A neural architecture for grounded cognition: Representation, structure, dynamics and learning

Frank van der Velde, Marc de Kamps

Abstract—**Human cognition is characterized by three important features: productivity, dynamics and grounding. These features can be integrated in a neural architecture. The representations in this architecture are not symbol tokens, that can be copied and transported. Instead, the representations always remain "in situ", because they are grounded in perception, action, emotion, associations and (semantic) relations. The neural architecture shows how these representations can be combined in a productive manner, and how dynamics influences this process. The constraints that each of these features impose on each other could result in an architecture in which the local and the global aspects of cognition interact in processing and learning.**

I. Introduction

In this paper, we discuss a neural architecture that aims to integrate three important features of human cognition: productivity, dynamics and grounding. Productivity refers to the combinatorial nature of cognition, as found in language, reasoning and vision. Dynamics refers to the ability to interact with the environment in a dynamical way. Grounding refers to the nature of cognitive representations: representations for concepts are always grounded in perception, action, emotion, and associations, and embedded in semantic relations and other cognitive structures.

Individually, each of these features is important for cognition. But, in particular, their combination is important, because of the constraints that they impose on each other. Productivity is found in computer systems, but it is unlimited, and not dynamic and grounded. Dynamical systems are abound in nature, but most of them are not cognitive. Grounded representations form the backbone of associative processing in all neural systems, but associative processing is not productive. So, the combination of productivity and grounding, instantiated in a dynamical system, is rare. The human brain is perhaps the only known system of this kind. An architecture that integrates these features is therefore important for understanding the nature of the brain, and for understanding the nature of human-like cognition.

We begin by briefly discussing the relation between productivity and grounding in an architecture based on symbol manipulation.

Manuscript received November 30, 2007. F. van der Velde is with the Leiden Institute for Brain and Cognition, Leiden University, 2333 AK, Leiden, The Netherlands (corresponding author, phone: +31-71-5273637; e-mail: vdvelde@fsw.leidenuniv.nl). M. de Kamps is with the Biosystems Group, School of Computing, University of Leeds, LS29JT Leeds, United Kingdom (e-mail: dekamps@comp.leeds.ac.uk).

II. Productivity

An example of productivity in cognition is found in language processing. We can produce and understand novel sentences, consisting of familiar words. Perhaps we cannot assume that this productivity is infinite (as is typically assumed in linguistics), because that would involve arbitrarily complex and long sentences. But the productivity is nevertheless huge. Miller [1] estimated that humans can understand a set of sentences in the order of 10^{20} or more. One can call this set the "performance set" of human language, in the sense that an average language user would be able to understand any sentence chosen from this set. To put the magnitude of the performance set in perspective: the number of seconds in the (estimated) life time of the universe (since the Big Bang) is in the order of 10^{17} to 10^{18}. This number shows that there are severe limits on the ratio between the number of sentences that can be learned in the lifetime of a language user, and the number of sentences in the performance set [2]. Therefore, the ability to produce and understand language requires some form of computational productivity. In turn, the need for computational productivity for language processing has been used as an argument that human language processing, and higher level cognition in general, depend on forms of symbol manipulation [3,4].

III. Symbol manipulation

Symbol manipulation depends on the ability to make copies of symbols and to transport them to other locations. Newell [3] argued that symbols are needed for cognition because only a limited amount of information can be stored physically at a given location. The symbol token is then needed to obtain more information when that is required for a given process. In his words (p. 74): 'The symbol token is the device in the medium that determines where to go outside the local region to obtain more structure. The process has two phases: first, the opening of access to the distal structure that is needed; and second, the retrieval (transport) of that structure from its distal location to the local site, so it can actually affect the processing. (…) Thus, when processing "The cat is on the mat" (which is itself a physical structure of some sort) the local computation at some point encounters "cat"; it must go from "cat" to a body of (encoded) knowledge associated with "cat" and bring back something that represents that a cat is being referred to, that the word "cat" is a noun (and perhaps other possibilities), and so on.'

978-1-4244-1821-3/08/$25.00©2008 IEEE

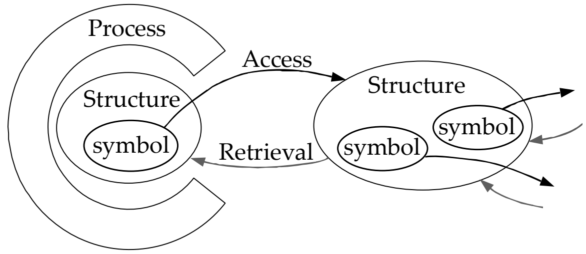

Fig. 1. Symbols provide distal access in an architecture based on symbol manipulation (adapted from [1]).

