Chapter 0

Representations of syntax and semantics in a simple recurrent network trained on an idiomatic language

Martin Takáè

Centre for Cognitive Science, Faculty of Mathematics, Physics and Informatics, Comenius University, Bratislava

Slovakia

Alistair Knott and Lubica Benuskova

Department of Computer Science, University of Otago, Dunedin

New Zealand

1. Introduction

While language allows words to be productively combined to convey new meanings, everyday language also contains many **fixed expressions**: sequences or patterns of words which occur together with particularly high frequency. Fixed expressions occupy a continuum, from songs and proverbs at the ‘fixed’ end of the spectrum, through idioms (e.g. *kick the bucket*, *take X to task*) and phrasal verbs (e.g. *come across*, *pull up*) to statistically-defined collocations at the other end (e.g. *good as gold*, *as soon as possible*); for a taxonomy, see Cowie 1998. What they have in common is that they are in some sense ‘regularly occurring’ structures of words: phrases whose component words occur together more frequently than one would expect by chance.

Fixed expressions can have various origins. Some fixed expressions have their origin in the fact that certain *situations* which need linguistic expression occur with particularly high frequency. For instance, the origin of the frequently-occurring expression *How are you?* is likely to be the fact that people frequently ask after one another’s health. Other fixed expressions are patterns of words whose meanings have become conventionalised over time, so that they contribute their meaning collectively rather than through their individual component words. Of course, these two origins are not exclusive; high-frequency messages are good candidates for expression through conventionalised fixed expressions. (E.g. *Howya doin?* conventionally realises the message ‘How are you?’.)

In this chapter, we will be focusing on fixed expressions which convey conventionalised meanings, which we will term **idioms**. We will define an idiom as a fixed expression which displays restricted syntax and whose meaning cannot be derived compositionally from its constituents. Idioms vary in degree of syntactic and lexical frozenness; some of them permit several transformations, e.g. *she kicked the bucket* or *she will kick the bucket*, but not *the bucket was kicked by her*, others are completely fixed. They also vary in the way their meaning is related to meanings of their components. In some cases, the original meaning of an idiom is quite easily recoverable; for instance if it has a metaphorical meaning which is still apparent (e.g. '*meet your maker*'). In other cases, the meaning is not easy to recover (e.g. above-mentioned *kick the bucket*).[1]

Estimates of the proportion of fixed expressions in the vocabulary of language users vary from a conservative 7% Sprenger (2003)[2] to almost half of the lexicon Jackendoff (1995). In either case, these estimates suggest that fixed expressions are nothing exceptional, but rather form a significant part of language. In terms of frequency of usage, fixed expressions which convey their meanings compositionally (e.g. *looking forward to* or *as soon as possible*) in fact outnumber idioms Sprenger (2003). However, our interest will be in fixed expressions which convey a particular semantic concept collectively, rather than compositionally.

Studying idioms is important for several reasons. First, idioms are surface patterns and have to be acquired and represented as wholes. The need to account for idioms places important constraints on the architecture of a syntactic theory. For instance, Jackendoff 2002 argues against generative theories in the Chomskyan tradition on the grounds that they cannot represent idioms. On the other hand, constructivist models of syntax, e.g. Tomasello (2003); **?**, are expressly designed to deal with idioms as well as with productive grammatical rules. For constructivists, fixed expressions are qualitatively similar to regular grammatical rules: they are both learned associations between surface linguistic patterns and semantic patterns, which differ only in the complexity of the associated patterns. Constructivist accounts of language postulate a single unified mechanism for learning both types of structure.

Second, idioms are an ideal testbed for studying the interplay of lexical and syntactic mechanisms in language production. Some leave these mechanisms undistinguished or intertwined; for instance Elman 1991 argues for a distributed representation of words that inherently also carries contextual/syntactic information. Others have postulated special representational units carrying syntactic information specific for particular lexical concepts: **lemmas** for words Levelt et al. (1999) and **superlemmas** for idioms Levelt & Meyer (2000); Sprenger et al. (2006). Others Chang (2002); Chang et al. (2006); Konopka & Bock (2009) accept the need for lexically specific syntactical information, but argue for more general abstract syntactic mechanisms that map event structure to syntactic frames without being necessarily triggered by/dependent on lexical retrieval.

TODO rozsirit cast o comp. modeling a SRN.

[1] Both of these phrases mean *to die*.

[2] results for Dutch

1.1 Computational Modeling

Computational modeling has become a valuable tool in many areas, including linguistics, cognitive science and psychology. A useful property of computational models is the ability to control parameters in simulated experiments and also to inspect internal states and representations that are inaccessible in 'real' humans. In this chapter we will present a computational model of language acquisition with the focus on language production. The model is based on a Simple Recurrent Network (SRN) Elman (1990), which has proven to be a good connectionist architecture for tasks involving learning of temporal dependencies and sequences in general, and for learning the syntactic features of natural language which manifest themselves in sequential patterns Cernansky et al. (2007); Christiansen & Chater (1999); Elman (1991).

