot interaction not only about scene analysis but about action as well. This will provide the scenario in which language can be used to command and instruct the robot. Human based robot instruction has often relied on imitation, but clearly the use of verbal coaching and explaining will also provide a powerful information transfer mechanism. The current system has two important features that should make it of interest to the robot community. First, it is adaptable in that the system will learn the language structures adapted to a given interaction context. Second, the system has a very flexible semantics in the form of predicate—argument relations. We have demonstrated that this is highly appropriate for event and spatial relation descriptions, but it will also be highly suitable for the syntax of robot commands, and should thus be of immediate practical value within the community.

Perhaps the most important remaining question to be answered concerns the extent to which these results generalize and scale. This brings us back to the principal assumptions and limitations—that grammatical constructions are identified in terms of configurations of function words at the sentence level. Thus, individual roles of function words in linking aspects of constituent structure to meaning structure currently cannot be exploited to render the system generative. Likewise the system lacks the ability to recognize noun phrases and verb phrases as constituents that can fill in the roles currently taken exclusively by single open-class words. This would allow the system to accommodate a sentence such as "John was taught to run by Jim" by processing "taught to run" as a verb phrase, relying on knowledge about differences between "to NOUN" and "to VERB". Indeed, it is pre-

cisely this type of processing that defines the next step in our research program for the evolution of the system. However, despite this limitation, the system has demonstrated that indeed, this method for managing constructions via the ConstructionIndex is robust. For all of the grammatical constructions we have presented, the configuration of function words does reliably distinguish between constructions. In this context, one must be careful about examples such as "John saw the tree on the hill" and the potential semantic ambiguity. Neither a generative nor a construction-based approach can resolve such ambiguity (i.e., determine which of multiple constructions is being employed) without taking extra-phrasal pragmatic factors into account. Such factors can be taken into account, but the current research does not address this type of problem. Still, this limited capacity allows a good deal of productive generalization: It can generalize both for nominal arguments, as well as handling different verbs, and verbs can be used with novel argument structures. This is because the coding of lexical semantics is dissociated from the coding of phrasal semantics. For example, phrasal semantics encodes the fact that for an active transitive sentence, the second open-class element in the sentence corresponds to the *event* or *action* element of the meaning. Once a verb meaning has been learned for a given argument structure it can then generalize to any learned construction and corresponding argument structure. Thus, if the verb push is only used in input training sentences with active or passive transitive forms, when the system encounters the input sentence "The block pushed the moon to the cylinder" it correctly generates the meaning push(block, moon, cylinder). Likewise, we can ask if the system sees the sentence 'Mary was given the ball by John' in training, will it be able to understand the sentence 'Mary was given the ball'? The answer is yes, as long as it has also seen constructions of the form

$$\langle \textit{Recipient} \text{ was } \textit{event object}; \textit{event}(\_\_, \textit{object}, \textit{recipient}) \rangle,$$

in which the agent is not specified. The point here is that these semantically related mapping patterns are learned completely independently by the system. Interestingly this type of phenomenon in which a verb can be used with one configuration of arguments but not another (e.g., intransitive but not transitive) is a commonly observed in language development [47].

In this context, we must distinguish limitations of the event processing system from limitations in the language model. Clearly, the event processing system we have developed here, which has a fixed set of possible meaning predicates, is limited. Again, however, the proposal is that given a more robust meaning extraction and representation capability, the sentence-to-meaning mapping model will continue to operate. This has been already partially validated by the use of more complex constructions (illustrated in Table 1) that are beyond the scope of our visual event processing system, yet are perfectly accommodated by the language model. Thus, while the event processing system will not generate "push" in a three-argument predicate, the language system can nonetheless understand the use of push in a sentence such as "The block pushed the triangle to the moon". More generally, once a construction is learned, it can then be used with any verb, and thus potentially incorrectly in some cases. This is a property of the total separation of lexical and phrasal semantics in this model. These incorrect uses will be constrained by the meanings provided by the event processing system.

With respect to scaling for the lexicon, in all of the experiments the vocabulary size was limited to 25 lexical items by the size of the Open-Class and Closed-Class word vectors. This is "ridiculously" small, but as stated above our principal objective was not to develop a model of lexical acquisition, but a model of sentence to meaning mapping. Thus, the use of a small lexicon is warranted in this context. The use of a distributed code within these 25 element vectors would allow a much greater vocabulary size, and the fact that the lexical and phrasal semantic systems are separated means that we could even insert a more powerful lexical semantic system into the current system. Finally, with respect to scaling to more complex grammatical structure, we demonstrated that once the meaning representation can accommodate multiple (in this case two) event predicate-argument descriptions, the model can directly accommodate a variety of conjoined and relative phrase constructions as illustrated in Table 4.