Fig. 1 illustrates the use of symbol tokens in this kind of architecture. One part of the architecture (e.g., a sentence processor) processes the sentence structure of “The cat is on the mat.” Other parts of the architecture store information about words, e.g., whether they are nouns, pronouns or verbs. So, when the sentence processor needs information about the word type of a word, e.g., of the word “cat”, it makes a copy of the symbol token for “cat” and transports it to another part of the architecture that stores information about word types. This part receives the copy of the symbol token “cat”, retrieves that “cat” is a noun, and transports a copy of “cat is a noun” back to the sentence processor. The sentence processor can use that information to build a structure of the sentence “The cat is on the mat.”

When a symbol token is copied and transported from one location to another location, all its relations and associations at the first location are lost. This is in fact the motivation for using symbol tokens [3], because the amount of information that can be stored at a given location is (physically) limited. Symbol tokens are then used to transport information to and from other locations.

Because relations and associations are restricted to the location at which a symbol token is used, symbol tokens are ungrounded. That is, at each location only a limited amount of information related to the symbol token is stored. For example, the perceptual information related to cats is lost when the symbol token “cat” is copied and transported to a location outside the location where perceptual information is processed.

The ungrounded nature of symbol tokens has consequences for processing. Different kinds of information have to be stored in relation to a given symbol token. For example, the architecture needs to store perceptual information about cats (visual, auditory, or tactile), it has to store information needed to produce actions related to cats (e.g., stroking, speaking the word “cat”), it has to store information needed to read the word “cat”, or semantic information related to cats (e.g., “is animal”), or associations about cats (e.g., cats vs. dogs). In the architecture depicted in fig. 1, these different kinds of information are stored and processed at different locations. They can be related to each other only by an active decision to gain access to other locations, to retrieve the information needed. Without such active decisions, the different kinds information related to cats remain unrelated.

The ungrounded nature of symbol tokens has consequences for learning as well. Because different kinds of information concerning a concept are stored at different locations, the changes that occur at one location have no effect on the information stored at another location, unless again and active process is initiated to transfer the new information to the other locations. Furthermore, new information that is learned in an implicit manner cannot be transferred in this way.

IV. Grounded Cognition

A. Representations

Fig. 2 illustrates how the sentence “The cat is on the mat” is represented in the neural architecture of grounded cognition (ignoring the determiner “the”). The circles and ovals in fig. 2 represent populations of neurons.

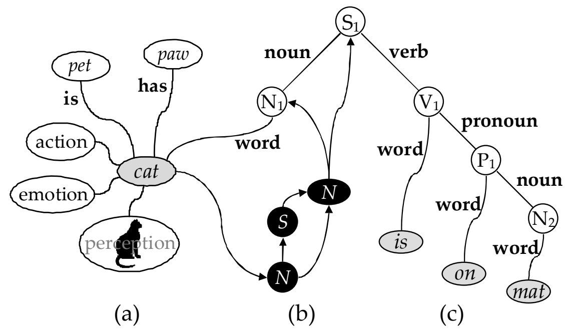

Fig. 2. Embedding of the grounded neural structure for the word “cat” in the bare neural sentence structure for “The cat is on the mat” and the control circuit for structure binding (based on [2]). The labeled connections are conditional connections. All circles and ovals represent (local) populations of neurons.

In grounded cognition, the representation of a word (concept) does not consist of a symbol token that can be copied and transported. Instead, each representation consists of a network structure, which can be interconnected temporarily to other network structures (representing other words) to form the representation of a sentence.

Fig. 2a illustrates the grounded representation for the word “cat.” It consists of a network that interconnects all aspects related to the word (concept) “cat.” This includes all perceptual information related to cats, action processes related to cats (e.g., the experience of stroking a cat, or the ability to pronounce the word “cat”), all emotional content associated with cats, and all other information related to or associated with cats, such as the semantic information that a cat is a pet, or the (negative) association between cats and dogs.

The grey oval in the center, labeled “cat”, plays an important role, because it interconnects the neural structures in the representation of “cat”, and because it can be connected to sentence structures, as in “The cat is on the mat" (thus, it can embed the word structure of "cat" in sentence structures). However, it would be wrong to see this oval itself as the neural representation of "cat", because its representational value derives entirely from the network structure of which it is a part. Therefore, even if it were technically possible to make a copy of the oval (e.g., of its internal network structure or pattern of activation) and transport that to another location (as in the architecture of fig. 1), it is useless to do so. When the internal network structure or pattern of activation of the oval is copied and transported, it is separated from the remaining network structure of the representation "cat", and thus it loses its representational value. For example, if this copied and detached oval were connected to the structure of a sentence, it would not represent the word "cat" in that sentence (or any other word).