As a connectionist model, it learns to produce syntactically correct sentences from the exposure to a subset of a target language – in this case an artificial language .

1.2 Simple Recurrent network

A Simple Recurrent Network (SRN) Elman (1990) has proven to be especially suitable for tasks involving learning of temporal dependencies and sequences and language syntax in particular Cernansky et al. (2007); Christiansen & Chater (1999); Elman (1991).

In this paper we present a SRN-based connectionist model of sentence production, which learns to produce a sequence of words. The model is a regular SRN, augmented with an additional input holding a semantic representation of the episode to be expressed. The semantic representation is tonically active while the sentence is being generated.

An SRN architecture augmented with a tonic semantic input has frequently been used to model the production of individual words, construed as phoneme sequences Dell et al. (1993). In order to model sentence production, the conventional wisdom is that recurrent networks have to learn sequences of elements which are more abstract than words, whether these be semantic roles Chang (2002), word classes Pulvermüller & Knoblauch (2009) or multi-word phrasal units Dominey et al. (2006). While we ultimately agree that knowledge of syntax involves something more than simple word sequencing, the capabilities of a basic word-sequencing SRN augmented with semantics have not yet been very thoroughly studied. Some aspects of linguistic knowledge may in fact be quite well modelled by a simple network of this kind. We want to explore two questions. Firstly, what sorts of implicit representation are learned by a word-sequencing SRN augmented with semantic inputs? In particular, does it learn to separate syntactic and semantic representations? (It certainly should, in order to reproduce the patterns of dissociation found in Broca's and Wernicke's aphasia.) Secondly, how good is such an SRN at learning idiomatic linguistic forms? Recent syntactic theories argue persuasively that syntactic knowledge involves knowledge of idiomatic surface patterns of language as well as more abstract syntactic rules Jackendoff (2002). By recent estimates, idioms make up anything from 7% to almost half of the adult lexicon Jackendoff (1995); Sprenger (2003); for children, the proportion may be even higher Tomasello (2003). A word-sequencing SRN should be good at learning the surface patterns associated with the idioms in a language. But it must be able to learn the meanings of idioms as well. The component words of an idiom map *collectively* to a semantic representation, rather than individually. We want to investigate how well a basic word-sequencing SRN can learn idiomatic patterns of this kind.

To explore these two questions, we have devised and tested a simple model of language production on an artificial language comprised of both idioms and non-idioms. We will describe the model in Section 2 and the language in Section 3. We explore the model's ability to learn to produce syntactically well-formed and semantically correct sentences for given meanings in Section 4. In Section 5 we analyse the model's state space and draw conclusions about how it represents the structure of the language. In Section 6 we explore how these representations change in lesion experiments, and in Section 7 we conclude with a discussion.

2. Model architecture

Language production involves mapping a semantic message to a sequence of words. Our model is a variant of a SRN enhanced with a semantic input (see Figure 1left). The *Semantic input* block encodes the meaning of a transitive sentence in a simple theta-role frame with three roles AGENT, ACTION, PATIENT (abbreviated hereafter as AG, ACT, PAT). Each role is bound to a symbolic token representing a particular meaning; for example the token DADDY represents the meaning of the word *daddy*.[3] Thus the meaning of the sentence *daddy bites bread* is represented as AG: DADDY, ACT: BITE, PAT: BREAD. Roles are coded using binding-by-space Chang (2002); McClelland & Kawamoto (1986), where each role has its own field of units with a dedicated unit for each concept admissible in that particular role (as shown in Figure 1left). Hence for example, there is a unit for a concept DADDY in the agent role, and a separate unit for DADDY in the patient role. (This scheme has well-known problems Chang (2002), but it has the benefit of simplicity, which helps us to clearly frame the questions about idioms and internal representations which are our focus in this paper.) There are 53 semantics units altogether (7 for AGENT, 11 for ACTION, and 35 for PATIENT role). In the *Current word* block, there are 58 localist input units for 57 possible words plus 1 unit coding a sentence boundary (SB). The network has 100 hidden units with sigmoidal activation functions, activities of which are copied back to the context layer (100 units) in the next time step. More hidden units does not improve the performance. The output layer consists of 58 linear units (one for each word plus one for SB).

