Emergence of grammatical constructions: evidence from simulation and grounded agent experiments

PETER FORD DOMINEY*
Institut des Sciences Cognitives, CNRS UMR 5015, 67, Boulevard Pinel, 69675 BRON Cedex, France

This research takes grammatical constructions (sentence form-to-meaning mappings) as an alternative to abstract generative grammars in the context of understanding the emergence of language. A model of sentence processing based on this construction grammar approach is presented, and then a series of neuropsychological and neurophysiological studies are reviewed that attempt to validate the model and to establish its neurophysiological underpinnings. The resulting model is demonstrated to provide insight into a developmental and evolutionary passage from unitary idiom-like holophrases to progressively more abstract grammatical constructions. The model is then functionally validated by its insertion into a perceptually grounded system that allows spoken language interaction with a human interlocutor. The potential utility of this emergence approach in understanding language is discussed.

Keywords: Neural network; Language; Sensorimotor sequence; Grammatical construction

1. Introduction

Clearly, the emergence of human language at the behavioural level is one of the most phenomenal events in biology, and characterizing the underlying neurophysiological basis for language and its evolution and development remains one of the principal open challenges in cognitive neuroscience; because of its uniquely human and apparently discontinuous sudden appearance in evolution, understanding language and its neurophysiology is all the more mysterious. In this context, Chomsky (1959, 1965) took a theoretical stance that has had a highly significant impact on language and linguistics research for the last 50 years. He held that the child’s task of inducing the target language grammar based on the very limited quantity of available input data rendered the acquisition task highly under-specified: the input data could correspond to many possible grammars—How does the child determine the correct one? The proposed solution to this ‘poverty of the stimulus’ problem was an innate universal grammar (UG), such that limited input data allowed the infant to adapt (or set the parameters) of the UG in order to tailor it to the target language. This line of reasoning, and the argument against language being acquired by general learning mechanisms, was strengthened by two related theoretical positions. In 1967, Gold published a paper on ‘Learnability in the limit’, in which he provided formal proof that under rigorous constraints of learning success, language

*Email: dominey@isc.cnrs.fr
cannot be learned with a general learning system by positive evidence alone, and thus requires some alternative method to restrict the learning problem, with UG providing a suitable alternative. Similarly, the argument that UG is innate and is not developed implied the ‘continuity hypothesis’, which holds that via UG children have access to an adult-like syntactic system that they bring to the problem of language acquisition (Pinker 1984, and see discussion in Tomasello 2000). What this would predict is that once children have some aspect of the grammar worked out (i.e. a given parameter is set), then this aspect should be fully available in order to generalize to new cases. From this perspective, then, language and its evolution and development is a truly formidable problem, as it is quite ‘discontinuous’—with at least two large gaps: the behavioural gap, from no language to UG and access to an adult grammar; and the neurophysiological gap, from pre-linguistic to linguistic beings.

This paper will attempt to bridge these two gaps: first by considering language acquisition from the usage-based perspective as proposed by Tomasello (2003), in which the successive progression of capabilities leads to a much more learning-based approach; and second, by situating the neurophysiological language organ within the known primate neurophysiology of sensorimotor sequence processing. This emergence perspective on language will be supported by data from neurophysiologically guided simulation studies and experiments with a perceptually grounded robot system for human–robot interaction.

Section 2 will introduce a functional model of language processing in the context of construction grammar (Clark 2003, Goldberg 1995), along with a demonstration of the language processing capabilities and limitations of this model. This will be followed in section 3 by a characterization of the neurophysiological correlates of the functional model. In section 4, the model will be used to provide insight into the progressive ‘usage-based’ (Tomasello 2003) development of increasingly abstract grammatical constructions, both from a developmental perspective and from an evolutionary perspective. Section 5 then demonstrates the feasibility of this system in a grounded agent for human–robot interaction.

2. Functional characterization of the model

From a functional perspective, given a set of <sentence, meaning> pairs, a language acquisition system should learn the mappings between sentences and meanings in a manner that can generalize to new sentences. The current approach (from Dominey and Boucher 2005a) is based on a characterization of language as a structured inventory of grammatical constructions, each of which defines this sentence—meaning mapping for a class of sentences corresponding to that construction. Thus, the sentence ‘John gave the ball to Mary’ can be considered in terms of the dative construction:

‘Agent action object to recipient’; <ACTION(AGENT, OBJECT, RECIPIENT)>

in which the left-hand component corresponds to the sentence with italicized words corresponding to lexical categories that can be replaced by specific noun-phrases and verb-phrases, and the right-hand side corresponds to the meaning, represented in a predicate–argument form, with upper case words corresponding to conceptual representations. In this context, the problem to be solved by the language system involves learning how to map from the structure of the sentence on to the structure of the meaning representation. This corresponds to the problem of thematic role assignment, or determining ‘who did what to whom’. In this context, cross-linguistic studies have revealed that open class words (e.g. nouns, verbs, adjectives and adverbs) are assigned to their thematic roles based on word order and/or the pattern of closed class words (grammatical function words or morphemes including prepositions, determiners)
in the sentence (Bates et al. 1982). Thus, ‘John gave the ball to Mary’ and ‘The ball was given to Mary by John’ will map in a different manner (with respect to the order of the words in the sentences) on to the same meaning, and this mapping will be in part guided by the grammatical function words ‘was, to, by’.