The grounded nature of word representation in the brain has been investigated in a number of studies in recent years. In a word comprehension task, modality-specific brain activation was found [5], in which motor related words (nouns and verbs) activated motor related brain areas and sensory related words (nouns and verbs) activated sensory related areas. A similar activation difference was observed [6] between action related words (verbs), in which parts of the premotor cortex that code for specific actions (related to mouth, hand or leg actions) were also activated by the words describing these actions (e.g., "bite", "grasp" or "kick"). In this study, the words were presented sentence contexts (e.g., "I bite an apple", "I grasp a knife" or "I kick the ball"). So, the observed word activations strongly suggest that the representations of the words remain grounded when the words are part of sentences (word combinations).

B. Conditional connections

The labeled connections in fig.2 are of a special kind. In a connection between two neurons, activation flows from the pre-synaptic neuron to the post-synaptic neuron, when the pre-synaptic neuron is active. This kind of connection is associative, because activation flows without any form of control. In contrast, the labeled connections are conditional: activation flows only when the condition indicated by the label is met. For example, the activation of the label **is**, e.g., by the query "What is a cat?", opens the connection between the network structures for "cat" and "pet", so that "pet" can be given as an answer to this query [7]. Conditional connections are needed to represent relations in network structures [7]–[10]. They can be implemented by specific neural circuits, e.g., such as circuits based on disinhibition [7], [8], or by conjunctive connections [9], or by specific activation rules [10]. Conditional forms of activation have been found in brain studies of rule behavior in monkeys [11].

The other grey ovals in fig. 2c, labeled "is", "on", and "mat", belong to the grounded network structures for the words "is", "on", and "mat", respectively.

C. Structures

To represent a sentence, the network structures for the words (or "word structures", for short) are (temporarily) bound to neural "syntax structures" that represent elements of syntactic information. In fig. 2c these are "sentence" (S_1), "noun phrase" (N_1 and N_2), "verb phrase" (V_1), and "prepositional phrase" (P_1). The syntax structures are then bound to each other in agreement with the structure of the sentence [7]. Binding is achieved by reverberating ("delay") activity, which is neural activity that persists for a while, even when the stimulus that initiated the activity is gone. The delay activity creates a conditional connection, by which activation can flow between the bound structures. Because the delay activity decays after a while, the binding decays as well (although it can be transferred to long-term memory under certain conditions [7]).

The process of binding word and syntax structures is controlled by a neural "control" circuit. A part of this circuit is shown in fig. 2b (the black nodes and oval). The circuit recognizes the nature of a word (e.g., whether it is a noun or verb), and initiates the binding of the word structure of that word to a syntax structure, and the binding of the syntax structure to other syntax structures. The full control circuit is also influenced by the current state of the sentence structure (fig. 2c). The part of the control circuit in fig. 2b recognizes that "cat" is a noun (the black node *N*), and that it is the first word (the black node *S*). Their combination activates the black oval (*N*) that binds "cat" to a noun phrase structure (N_1), and binds N_1 to a sentence structure (S_1), which represents "cat" as the subject of the sentence.

Figs. 1 and 2 illustrate the difference between grounded cognition and symbol manipulation. In fig. 2, no symbol tokens are copied, transported or retrieved. For example, there is no need to go to a distal location and retrieve a symbol token that represents that "cat" is a noun. Instead, the word structure for "cat" is connected to the control circuit, and it activates the noun (*N*) unit in that circuit. The same unit will be activated by other nouns. So, there is no need to store a list of words and their syntactic labels.

Instead of an architecture consisting of different processors that copy and transport symbol tokens, the architecture for grounded cognition consists of specific connection structures and selective processes of activation. The sentence structure in fig. 2c interconnects the word structures for "cat" and "mat" in such a way that the query "Where is the cat?" produces the activation of "mat" and the query "Who is on the mat?" produces the activation of "cat." The architecture can also represent the sentence "The cat is on the cat" (or "The mat is on the mat"). In "The cat is on the cat" the word structure for "cat" is not copied. Instead, the same word structure is used (the one in fig. 2a), connected (bound) to N_1 and N_2 in fig. 2c.

V. Visibility of Grounded Representations

Fig. 2 illustrates an important aspect of grounded representations: they are always accessible or "visible", even when they are a part of a representation of a combinatorial structure, such as a sentence. Because the representation of "cat" remains visible in the structure of a sentence like "The cat is on the mat", the connection structure of "cat" can be activated when "cat" is part of a sentence. In this way, the word structure for "cat" can influence the binding process that produces the sentence structure, or the word "cat" can be produced, e.g., in pronouncing the sentence. Furthermore, new information given by the sentence, e.g., an association between "cat" and "mat", can be integrated with the word structures for "cat" and "mat." The visibility of grounded word representations in this architecture is in agreement with the observations in [6] discussed above.