In conclusion, these results demonstrate that within the domain of physical contact events (including touch, push, take and give) and simple spatial relations (including above, below, left and right) all of which can be expressed as predicate-argument structures, the mappings between these structured meanings and sentences that can express them can be captured in grammatical constructions that can subsequently be re-used in generalizing to new sentences of the same structural type. Indeed, part of the central propositions of construction grammar is that there is a direct structural relation between basic grammatical constructions and the semantic structures representing scenes that are basic to human experience [24]. Clearly, the current incorporation of this concept in a sort of template-based approach to language processing has limitations (e.g., it would not fare well with this or the previous sentence). However, particularly within domains of structured interactions, a good proportion of human speech can be accommodated by this type of "routinized" approach [32]. And perhaps of equal importance, this template-based approach is a clearly defined phase in the human developmental trajectory of language acquisition [47]. Our future research will begin to address how to move on from here, including specification of a more compositional generalization capability.

Acknowledgements

This work has been supported in part by the French ACI NIC and ACI TTT, the HFSP, the European OMLL and ECRPSS projects and LAFMI. We gratefully acknowledge Nicolas Dermine who worked on the RAD integration for an earlier version of the system as part of an internship for the Lyon Ecole Centrale.

References

[1] E. Bates, S. McNew, B. MacWhinney, A. Devescovi, S. Smith, Functional constraints on sentence processing: A cross linguistic study, Cognition 11 (1982) 245–299.

[2] J.M. Blanc, C. Dodane, P.F. Dominey, Temporal processing for syntax acquisition: A simulation study, in: Proceedings of the 25th Ann Conf. Cog. Sci. Soc., MIT Press, Cambridge, MA, 2003.

[3] C.M. Brown, P. Hagoort, M. ter Keurs, Electrophysiological signatures of visual lexical processing: Open- and closed-class words, J. Cognitive Neurosci. 11 (3) (1999) 261–281.

[4] D. Caplan, C. Baker, F. Dehaut, Syntactic determinants of sentence comprehension in aphasia, Cognition 21 (1985) 117–175.
[5] N. Chomsky, The Minimalist Program, MIT Press, Cambridge, MA, 1995.
[6] F. Chang, Symbolically speaking: A connectionist model of sentence production, Cognitive Sci. 93 (2002) 1–43.
[7] N.C. Chang, T.V. Maia, Grounded learning of grammatical constructions, in: Proc. AAAI Spring Symp. on Learning Grounded Representations, Stanford, CA, 2001.
[8] E. Clark, First Language Acquisition, Cambridge University Press, Cambridge, 2003.
[9] G.W. Cottrel, B. Bartell, C. Haupt, Grounding meaning in perception, in: Proc. GWAI90,14th German Workshop on Artificial Intelligence, Springer, Berlin, 1990, pp. 307–321, .
[10] P.F. Dominey, Conceptual grounding in simulation studies of language acquisition, Evolution of Communication 4 (1) (2000) 57–85.
[11] P.F. Dominey, Learning grammatical constructions in a miniature language from narrated video events, in: Proceedings of the 25th Annual Meeting of the Cognitive Science Society, Boston, 2003.
[12] P.F. Dominey, Learning grammatical constructions from narrated video events for Human-Robot interaction, in: Proc. IEEE Conf. on Humanoid Robotics, Karlsruhe, 2003.
[13] P.F. Dominey, Boucher, Developmental stages of perception and language acquisition in a perceptually grounded robot, Cognitive Systems Res. (2005), in press.
[14] P.F. Dominey, M. Hoen, T. Lelekov, J.M. Blanc, Neurological basis of language in sequential cognition: Evidence from simulation, aphasia and ERP studies, Brain and Language 86 (2) (2003) 207–225.
[15] P.F. Dominey, T. Inui, A developmental model of syntax acquisition in the construction grammar framework with cross-linguistic validation in English and Japanese, in: Proceedings of the CoLing Workshop on Psycho-Computational Models of Language Acquisition, Geneva, 2004, pp. 33–40.
[16] P.F. Dominey, F. Ramus, Neural network processing of natural language: I. Sensitivity to serial, temporal and abstract structure of language in the infant, Lang. Cognitive Processes 15 (1) (2000) 87–127.
[17] J. Elman, Finding structure in time, Cognitive Sci. 14 (1990) 179–211.
[18] J.A. Feldman, G. Lakoff, A. Stolcke, S.H. Weber, Miniature language acquisition: A touchstone for cognitive science, in: Proceedings of the 12th Ann Conf. Cog. Sci. Soc., MIT Press, Cambridge, MA, 1990, pp. 686–693.
[19] J. Feldman, G. Lakoff, D. Bailey, S. Narayanan, T. Regier, A. Stolcke, L0: The first five years, Artificial Intelligence Rev. 10 (1996) 103–129.
[20] A. Fern, R. Givan, J.M. Siskind, Specific-to-general learning for temporal events with application to learning event definitions from video, J. Artificial Intelligence Res. 17 (2002) 379–449.
[21] F. Ferreira, The misinterpretation of noncanonical sentences, Cognitive Psychol. 47 (2003) 164–203.
[22] C. Fisher, Structural limits on verb mapping: The role of analogy in children's interpretation of sentences, Cognitive Psychol. 31 (1996) 41–81.
[23] J. Gillette, H. Gleitman, L. Gleitman, A. Lederer, Human simulation of vocabulary learning, Cognition 73 (1999) 135–151.
[24] A. Goldberg, Constructions: A Construction Grammar Approach to Argument Structure, Univ. Chicago Press, Chicago, 1995.
[25] P. Gorniak, D. Roy, Grounded semantic composition for visual scenes, J. Artificial Intelligence Res. 21 (2004) 429–470.
[26] P.J. Kellman, H. Gleitman, E.S. Spelke, Object and observer motion in the perception of objects by infants, J. Experimental Psychol.—Human Perception and Performance 13 (4) (1987) 586–593.
[27] L. Kotovsky, R. Baillargeon, The development of calibration-based reasoning about collision events in young infants, Cognition 67 (1998) 311–351.
[28] R. Langacker, Foundations of Cognitive Grammar. Practical Applications, vol. 2, Stanford University Press, Stanford, CA, 1991.
[29] J. Mandler, Preverbal representations and language, in: P. Bloom, M.A. Peterson, L. Nadel, M.F. Garrett (Eds.), Language and Space, MIT Press, Cambridge, MA, 1999, pp. 365–384.
[30] R. Miikkulainen, Subsymbolic case-role analysis of sentences with embedded clauses, Cognitive Sci. 20 (1996) 47–73.
[31] J.L. Morgan, R. Shi, P. Allopenna, Perceptual bases of rudimentary grammatical categories, in: J.L. Morgan, K. Demuth (Eds.), Signal to Syntax, Lawrence Erlbaum, Mahwah, NJ, 1996, pp. 263–286.