The actual mapping of sentence form on to meaning for sentence comprehension takes place at two distinct levels: words are associated with individual components of event descriptions, and grammatical structure is associated with functional roles within scene events. The first level of lexical semantics has been addressed by Siskind (1996), Roy and Pentland (2002) and Steels and Kaplan (2001), and it is treated here in a relatively simple but effective manner. Our principle interest lies more in the second level of phrasal semantics, or mapping between sentence and meaning structure, and the ability to handle a large variety of different mappings, or grammatical constructions. Figure 1(A, B) illustrates how two different grammatical constructions are processed by the model. The passive construction ‘object was verb to recipient by agent’ in A and the active construction ‘agent verb object to recipient’ in B both map (with different transformations) to the semantic representation of the event ACTION(AGENT, OBJECT, RECIPIENT) as illustrated.

Before going into technical detail, we first provide an abstract overview of how the model works, by describing the processes illustrated in figure 1. Again, the input to the model is a
matched <sentence, meaning> pair. As the input sentence is processed word by word, open and closed class words are segregated. Open class words populate the OpenClassArray (OCA), while closed class words populate the ConstructionIndex, which will play a crucial role in assigning the correct sentence–meaning mappings for distinct grammatical constructions. In parallel, the meaning component of the input pair is used to populate the scene event array (SEA) in a predicate–argument representation. Once this initial input processing has occurred, words in the OpenClassArray are translated to predicted referents via the WordToReferent mapping to populate the predicted referents array (PRA). WordToReferent is an associative memory ‘lexicon’ that links words to their meanings. The crucial ‘grammatical’ function is now to map these individual meanings on to their respective roles in the SEA. This mapping is stored in the SentenceToScene mapping, and is specific to each construction type. The structured inventory of these mappings is stored in construction inventory. The key to the model is that each distinct grammatical construction has a unique characteristic configuration of closed class words that is encoded in the ConstructionIndex. Thus, the ConstructionIndex can be used as an index into the ConstructionInventory memory for storing and retrieving the SentenceToScene mapping specific to a given construction. Once the SentenceToScene mapping has been retrieved, the elements in the PredictedReferentsArray can be correctly associated with their functions in the SceneEventArray. Once the model has been trained, it can be tested with new <sentence, meaning> pairs to verify that, for a given sentence, the model can generate the corresponding meaning.

Equations (1)–(7) implement the model depicted in figure 1, and are derived from a neurophysiologically motivated model of sensorimotor sequence learning (Dominey 2000a, Dominey et al. 2003, Dominey and Hoen 2004). In these equations, ‘=’ designates an update of the left side by the right side. The associative memories are implemented as neural networks that correspond to modifiable cortico–cortico and cortico–striatal synapses (see Dominey 1995, 1998a,b, Dominey et al. 1995, Dominey and Ramus, 2000). The ConstructionIndex corresponds functionally to a recurrent cortico–cortical network that has here been simplified for computational complexity reduction (see Dominey et al. (2003) for more extensive presentation of the underlying neurophysiology). Corresponding human neurophysiology can be seen in Hoen et al. (2004) and Dominey and Hoen (2004), and will be presented in section 3. Once the model has been trained on well formed <sentence, meaning> pairs, it can then process new sentences that were not used in training (with the learned vocabulary or lexicon) and generate for these sentences their corresponding meaning. This is the desired output processing of the trained model. Performance is measured by comparing this predicted meaning with the actual meaning that is provided in the <sentence, meaning> input pair.

In this approach, the first step in sentence input processing is to discriminate between open class (e.g. nouns, verbs) and closed class (e.g. determiners, prepositions) words, and to process them in two distinct pathways, as illustrated in figure 1. Newborn infants are sensitive to the perceptual properties that distinguish these two categories (Shi et al. 1999), and in adults these categories are processed by dissociable neural systems (Brown et al. 1999). Similarly, artificial neural networks can also learn to make this function/content distinction (Morgan et al. 1996, Blanc et al. 2003). 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 figure 1, with open class words populating the OpenClassArray.

2.1 Learning word meaning

For this explanation of learning, it is assumed that the inputs to the model, a <sentence, meaning> pair, are valid and well formed. Equation (1) describes the associative memory,
WordToReferent, that links word vectors in the OpenClassArray (OCA) with their referent vectors in the SceneEventArray (SEA). The following holds for all $k, m, 1 \leq k \leq 6$, corresponding to the maximum number of words in the OCA, and $1 \leq m \leq 6$, corresponding to the maximum number of elements in the SEA. For all $i$ and $j$, $1 \leq i, j \leq 25$, corresponding to the word and scene item vector sizes, respectively. In the initial learning phases there is no influence of syntactic knowledge and the word–referent associations are stored in the WordToReferent matrix (equation (1)) by associating every word with every referent in the current scene ($\alpha = 1$), exploiting the cross-situational regularity (Siskind 1996) that a given word will have a higher coincidence with the referent to which it refers than with other referents. This initial word learning contributes to learning the mapping between sentence and scene structure (equations (4)–(6)). Then, knowledge of the syntactic structure, encoded in SentenceToScene, can be used to identify the appropriate referent (in the SEA) for a given word (in the OCA), corresponding to a zero value of $\alpha$ in equation (1). In the current studies this transition is made manually. In actual development, a threshold of confidence in the syntactic knowledge could be used to determine this transition automatically. In this ‘syntactic bootstrapping’ mode, for the new word ‘gugle,’ for example, syntactic knowledge of Agent–Event–Object structure of the sentence ‘John pushed the gugle’ can be used to assign ‘gugle’ to the object of push, rather than ‘blindly’ associating it with all of the possible referents as was done before the SentenceToScene knowledge was acquired. See Dominey (2000b) for a detailed study of this syntactic–semantic interaction.