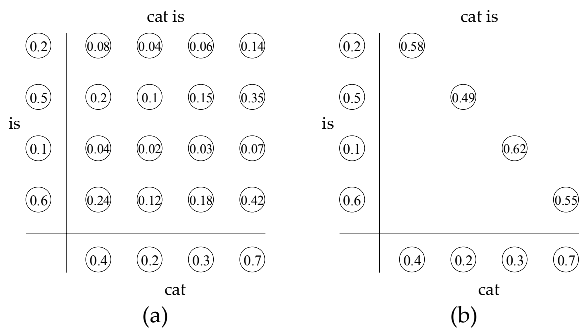

Fig. 3. Loss of grounding in tensor and reduced vector representations of combinatorial structures. In (a), vectors for "cat" and "is" are combined into a tensor for "cat is" [7]. In (b), vectors for "cat" and "is" are combined into a reduced vector for "cat is" [8].

Fig. 3 illustrates neural representations of combinatorial structures in which the word representations are not visible. In fig. 3a, neural vectors represent the words "cat" and "is", and the combination "cat is" is then formed by the tensor product of these vectors [12]. Continuing in this manner, a tensor representation can be made of the sentence "cat is on mat". It is clear that the vector for "cat" is not accessible (visible) in the tensor for "cat is" (or that of "The cat is on the mat").

In fig. 3b, the vectors for "cat" and "is" are combined in a reduced vector representation of "cat is" [13]. A reduced vector representation produces a representation of a combinatorial structure that has the same magnitude (dimension) as the vectors of the components, which eliminates the problem of the increasing dimensions of tensors for hierarchical combinatorial structures [13]. Continuing in the same manner, a reduced vector representation can be made of the sentence "cat is on mat". Again it is clear that the vector for "cat" is not accessible (visible) in the reduced vector for "cat is" (or that of "The cat is on the mat").

The vector for "cat" itself can be grounded, and it can be retrieved from the tensor of reduced vector representation of the sentence "The cat is on the mat". But when the vector is retrieved, the structure of the sentence is lost (e.g., there is no way to tell the difference between "cat is on mat" or "mat is on cat" is that case). In the combinatorial representation as given by tensor coding or reduced vector coding, the vector for "cat" is not accessible, and hence it is not grounded.

Thus, when "cat" is part of "The cat is on the mat", it cannot activate the rest of its neural structure to influence the process that produces the sentence structure, or any other process in which the word structure for "cat" is a part. Furthermore, new information given by the sentence, e.g., a relation between "cat" and "mat", cannot be integrated with the word structures for "cat" and "mat."

Fig. 3 illustrates that accessibility (visibility) is a necessary requirement for a representation to be grounded. But it is not sufficient. Symbol tokens in the architecture of fig. 1 are accessible (visible), but they are not grounded. A representation is grounded when it is accessible (visible) and remains "in situ", that is, when its embedded information structure always remains intact.

VI. Productivity and Grounded Representations

The productivity of natural language requires that word representations can be combined on the fly, to represent novel sentences. Symbol manipulation achieves this by copying and pasting symbols. Because grounded representations remain "in situ", copying and pasting is not possible in grounded cognition. Instead, grounded word representations are temporarily bound in a sentence structure (fig. 2). It is important to understand the difference between this temporal sentence structure and a semantic network. The latter consists of relations between words and concepts that have been learned over time. Because it has been learned, the set of relations is limited, and it can be learned in a gradual manner. In the Rumelhart semantic network [9], a word like "cat" is presented to a multiplayer feedforward network, together with a word like "is". The network is then trained to produce words like "pet" or "animal" as an output for the combination "cat" and "is". The word "is" thus operates as a condition to answer the query "What is a cat?". Learning these conditional relations in a feedforward network works for semantic relations, but it is too slow to build a sentence structure on the fly. For example, a language user can understand a sentence like "The terminator wants to be president", even if it is a novel sentence. The language user can answer a query like "What does the terminator want?", because sentences of the type "X wants to be Y" are familiar, and the words "terminator" and "president" are familiar as well.

VII. Grounded Cognition and Dynamics

A. Ambiguity resolution

Because grounded representations are accessible (visible), they can influence the combinatorial structures of which they are a part.

For example, consider the sentence:

Time flies like an arrow

The usual interpretation of this sentence is that of a metaphor, that states that time changes very fast [14]. In this interpretation, "time" is a noun and "flies" is a verb. But there are (at least) four other interpretations of this sentence [14]. In some of these, "time" is a verb and "flies" is a noun. So, the sentence could mean that one should measure the speed of flies in the way as one measures the speed of an arrow.

Ambiguities of this kind are common in language and cognition in general, but humans often do not notice them [15], [16]. This is also the case for the sentence above. Whereas humans usually end up with one interpretation, a computer program of sentence analysis gave all five interpretations [14].

Fig. 4 illustrates how ambiguity resolution could occur in the neural architecture of grounded cognition. In the architecture, dynamical interactions can occur between encoded sentences [7]. Similar interactions can also influence the binding process, that is, the process by which a sentence structure is formed.