[32] M.J. Pickering, S. Garrod, Toward a mechanistic psychology of dialogue, Behav. Brain. Sci. 27 (2) (2004) 169–190; discussion 190–226.
[33] P.C. Quinn, Concepts are not just for objects: Categorization of spatial relation information by infants, in: D.H. Rakison, L.M. Oakes (Eds.), Early Category and Concept Development: Making Sense of the Blooming, Buzzing Confusion, Oxford University Press, Oxford, 2003, pp. 50–76.
[34] P.C. Quinn, J.L. Polly, M.J. Furer, V. Dobson, D.B. Nanter, Young infants' performance in the object-variation version of the above-below categorization task, Infancy 3 (2002) 323–347.
[35] T. Regier, The Human Semantic Potential, MIT Press, Cambridge, MA, 1996.
[36] D. Roy, Learning visually grounded words and syntax for a scene description task, Computer Speech and Language 16 (2002) 353–385.
[37] D. Roy, A. Pentland, Learning words from sights and sounds: A computational model, Cognitive Sci. 26 (1) (2002) 113–146.
[38] A.J. Sanford, P. Sturt, Depth of processing in language comprehension: Not noticing the evidence, Trends Cognitive Sci. 6 (9) (2002) 382–386.
[39] R. Shi, J.F. Werker, J.L. Morgan, Newborn infants' sensitivity to perceptual cues to lexical and grammatical words, Cognition 72 (2) (1999) B11–B21.
[40] J.M. Siskind, A computational study of cross-situational techniques for learning word-to-meaning mappings, Cognition 61 (1996) 39–91.
[41] J.M. Siskind, Grounding the lexical semantics of verbs in visual perception using force dynamics and event logic, J. Artificial Intelligence Res. 15 (2001) 31–90.
[42] L. Steels, Language games for autonomous robots, IEEE Intelligent Syst. 16 (5) (2001) 16–22.
[43] L. Steels, J.C. Baillie, Shared grounding of event descriptions by autonomous robots, Robotics and Autonomous Systems 43 (2–3) (2003) 163–173.
[44] A. Stolcke, S.M. Omohundro, Inducing probabilistic grammars by Bayesian model merging, in: Grammatical Inference and Applications: Proc. 2nd Intl. Colloq. on Grammatical Inference, Springer, Berlin, 1994.
[45] L. Talmy, Force dynamics in language and cognition, Cognitive Sci. 10 (2) (1988) 117–149.
[46] M. Tomasello, The item-based nature of children's early syntactic development, Trends Cognitive Sci. 4 (4) (1999) 156–163.
[47] M. Tomasello, Constructing a Language: A Usage-Based Theory of Language Acquisition, Harvard University Press, Cambridge, 2003.