$$\text{WordToReferent}(i, j) = \text{WordToReferent}(i, j) + \text{OCA}(k, i) \times \text{SEA}(m, j) \times \max(\alpha, \text{SentenceToScene}(m, k)).$$ (1)

### 2.2 Mapping sentence to meaning

In terms of the architecture in figure 1, this mapping can be characterized in the following successive steps. First, words in the OCA are decoded into their corresponding scene referents (via the WordToReferent mapping) to yield the PRA that contains the translated words while preserving their original order from the OCA (equation (2)):

$$\text{PRA}(k, j) = \sum_{i=1}^{n} \text{OCA}(k, i) \times \text{WordToReferent}(i, j).$$ (2)

Next, each sentence type will correspond to a specific *form to meaning* mapping between the PRA and the SEA, encoded in the SentenceToScene array. Two possible such mappings are illustrated in figure 1(A, B). The problem will be to retrieve for each sentence type or grammatical form, the appropriate corresponding SentenceToScene mapping.

### 2.3 Generalizing to different grammatical constructions

Given the capability to discriminate between open and closed class words, described earlier, the problem of using this information to discriminate between different sentence types can be addressed. To solve this problem, recall that each sentence type will have a unique constellation of closed class words and/or bound morphemes (Bates et al. 1982) that can be coded in a ConstructionIndex (equation (3)) that forms a unique identifier for each sentence type. The ConstructionIndex is a 25-element vector. Each function word is encoded as a single
bit in a 25-element FunctionWord vector. When a function word is encountered during sentence processing, the current contents of ConstructionIndex are shifted (with wrap-around) by \( n + m \) bits, where \( n \) corresponds to the bit that is on in the FunctionWord and \( m \) corresponds to the number of open class words that have been encountered since the previous function word (or the beginning of the sentence). Finally, a vector addition is performed on this result and the FunctionWord vector. The desired effect is a unique ConstructionIndex for each construction type. Thus, the appropriate SentenceToScene mapping for each sentence type can be indexed in ConstructionInventory by its corresponding ConstructionIndex. We have previously demonstrated (Dominey et al. 2003) how a recurrent network can perform this ConstructionIndex function as a form of discrimination between sequences of closed class elements:

\[
\text{ConstructionIndex} = f_{\text{circularShift}}(\text{ConstructionIndex}, \text{FunctionWord}).
\]  

(3)

The link between the ConstructionIndex and the corresponding SentenceToScene mapping is established as follows. As each new sentence is processed, we first reconstruct the specific SentenceToScene mapping for that sentence (equation (4)), by mapping words to referents (in PRA) and referents to scene elements (in SEA). The resulting SentenceToSceneCurrent encodes the correspondence between word order (that is preserved in the PRA equation (2)) and thematic roles in the SEA. Note that the quality of SentenceToSceneCurrent will depend on the quality of acquired word meanings in WordToReferent. Thus, syntactic learning requires a minimum baseline of semantic knowledge. Given the SentenceToSceneCurrent mapping for the current sentence, we can now associate this mapping with the corresponding function word configuration or ConstructionIndex for that sentence in the ConstructionInventory, expressed in equation (5). In equations (5) and (6), SentenceToScene is linearized for simplification of the matrix multiplication

\[
\text{SentenceToSceneCurrent}(m, k) = \sum_{i=1}^{n} \text{PRA}(k, i) \ast \text{SEA}(m, i)
\]  

(4)

\[
\text{ConstructionInventory}(i, j) = \text{ConstructionInventory}(i, j) + \text{ConstructionIndex}(i) \ast \text{SentenceToSceneCurrent}(j).
\]  

(5)

Finally, once this learning has occurred, for new sentences the SentenceToScene mapping can be extracted from the learned ConstructionInventory by using the ConstructionIndex as an index into this associative memory, illustrated in equation (6).

Figure 2 illustrates how the model can accommodate sentences with relative phrases that describe two events or ‘dual scenes’ in the same sentence. To accommodate the dual scenes for complex events, equations (4)–(7) are instantiated twice each, 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. This extension is illustrated with an example in figure 2. The novel and ‘revolutionary’ aspect of this analysis of relative phrase processing is that the structural complexity derives directly from that of the semantic or conceptual representation, rather than from an independent and abstract syntactic structural complexity:

\[
\text{SentenceToScene}(i) = \sum_{i=1}^{n} \text{ConstructionInventory}(i, j) \ast \text{ConstructionIndex}(j).
\]  

(6)
Emergence of grammatical constructions

Figure 2. Processing of relative sentences. The relativized input sentence describes two distinct events that are linked by the common argument, ball. Thus, by adding a second event representation in the SceneEventArray component, the meaning structure on to which the sentence can map is made available. The ConstructionIndex thus becomes associated with a pair of mapping structures, one that maps elements from the PRA on to the first event representation, and the second that maps elements from the PRA on to the second event representation.