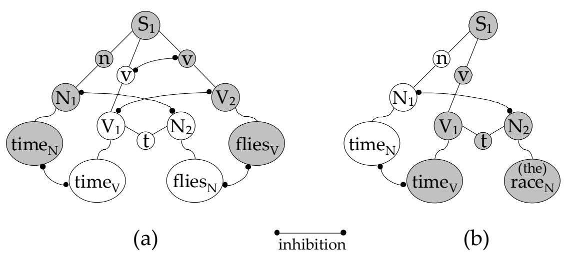

Fig. 4. Competing sentence structures in the neural architecture of grounded cognition. The ovals and large circles depict the structures as in fig. 2. The small circles depict the circuits that activate conditional connections as in fig. 2 (n = **noun**, v = **verb,** t = **theme**). Grey nodes are active.

Fig. 4a shows the competing sentence structures of "time flies." The word "time" activates two word structures, one for "time" as a noun ("time$_N$") and one for "time" as a verb ("time$_V$"). in the same way, "flies" activates "flies$_N$" and "flies$_V$." All word structures initially bind to the sentence structure.

But there is a dynamic competition within the sentence structure and between word structures. In particular, the word structures for "time$_N$" and for "time$_V$", and those for "flies$_N$" and "flies$_V$" inhibit each other, which implements the constraint that a word can have only one interpretation at the same time. Within the sentence structure there is inhibition between the circuits that activate conditional connections of the same kind (in fig. 4a those for the **verb** connections), and inhibition between similar syntax structures (e.g., between the noun phrases N_1 and N_2 and between the verb phrases V_1 and V_2).

In the binding process that creates the sentence structure, the conditional connections (e.g., those for **noun**, **verb,** and **theme**) are active, so that word structures activate the syntax structures to which they are bound (and vice versa). Through these connections, word structures also activate each other.

The dynamic interaction in the architecture is resolved when there is a clear advantage for one of the competing structures [7]. In fig. 4a, an advantage for one of the structures can arise from the fact that the interpretation of "time" as a noun is more frequent than the interpretation of "time" as a verb. In that case, the activation of "time$_N$" will be stronger than that of "time$_V$", so that the first inhibits the second. Then, "flies$_V$" inhibits "flies$_N$", because "flies$_V$" is activated by "time$_N$" through the sentence structure, whereas N_2 is inhibited by N_1 (this inhibition becomes stronger with increasing activation of "time$_N$"). In this way, the grey structure in fig. 4a remains as the active structure to which the rest of the sentence, "like an arrow", binds.

Fig. 4b shows how "time$_V$" will inhibit "time$_N$" in the sentence structure "time the race". The noun phrase "the race" matches "time$_V$" as its theme (verb object). Through the active **theme** conditional connection, "race$_N$" enhances the activation of "time$_V$", which affects the inhibition of "time$_N$" by "time$_V$." Furthermore, N_1 is inhibited by N_2. The resulting activation pattern favors the interpretation of the sequence "time (the) race" as a verb and its object over the interpretation of this sequence as two nouns.

The outcome of a dynamic competition in fig. 4 depends on the direct competition between the word structures and the indirect competition between these word structures through the sentence structure to which they are bound. This double interaction process illustrates the importance of the accessibility (visibility) of grounded representations in combinatorial (e.g., sentence) structures.

B. Misinterpretations

Humans can also make mistakes in the interpretation of a sentence, presumably because they opt to soon for a particular interpretation of the sentence. An example is a "garden path" sentence like [14]:

The man who hunts ducks out on weekend

In this sentence, the word "ducks" is usually interpreted as a noun (and the object/theme of "hunt"), whereas "ducks" belongs to "ducks out", which is the verb of the main sentence. The interpretation of "ducks" as a noun prevents the correct interpretation of the phrase "out on weekend", and thus results in the misinterpretation of the sentence.

Fig. 5a illustrates how this misrepresentation can result from the dynamic interaction between grounded representations. The sentence structure in this example is extended by a syntax structure for relative clause (C_1). When the word "ducks" appears after the sequence "The man who hunts", it can be interpreted as the object/theme of "hunts" in the relative clause.

This interpretation is strengthened by the association between "hunts" and "ducks", which results in additional activation of the word structure "ducks$_N$" by the word structure "hunts." As a result, the word structure "ducks$_N$" prevents the activation of the word structure "ducks$_V$" even when the word "out" appears. This prevents the embedding of the phrase "out on weekends", creating the misrepresentation of the sentence.