We evaluate performance of the model by using the WordToReferent and SentenceToScene knowledge to construct for a given input sentence the ‘predicted scene’. That is, the model will construct an internal representation of the scene that 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 equation (2). The referents are then reordered into the proper scene representation in the PredictedSceneArray via application of the SentenceToScene transformation as described in equation (7):

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

When learning has proceeded correctly, the predicted scene array (PSA) contents should match those of the 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. Dominey and Inui (2004) tested the model with 38 different grammatical construction types, some of which are presented in table 1. The model was able to learn all of these constructions, and thus able to use each of them to understand new sentences that had not be presented during the learning phase. We also validated the model using Japanese sentences. This demonstrated that, at least for these constructions in English and Japanese, the configuration of closed class elements uniquely identified each of the constructions and thus provided a basis for storing and retrieving the appropriate mappings. With respect to robustness to noise, we also tested the system with degraded input and observed the desired effect of a ‘graceful degradation’ proportional to the noise in the input (Dominey and Inui 2004).
Table 1. Sample sentences with their meanings (left column) and the corresponding abstract grammatical constructions (right column).

<table>
<thead>
<tr>
<th>Example sentences and meanings</th>
<th>Grammatical constructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The block pushed the cylinder.</td>
<td>1. Agent verb object. (Active)</td>
</tr>
<tr>
<td>Push(block, cylinder)</td>
<td>Verb(agent, object)</td>
</tr>
<tr>
<td>2. The cylinder was pushed by the block.</td>
<td>2. Object was verbed by agent. (Passive)</td>
</tr>
<tr>
<td>Push(block, cylinder)</td>
<td>Verb(agent, object)</td>
</tr>
<tr>
<td>3. The block gave the cylinder to the moon.</td>
<td>3. Agent verbed object to recipient. (Dative)</td>
</tr>
<tr>
<td>Give(block, cylinder, moon)</td>
<td>Verb(agent, object, recipient)</td>
</tr>
<tr>
<td>4. The cylinder was given to the moon by the block.</td>
<td>4. Object was verbed to recipient by agent. (Dative passive)</td>
</tr>
<tr>
<td>Give(block, cylinder, moon)</td>
<td>Action1(agent1, object2, recipient3).</td>
</tr>
</tbody>
</table>

Dual-event relative constructions

<table>
<thead>
<tr>
<th>Example sentences and meanings</th>
<th>Grammatical constructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>6. The block that pushed the cylinder touched the moon.</td>
<td>6. Agent1 that verb1ed object2 verb2ed object3.</td>
</tr>
<tr>
<td>Push(block, cylinder), Touch(block, moon)</td>
<td>Action1(agent1, object2), Action2 (agent1, object3)</td>
</tr>
<tr>
<td>7. The block was pushed by the moon that touched the cylinder.</td>
<td>7. Object3 was action2ed by agent1 that action1ed object2.</td>
</tr>
<tr>
<td>Touch(moon, cylinder), Push(moon, block)</td>
<td>Action1(agent1, object2), Action2 (agent1, object3)</td>
</tr>
<tr>
<td>17. The cat was given from the dog to the block that pushed the cylinder.</td>
<td>17. Ag3 act2ed obj4 to recip1 that act1ed obj2</td>
</tr>
<tr>
<td>Push(block, cylinder), Give(dog, cat, block)</td>
<td>Action1(agent1, object2), Action2 (agent3, object4, recipient1)</td>
</tr>
<tr>
<td>18. The cylinder that was pushed by the block gave the cat to the dog.</td>
<td>18. Obj4 was act2ed from ag3 to recip1 that act1ed obj2</td>
</tr>
<tr>
<td>Push(block, cylinder), Give(cylinder, cat, dog)</td>
<td>Action1(agent1, object2), Action2 (agent3, object4, recipient1)</td>
</tr>
</tbody>
</table>

Dual-event conjoined constructions

<table>
<thead>
<tr>
<th>Example sentences and meanings</th>
<th>Grammatical constructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>27. The block pushed the cylinder and the moon.</td>
<td>27. Agent1 action1 object1 and object.</td>
</tr>
<tr>
<td>Push(block, cylinder), Push(block, moon)</td>
<td>Action1(agent1, object1), Action1(agent1, object2)</td>
</tr>
<tr>
<td>28. The block and the cylinder pushed the moon.</td>
<td>28. Agent1 and agent3 action1ed object2.</td>
</tr>
<tr>
<td>Push(block, moon), Push(cylinder, moon)</td>
<td>Action1(agent1, object2), Action1(agent3, object2)</td>
</tr>
<tr>
<td>29. The block pushed the cylinder and touched the moon.</td>
<td>29. Agent1 action1ed object2 and action2 object3.</td>
</tr>
<tr>
<td>Push(block, cylinder), Touch(block, moon)</td>
<td>Action1(agent1, object2), Action2(agent1, object3)</td>
</tr>
<tr>
<td>30. The moon and the block were given to the cylinder by the cat.</td>
<td>30. Object2 and object3 were action1ed to recipient4 by agent1.</td>
</tr>
<tr>
<td>Give(cat, moon, cylinder), Give(cylinder, block)</td>
<td>Action1(agent1, object2, recipient4), Action1(agent1, object3, recipient4)</td>
</tr>
</tbody>
</table>

3. Neurophysiological basis of the model

One of the interesting predictions that this grammatical construction model makes is that its functional framework for sentence-to-meaning mapping can also be invoked in a non-linguistic sequence processing context. As illustrated in figure 3(B), the model can be presented with a form of artificial grammar task in which a special set of symbols, X and Y for example, can be associated with particular transformations, as in the following: ABCXBAC and ABCYABC. In these two sequences, X corresponds to the performance of a systematic transformation of the input sequence triplet and Y corresponds to an identity operation on that triplet. These abstract structure ‘rules’ can be used to generate an open set of sequences, in which A–C are systematically replaced by other elements (Dominey et al. 1998). The net result is that the processing of grammatical constructions and of these abstract sequences should rely on a common shared neural network. We tested this hypothesis in several experiments.

In a series of neurophysiological experiments, we tested the prediction that patients that are specifically impaired in syntactic comprehension, i.e. using syntactic cues to determine ‘who did what to whom’, would be impaired in a correlated manner in performing these non-linguistic abstract structure processing tasks. In particular, it was observed that left-hemisphere-damaged patients with specific deficits in grammatical structure processing
Emergence of grammatical constructions

Figure 3. Shared mechanism for grammatical constructions and abstract structure processing. (A) Sentence comprehension. (1) Lexical categorization—open and closed class words processed in separate streams. (2) Open class words in OCA are translated to their referent meanings via the WordToReferent mapping. Insertion of this referent semantic content into the PRA is realized in pars triangularis Ba45. (3) PRA elements are mapped on to their roles in the SceneEventArray by the transformation 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 grammatical construction type. The structure mapping process is associated with activation of pars opercularis Ba44. (B) Abstract sequence processing. Lexical categorization takes place for function and content elements of non-linguistic sequences (see Hoen and Dominey 2000). As with sentences, function elements allow retrieval of learned transformation from ConstructionInventory via ConstructionIndex. The transformation processing will continue to be associated with activation of pars opercularis Ba44, but not with activation of pars triangularis Ba45.

demonstrated correlated impairments in their ability to process the abstract structure of non-linguistic sequences (Lelekov et al. 2000, Dominey et al. 2003). According to this model, the correlation derives from the common functional system that performs both tasks (Dominey 2002). The existence of such a shared common system would predict that training that improves performance on one of these tasks should yield improved performance on the other.

In this context, it was observed that the aphasic patients were particularly impaired in the understanding of sentences with a relativized structure, such as ‘It was the apple that Bob caught’. We thus developed a re-education programme using sequences constructed from the abstract structure ABC–BCA that corresponds to the transformation of relativized sentences ‘It was the Apple that Bob Caught’ to the canonical form ‘Bob Caught the Apple.’ The idea is that this abstract transformational structure ABC–BAC corresponds to the grammatical construction that maps the relativized form on to the canonical form, and that something like this mapping is required for comprehension of the relativized sentences. We thus trained six agrammatic aphasics with non-linguistic sequences generated from this particular ABC–BAC
abstract structure in one session per week for 10 weeks, and then compared their sentence comprehension before and after this re-education programme. Interestingly, their performance on active and passive sentences remained unchanged in the before and after comparison, while they displayed a significant improvement in comprehension of the relativized sentences after the re-education (Hoen et al. 2003). This reinforces the purely correlational observations in the argument that the respective processing of these abstract structures and grammatical constructions rely on a partially overlapping neural network.

In a series of event-related potential (ERP) experiments, we continued to accumulate evidence in favour of this hypothesis. Neural activity in large populations of neurons that are aligned in the cortical surface generates electrical dipoles that can be measured with surface electrodes placed in contact with the scalp. Using this technique for measuring brain activity, we first demonstrated that the processing of simple serial structure versus abstract rule structure relied on dissociable neural processes, with abstract structure violations resulting in a P600 response similar to that seen in syntactic structure violations, which was absent in the case of simple serial order violations (Lelekov et al. 2000). Subsequently, a task was devised in which the choice of the abstract structure to apply was guided by the presence of a special ‘function’ symbol, analogous to function words in language. In the sequences ABCZBAC and ABCXABC, the ‘function’ symbol Z indicates that the second triplet is a transformation of the first, based on the rule 123 Z 213, while X indicates an identity operation. Hoen and Dominey (2000) demonstrated that processing of the function symbol Z that indicates a transformation results in a left anterior negativity, or LAN, that is also characteristic of function words that mark subsequent syntactic structural complexity. Subsequent direct comparisons between sequence and sentence processing revealed that the LAN effects for function words and function symbols were statistically indistinguishable (Hoen and Dominey 2004).