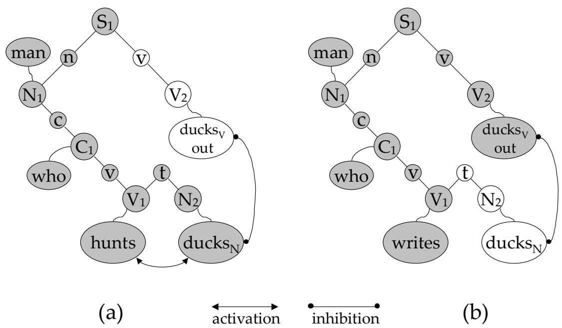

Fig. 5. Competing sentence structures in the neural architecture of grounded cognition. The ovals and circles depict the structures as in fig. 2. (n = **noun**, v = **verb,** t = **theme,** c = **clause**). Grey nodes are active.

Fig. 5b illustrates the role of the association between the word structures in fig. 5a. The words "writes" and "ducks" are not associated, so the activation of the word structure "ducks_N" by the word structure "writes" is missing. The appearance of "out" can now result in the (stronger) activation of the word structure "ducks_V", so that the phrase "out on weekends" can be embedded in the sentence structure.

VIII. Grounded Cognition and Learning

A. Associations

Figs. 4 and 5 illustrate the importance of the visibility of grounded representations in learning associations. For example, the association "hunts" and "ducks_N" grows over time, based on experience. It could result from hearing or reading sentences like "The man hunts ducks" and from seeing events of this kind. These experiences will result in an associative connection between the word (concept) structures of "hunts" and "ducks_N" (e.g., based on activation depended learning, such as Hebbian learning [17]). But this can work only when the concept structures that are activated by perception are the same as the concept structures activated in combinatorial structures like sentences. Thus, the concept structures have to be visible in combinatorial structures and have to be grounded (in visual perception, in this case [17]). This kind of learning behavior is assured with the grounded representations illustrated in fig. 2.

B. Relations

Visibility of grounded representations is also important in learning relations. For example, fig. 6 illustrates what happens when the combinatorial structures "X gives Mary a book" and "Mary gives X a pen" are encoded in the architecture of fig. 2, and the query "What does Mary own?" has to be answered. In this example, "Mary" is the recipient of the predicate "give(X, Mary, book)", but also the agent of the predicate "give(Mary, X, pen)." The query gives the information that "Mary" is the agent of the predicate "own(Mary, Y)", and it asks for the object/theme Y of this predicate.

The query can be answered because of the relation between the agent (a) of "own" and the recipient (r) of "give" ("own-a" = "give-r"), and the relation between the theme (t) of "own" and the theme of "give" ("own-t" = "give-t"). In fig. 6a it is assumed that the query first activates a neural circuit (representation) for "own-a". This has no direct effect, because "Mary owns Y" is not encoded. But "own-a" can activate a neural circuit (representation) for "give-r". This results in the activation of the word structure "give" and the conditional connections for **recipient**. Because "Mary" is also active (due to the query), the combined activation results in the selection of V_1 over V_2, thus in the selection of the predicate "give(X, Mary, book)" over the predicate "give(Mary, X, pen)."

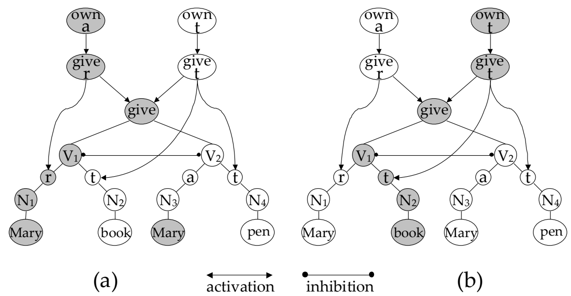

Fig. 6. Reasoning in the neural architecture of grounded cognition (fig. 2). The oval "own a" represents a neural structure for "own-agent". Likewise for "own-theme", "give-recipient", and "give-theme". (a = **agent**, r = **recipient,** t = **theme**). Grey nodes are active.

Fig. 6b illustrates that the query next activates a neural circuit (representation) for "own-t". Due to the relation "own-t" = "give-t", this results in the activation of "give" and the conditional connections for **theme** (all of them, because control circuits have no knowledge of where information is stored in the architecture). Because V_1 has won the competition, this process activates "book" as the (correct) answer to the query.

Relations are also learned over time, often based on different forms of experience. A child could be taught that "John gives Mary a book" entails that "Mary owns the book". But it could also see that when John gives Mary something, she is the owner of that object. To integrate these different types of experience in learning, the representations of the concepts involved have to be visible in combinatorial structures and grounded in perception, as those in fig. 2. In this way, circuits for "own-a" and "give-r" can develop and become integrated.

C. Control of binding

Combinatorial structures with grounded representations are formed by binding the grounded representations temporarily in a connection structure (fig. 4). The way in which a neural circuit (fig. 2) can learn to control the binding process is an important aspect of learning in the architecture of grounded cognition. We investigated this kind of learning by training [19] a feedforward network (FFN) to control the binding of a set of sentence structures.