This all leads to a question concerning the underlying brain architecture responsible for these neuropsychological and neurophysiological observations. In order to respond to this question, Hoen et al. (2005) tested subjects in sentence and abstract sequence processing while observing their brain activity in a functional magnetic resonance (fMRI) experiment. The structural transformation processing that involves the ConstructionIndex and selection of the appropriate transformation mapping (figure 3) relies on a non-language-specific transformation processing mechanism that corresponds to a local cortical network including Brodmann’s areas 44, 46, 9 and 6. Primate neuroanatomy and human brain imagery indicate that at least part of this network, in particular area 46, corresponds to the frontal component of the dorsal visual stream (Ungerleider et al. 1998). In this context, the dorsal stream is associated with spatial relation processing, and this frontal area would probably participate in a working memory for this type of relation processing, consistent with its proposed role here in structural transformation processing (see also Chang 2002). It has been suggested (Dominey et al. 2003) that the mechanism for storage and retrieval of the appropriate transformations relies on recurrent cortical networks and corticostriatal processing consistent with and extending the procedural component of Ullman’s (2004) sentence processing model. The ConstructionIndex reflects the cortical integration of closed class elements that, via corticostriatal circuitry, retrieve the appropriate transformation implemented in this frontal transformation processing network that includes Brodmann’s area 44 of Broca’s area.

In contrast, for sentence comprehension, the integration of lexico-semantic content into PredictedReferentsArray for subsequent transformation processing corresponds to a ventral stream mechanism that culminates in the pars triangularis (Ba 45) of Broca’s area, in the inferior frontal gyrus region (Ungerleider et al. 1998), consistent with the declarative component of Ullman’s model (2004). Interestingly, though this area (Ba 45) was specifically activated in the language task in our experiment (Hoen et al. 2005), it is more generally characterized as participating in object or semantic (versus spatial) working memory functions (reviewed in
Emergence of grammatical constructions

Ungerleider et al. (1998), consistent with its proposed role here for semantic integration (see Dominey and Hoen 2005). Thus, we begin to have access to the possible neurophysiological foundations of the processing of grammatical constructions.

4. Insight into evolution and development

As presented earlier, once the model learns the grammatical construction that defines the mapping between ‘John gave the ball to Mary’ and $<\text{GAVE}(\text{John, Ball, Mary})>$, this knowledge can then be used to generate and understand new sentences of the form ‘Agent action object to recipient’. In child language development, however, this abstract construction capability is preceded by the use of fixed, less abstract constructions such as ‘Gimme that’ that are sometimes referred to as ‘holophrases’ because they are unitary, non-composed utterances (reviewed in Tomasello 2003). Interestingly, the same transition from holophrase to more abstract construction has also been suggested on the evolutionary time-scale (Wray 2000). What then would be the mechanism of transition from holophrase to a more abstract construction type, and how might this transition be related to the formation of functional lexical categories (like noun versus verb)? In this context, we can now consider what happens when the category of concrete ‘things’ (in the context of concreteness or imaginability suggested by Gillette et al. (1999)) or nouns, but not verbs, becomes processed as a variable category or ‘slot’ with respect to the ConstructionIndex. In this intermediate phase, nouns would be processed as open class elements (as described above). Verbs, however, would not yet be considered as open class elements, and thus would be bound to distinct constructions via the ConstructionIndex. This would lead to an item-based ‘verb island’ phase in which constructions are based around verbs (that continue to contribute their identity to the ConstructionIndex) with free variables for the noun arguments. The subsequent emergence of verbs as a functional category and the resulting generalization over verbs would allow for the full abstract construction capability that has been demonstrated above. This free versus fixed distinction was perhaps first formalized in the pivot grammar of Braine (1963), in which certain fixed ‘pivot’ words such as ‘all’ in ‘all gone’, ‘all done’, ‘all dressed’ are bound into the pivot frame, while the following word is defined by a slot that can take free arguments.

In this context, the progressive emergence of generalized functional lexical categories becomes correlated with the emergence of progressively more abstract constructions that generalize over those categories. As revealed in table 2 this developmental construction framework was validated with a series of three simulations using the following sentences:

(1) The moon touched the block.
(2) The cylinder pushed the moon.
(3) The block took the cylinder.
(4) The moon gave the cylinder to the block.
(5) The moon pushed the cylinder.
(6) The block touched the cylinder.
(7) The moon gave the block to the cylinder.

In the first simulation of the ‘holophrase’ developmental phase, the input processing was structured such that all of the open class words would directly contribute to the ConstructionIndex and the OpenClassArray, as described above. Closed class words contributed to the ConstructionIndex as before. Under these conditions, the model learned each sentence as a distinct construction, successfully mapping each construction on to its meaning.
Table 2. Simulation results.†

<table>
<thead>
<tr>
<th>Construction type</th>
<th>Open class element types</th>
<th>Number of constructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Holophrase</td>
<td>None</td>
<td>7</td>
</tr>
<tr>
<td>Verb island</td>
<td>Nouns</td>
<td>4</td>
</tr>
<tr>
<td>Abstract constructions</td>
<td>Nouns and verbs</td>
<td>2</td>
</tr>
</tbody>
</table>

†With no open class elements, the holophrase simulations yield one construction per sentence. When nouns become ‘variables’ in the verb islands, the sentences can be represented by four distinct constructions. When nouns and verbs are open class variables, the set of sentences can be represented by the two transitive and ditransitive constructions.

In the second simulation of the ‘verb island’ phase, the input was processed as in the holophrase condition, with the exception that nouns no longer contributed directly to the ConstructionIndex. Instead, their status as open class words and their relative sentence positions were encoded in the ConstructionIndex as in the simulations described in previous sections. Under these conditions, it would be expected that constructions would be organized around specific verbs, as in the verb island scenario (Tomasello 2003). Indeed, the model learned four distinct verb island constructions corresponding to touch, push, take and give that could generalize and take different arguments in the agent, object and recipient roles.

In the final simulation of this section, both nouns and verbs were processed as open class elements, as described in the previous sections. Thus, there was no longer any noun- or verb-specific component encoded within the ConstructionIndex, so the item-based nature of constructions was eliminated, yielding the abstract constructions as previously observed. Under these simulation conditions, the model learned two distinct construction types, corresponding to the transitive and ditransitive constructions. Thus, as constructions become increasingly abstract, there is an increasing degree of data compression with respect to the size of the construction inventory required to accommodate a given set of utterances.

These simulated phases may be of use in the interpretation of developmental observations. Before 3 years of age, a child will not use a verb in a transitive construction if he/she has not heard it used that way. However, the child will regularly use other verbs in transitive forms that he/she has heard in such forms (reviewed in Tomasello 2003). The item based, or verb island phase, that we simulated above can explain this condition. Consider the sentences ‘The moon took the block’ and ‘The moon took the block to the cylinder’. After training on the seven sentences above in the verb island phase, the model will accept this new use of took in a transitive, but not in a ditransitive construction. However, once in the abstract phase, the ditransitive construction becomes ‘liberated’ and can accept ‘took’ as a ditransitive verb even if this ditransitive configuration of ‘took’ had not been experienced in training. A more detailed treatment of this development can be found in Dominey (2006).

5. Validation of the model in a grounded agent system

Given this mechanism for learning grammatical constructions as <sentence, meaning> mappings along with consideration of its functional neurophysiology and possible contributions to the transition from holophrases to abstract constructions, it remains to be seen whether the system can actually function in a perceptually grounded context. In order to respond to this question, the model was inserted into such a perceptually grounded context in which physical events were performed by a human subject who simultaneously narrated his/her actions, and the resulting spoken sentences and visual scenes were automatically processed to generate
<sentence, meaning> pairs for training the model. Figure 4(A) illustrates the physical set-up in which the human operator performs physical events with toy blocks in the field of view of a colour CCD camera. Figure 4(B) illustrates a snapshot of the visual scene as observed by the image processing system. Figure 4(C) provides a schematic characterization of how the physical events are recognized by the image processing system. Using this platform, the human operator performs physical events and narrates his/her events. An image processing algorithm extracts the meaning of the events in terms of action(agent, object, recipient) 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 the temporal schematic template illustrated in figure 4(C). While details can be found in Dominey (2003), 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 (e.g. Siskind 2001, Steels and Baillie 2002). Together with the event extraction system, a commercial speech to text system (IBM ViaVoice™)
was used, such that each narrated event generated a well formed <sentence, meaning> pair. The <sentence, meaning> pairs were provided as training input to a learning model whose architecture is depicted in figure 1.

It was thus demonstrated (Dominey 2005) that the model could successfully learn a rich variety of grammatical constructions with active, passive and relativized structures, each of which allowed the system to generate the correct meaning for new sentences that had not been used in training. These initial learning results were quite promising, and provided the bases for testing this learned language capability in an interactive human–robot communication scenario. Technically this raises several issues, 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.

Recall that 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 five grammatical constructions in table 3. These correspond to constructions with two and three verb arguments in which each of the different arguments can take the focus position at the head of the sentence. On the left are presented example sentences, and on the right, the corresponding generic construction.

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 sentences of type 3, 4 or 5, respectively. 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. Note that sentences (1)–(5) are specific sentences that exemplify the five constructions in question, and that these constructions each generalize to an open set of corresponding sentences. Thus, given an input meaning in the form event(arg1, arg2, arg3) and an optional focus item (one of the three arguments), the system will deterministically choose the appropriate two or three argument construction, with the appropriate focus structure, in a pragmatically relevant manner.

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 dialogue management with a flexible and powerful graphical user interface 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 file-transfer protocol and socket protocol. 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. This allows the learned constructions to be used in an interactive human–machine interface in which the human performs

Table 3. Sentence and corresponding constructions for robot language generation (extracted from the extended construction set in table 1).