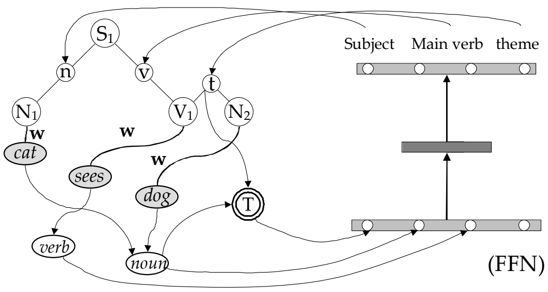

Fig. 7. A feedforward network (FFN) learns to control binding of sentence structures. The ovals and large circles depict the structures as in figs.2 and 4. The double-line circle T is a "conditional node" that anticipates the binding of a theme.

Fig. 7 illustrates the basic aspects of how the FFN was trained (all words are first bound to structure assemblies [7]). The FFN received word type information as input, and it produced binding signals as output. It also received feedback from the developing sentence structure as input. For the sentence "cat sees dog," the (trained) binding process proceeds as follows. The first word "cat" activates an input node of the FFN representing that "cat" is a noun that can have a relative clause. For this input, the FFN must learn to activate a node that produces a "binding signal" that initiates the binding of N_1 and S_1 (activated at the beginning of a sentence [7]) by their noun subassemblies. The binding of N_1 and S_1 represents "cat" as subject of the sentence. In [7] we showed how this binding process proceeds, and how it can be initiated by a "binding signal." The verb "sees" activates a "verb" node in the input layer of the FFN. The output consists of the activation of two nodes. The first one initiates the binding of V_1 and S_1 by their verb subassemblies. The second one activates the theme subassemblies of all V assemblies (by opening the gating circuit for these subassemblies [7]). The effect is that the theme subassembly of V_1 is activated, which produces feedback by activating a "conditional node" T. This nodes anticipates the binding of a noun as theme (object). The word "dog" activates the same input node as "cat," but in conjunction with the active conditional node T it also activates an input node that signals the anticipation of a theme. The (trained) output is the activation of a node that initiates the activation of the theme subassemblies of all N assemblies. The effect is the activation of the theme subassembly of N_2, which results in the binding of V_1 and N_2 by their theme subassemblies [7].

Next, we trained the FFN on a sentences with clauses, like a sentence with a subject-relative clause ("cat that sees dog chases mouse"), and a sentence with an object-relative clause ("cat that dog sees chases mouse"). For these sentences, we introduced new input and output nodes, and new conditional nodes (e.g., anticipating a relative clause).

We tested the ability of the FFN to control binding in a new set of sentence structures, in which the clauses in the training sentences were extended and recombined. Among others, we used a sentence with a multiple embedded subject-relative clause ("cat that sees dog that likes boy chases mouse"). The FFN produced the correct output (binding signals), and the binding process in the architecture proceeds in regular way (i.e., there are no binding conflicts, see below).

We also used a sentence with a multiple object-relative clause ("cat that dog that boy likes sees chases mouse"). The FFN produced the correct binding output for this sentence, on the assumption that the developing sentence structure produced the correct feedback. This feedback, however, is dependent on the binding process itself, and "binding conflicts" arise between the structure assemblies for sentences of this type [7]. Humans indeed have severe problems with sentences of this kind [20].

Nevertheless, the FFN produced the correct binding signals when the binding feedback is correct, that is when binding succeeds without conflict. This would occur in an idealized situation, when a more recently activated subassembly has a higher state of activation than a previously activated subassembly [7]. This perhaps suggests that the overall architecture (structure assemblies and FFN combined) possesses the "competence" to handle (arbitrary) recursive structures of this kind. The performance difficulties then arise from the noisy non-idealized dynamics in the architecture. However, the competence of the system is not embodied in the FFN alone. Instead, the input-output relations learned by the FFN suffice because a substantial part of the sentence structure is embodied in the relations between the structure assemblies. The feedback from the sentence structure is thus an integrated part of its competence, which indicates that competence and performance are integrated in this architecture.

IX. LEARNING AND DEVELOPMENT

The FFN in fig, 7 was trained on basic input-output relations to control the binding process in the architecture. It then dealt successfully with more complex and recombined sentence types. However, the input-output relations learned by the FFN suffice because a substantial part of sentence structure is embodied in the architecture itself. This raises the issue of how the structure of the architecture itself emerges. We argue that this could be the result of a different kind of process that might be referred to as "development" instead of just learning. The FFN we used shows what could be the difference between learning and development. The FFN learns by an adaptation of its weights, but its structure does not change. Perhaps one could introduce a structure change by allowing new connections to be formed in the FFN. But even then, the structure change is not on a par with that needed to develop the structure of the blackboard architecture (i.e., the different assemblies and subassemblies involved, and their corresponding connection matrices [7]).