<table>
<thead>
<tr>
<th>Sentence</th>
<th>Construction</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The triangle pushed the moon.</td>
<td>Agent event object.</td>
</tr>
<tr>
<td>2. The moon was pushed by the triangle.</td>
<td>Object was event by agent.</td>
</tr>
<tr>
<td>3. The block gave the moon to the triangle.</td>
<td>Agent event object to recipient.</td>
</tr>
<tr>
<td>4. The moon was given to the triangle by the block.</td>
<td>Object was event to recipient by agent.</td>
</tr>
<tr>
<td>5. The triangle was given the moon by the block.</td>
<td>Recipient was event object by agent.</td>
</tr>
</tbody>
</table>
actions and then the system describes the actions and uses the appropriate grammatical form in order to respond to questions (Dominey et al. 2004). Indeed, within this interactive dialogue context, the functional requirement for the different constructions such as (3)–(5) can be seen in the context of the event Gave(Triangle, moon, cylinder). Depending on whether the human asks about what happened to the triangle, the moon or the cylinder, the system responds ‘The triangle gave the moon to the cylinder’, ‘The moon was gave to the cylinder by the triangle’, or ‘The cylinder was gave the moon by the triangle’ (Dominey et al. 2004, Dominey and Boucher 2005b).

6. Discussion

From a functional perspective, the essential problem that the proposed grammatical construction model is designed to address is that of mapping grammatical structure of sentences onto the semantic structure of their meanings. As illustrated in figure 1 (A, 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 \(<\text{sentence, meaning}>\) mappings in terms of a generative grammar, these mappings can be represented directly in a structured inventory of grammatical constructions that are nothing more than these mappings. Growing evidence from studies of both human language development (Tomasello 1999, 2003) and adult processing (Sanford and Sturt 2002, Ferreira 2003) indicate that a substantial component of language behaviour can be accounted for in this manner. That is, that language production and comprehension are based on the reuse (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 figure 1. This does not exclude the existence of truly generative mechanisms for construction and decoding new grammatical forms.

If the language capability consists of a structured inventory of grammatical constructions, then a problem remains concerning how this inventory is managed. This is where the second great principle of developmental linguistics comes in: the cue competition hypothesis of Bates et al. (1982). They proposed 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, which allows the determination of ‘who did what to whom’. Thus, as illustrated in figure 1, the ensemble of closed class words together form a ‘construction index’ that serves as an index into an associative memory that stores the appropriate transformations. This memory store is referred to as the Construction-Inventory in figure 2. In a series of experiments (Dominey 2003a, b), it has been demonstrated that the system can thus learn an extensive set of grammatical constructions, including those in table 1.

This analysis of the model attempts to shed light on two related issues in the domain of the emergence of language. The first is that, both from an evolutionary and a developmental perspective, the behavioural emergence of language follows a certain form of progression that eliminates the need for a single mechanism that does ‘everything at once’. As suggested by Wray (2000) and Tomasello (2003), this progression begins with a limited collection of ‘holophrases’ that communicate packages of meaning in a manner that cannot readily be decomposed for the formation of ad hoc new messages. Through processes that are beginning to become more clearly understood and documented, particularly in development, these holoconstructions begin to take on more abstract forms via the liberation of functional slots or variables that can be filled in by a free set of arguments. This leads to a behavioural progression from holophrases via verb islands to full-blown abstract argument constructions, in
a hierarchy of progressively increasing generativity. If this analysis is at least partially correct, it radically changes the perspective on the emergence of language, as the proposed progression can be much more directly mapped on to functional/computational processes that can in turn be associated with their corresponding neurophysiological bases. In this context, one might ask how the current system that has been illustrated using constrained inputs will generalize to real data. The system has learned on the order of $N \times 10^4$ constructions—Can it learn $10^2$, $10^3$, $10^4$? To the extent that constructions are identifiable either by their internal closed class structure, or by extra-sentential cues such as discourse context, and to the extent that these factors can be encoded, the system will be extendable; but the tough issue is whether the system can learn to accommodate new constructions that it has not previously encountered. In this context, Miikkulainen (1996) has demonstrated a hybrid neural network system that can accommodate novel well-formed relative phrase structure based on a pre-wired parsing capability. Our future research will demonstrate that this kind of generative capability can also be acquired in a developmental context.

This leads to the second, neurophysiological, issue. Data reviewed above suggest that, in the functional neurophysiology of grammatical constructions, the structural transformations for mapping semantic components into the global meaning structure are realized in part by frontal cortical regions, including Brodmann’s areas 6, 44 and 46 that correspond to the spatial transformation working memory system of the dorsal pathway. In parallel, the insertion of the semantic meaning components into this transformation mechanism is realized in part by the frontal extension of the semantic object property system of the ventral pathway. As noted by Ungerleider et al. (1998), a relative displacement of frontal cortical areas for spatial working memory in humans with respect to monkeys may be related to the emergence of language-related processing during the course of human evolution. Thus, though clearly there are still more questions than answers, this emergence approach provides a perspective in which the evolution and development of language follow a trajectory of increasing complexity and functionality. Ideally, as this work hopes to illustrate, this trajectory provides a progressive series of targets for explanation that can provide a useful guide for progress in understanding the emergence of language.

Acknowledgements

This work was supported in part by the HFSP MCILA Project, the ACI Integrative and Computational Neuroscience ACI TTT Initiatives (Paris), the ESF Eurocores OMLL Project, and the LAFMI project.

References


Emergence of grammatical constructions


P.F. Dominey and M. Hoen, Structure mapping and semantic integration in a construction-based neurolinguistic model of sentence processing, Cortex, in press.