The success of learning with the FFN perhaps suggest that language learning is the result of such a dual process: an adaptation of control, in combination with a structural development of a binding architecture for grounded representations. This distinction might motivate the search for two different kinds of mechanisms: one needed for the structural development of a language architecture; and one needed for control, based on feedback from this architecture. The first could be referred to as development, because it is primarily related to structural changes that might occur only at an initial stage. The latter could be referred to as learning, based on continuously updated information.

Furthermore, the grounded (word) representations and their associations and relations develop/learn independently from the binding architecture. This shows that the overall architecture scales well under enlargement, and newly learned words can integrate easily in the architecture [7].

X. Conclusion

The importance of the combination of productivity, dynamics and grounding is that they impose constraints on each other. In turn, these constraints can reveal fundamental aspects of human cognition. For example, combinatorial structures can be created with grounded representations, but not all structures are equally feasible [7]. Furthermore, dynamics influences this process, which provides additional constraints on the ability to create combinatorial structures. The examples given here show that these constraints prevent the excessive production of sentence interpretations, as found in systems with unlimited productivity. But, on occasion, it can also result in misrepresentations, as found in human cognition.

Fodor, who has been an ardent supporter of the computational view of cognition, has claimed that a computational (symbol manipulation) account of cognition is nevertheless incomplete [18]. He argued that, because computational processes are always local (as illustrated in fig. 1), they do not capture the global flexibility of cognition (perhaps the most important feature of human cognition [18]). Connectionism cannot fill this gap, in his view, because it lacks the productivity of human cognition.

Grounded cognition, as presented here, is both local and global. Specific processes, such as the creation of sentence structures, are local. But the representations involved are grounded, and grounded representations are global. This combination provides a basis for the productivity and global flexibility of cognition.

References

[1] G. A. Miller, *The psychology of communication*. London: Penguin, 1967.

[2] F. van der Velde, G. T. van der Kleij van der Voort and M de Kamps, M, "Lack of combinatorial productivity in language processing with simple recurrent networks," *Connection Science*, vol. 16, pp. 21-46, 2004.

[3] A. Newell, *Unified Theories of Cognition*. Cambridge, MA: Harvard University Press, 1990.

[4] R. Jackendoff, "Linguistics in cognitive science: The state of the art," *The Linguistic Review*, vol. 24, pp. 347-401, 2007.

[5] G. Vigliocco, et al., "The role of semantics and grammatical class in the neural representation of words," *Cerebral Cortex*, vol. 16, pp. 1790-1796, 2006.

[6] M. Tettamanti et al., "Listening to action-related sentences activates fronto-parietal motor circuits," *Journal of Cognitive Neuroscience*, vol. 17, pp. 273-281, 2005.

[7] F. van der Velde and M de Kamps, "Neural blackboard architectures of combinatorial structures in cognition," *Behavioral and Brain Sciences*, vol. 22, pp. 37-70, 2006.

[8] R. C. O'Reilly, "Biologically based computational models of high-level cognition," *Science,* vol. 314, pp. 91-94, Oct 2006.

[9] J. L. McClelland and T. Rogers, "The parallel distributed processing approach to semantic cognition," *Nature Reviews Neuroscience*, vol. 4, pp. 310-322, Apr. 2003,

[10] J. A. Feldman, *From Molecule to Metaphor*. Cambridge, MA: MIT Press, 2006.

[11] E. K. Miller, "The prefrontal cortex: complex neural properties for complex behavior," *Neuron*, vol. 22, pp. 15-17, Jan. 1999.

[12] P. Smolensky, "Tensor product variable binding and the representation of symbolic structures in connectionist systems," *Artificial Intelligence*, vol. 46, pp. 159-216, 1990.

[13] T. A. Plate, *Holographic Reduced Representation*. Stanford, CA: CSLI Publications, 2003.

[14] S. Pinker, *The Language Instinct*. London: Penguin, 1994.

[15] B. J. Baars, "Conscious cognition and blackboard architectures", *Behavioral and Brain Sciences*, vol. 22, pp. 70-71, 2006.

[16] B. J. Baars and S. Franklin, "How conscious experience and working memory interact," *Trends in Cognitive Science*, vol. 7, pp. 166-172, 2003.

[17] F. van der Velde and M de Kamps, "From knowing what to knowing where: Modeling object-based attention with feedback disinhibition of activation," *Journal of Cognitive Neuroscience*, vol. 13, pp. 479-491, 2001.

[18] J. A. Fodor, *The mind doesn't work that way*. Cambridge, MA: MIT Press, 2000.

[19] K. Plunkett and J. L. Elman, *Exercises in rethinking innateness.* Cambridge, MA: MIT Press, 1997.

[20] E. Gibson, "Linguistic complexity: Locality of syntactic dependencies," *Cognition*, vol. 68, pp. 1-76, 1998.