After training for next-word prediction, the network can run in a *sentence generation mode*. In this mode, we begin by setting SB as the current word and a conventional 'initial context' signal[4] as the current context. Stuff deleted Then at each iteration we feed activity forward to the output layer, stochastically choose an output word, and use the selected word as the current word for the next iteration, and the context layer activity as the current context layer input for the next iteration. The semantic input stays the same and the cycle repeats until the network predicts another SB. At this point, a new semantic input is specified, and a new cycle starts. To prevent generation of unreasonably long sentences, we force SB on the input if it has not been selected as the next word within 20 words from the last SB.

[3] We will write concepts in CAPITALS, and words in *italics*.

[4] A constant vector of values of neuron activities randomly selected between $(-0.5, 0.5)$. This signal is only used once, at the beginning of the sentence generation mode; we do not reset the context layer after each generated sentence, neither during sentence generation mode nor during training.

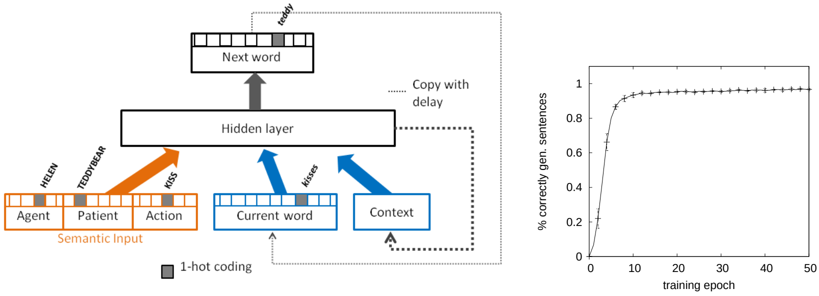

Fig. 1. Left: The SRN architecture enhanced with a semantic input. The thick arrows mean fully connected layers. In language production mode, the meaning (e.g. AG:HELEN, PAT:TEDDYBEAR, ACT:KISS) is delivered to the semantic input and each predicted next word (e.g. *Helen*, *kisses*, *teddy*, *bear*, *SB*) is fed back to the input until the sentence boundary (SB) is predicted. Then a new sentence meaning is delivered in semantic input. Right: Semantics-driven production performance. Results are averaged over 5 model subjects (the graph also shows standard deviations, which are very small).

For stochastic selection of the next word, we make use of the fact that the output units of a SRN trained for the next-word prediction task can be read as holding an estimate of the probability distribution of likely next words, provided they use localist coding Elman (1990). We use a modified softmax selection policy called Boltzmann selection Chambers (1995):

$$P_i = \frac{\exp\left(a_i/T\right)}{\sum_{j=1}^{n} \exp\left(a_j/T\right)} \,,$$

where P_i is a probability of selection of the i-th word, a_i is the activity of the i-th linear output neuron and T is a temperature. The probability of the selection of a neuron is proportional to its activity; the temperature T controls the stochasticity of the selection, from a deterministic choice of the most active neuron ($T = 0$) to a completely random selection ($T \to \infty$). We experimented with several values of T and in the end used $T = \frac{1}{3}$ in our experiments.

3. Artificial language

We investigated the performance of a SRN trained on an artificial language containing a mixture of idioms (fixed expressions) and syntactically regular constructions. Sentences in the language are syntactically homogeneous, in that they are all transitive; they differ only in their degree of idiomaticity.

The core of our 57-word vocabulary consisted of words commonly used by 16-30 month-old toddlers according to the Child Development Inventory (CDI) Fenson et al. (1994). The grammar of our language allowed for regular transitive sentences and also for two types of idioms:

- continuous noun phrase (NP) idioms (*teddy bear*, *Winnie the Pooh*, *ice cream*, *french fries*),
- discontinuous verb phrase (VP) idioms (e.g. *kisses NP good bye*, *gives NP a hug*).

The language also featured semantic dependencies, in that some verbs could only be followed by animate objects and the verb *eats* could only be followed by a noun phrase denoting food. It also contained synonyms (BUNNY: *rabbit / bunny*, TEDDYBEAR: *teddy bear / Winnie the Pooh*, HUG: *gives NP a hug / hugs NP*) and lexical ambiguities (in that words for people could appear in both subject and object positions, the word *gives* could be a part of either *gives NP a hug* or *gives NP five* and the word *kisses* could be either a regular verb as in *grandpa kisses grandma* or a part of an idiom with a different meaning as in *grandpa kisses grandma good bye*). The complete language consisted of 2200 transitive sentences generated from a context-free grammar (see Table 1).

Table 1. Top: Transcription rules for the language used in our simulations. '.' stands for sentence boundary (SB). Bottom: Examples of sentences composed of single words, continuous idioms, and discontinuous idioms.

<table>
<tr><td>TRANSITIVE</td><td>→</td><td>SUBJ VERB_GEN OBJ_GEN . | SUBJ VERB_ANIM OBJ_ANIM . | SUBJ eats FOOD . | SUBJ kisses OBJ_ANIM good bye . | SUBJ gives OBJ_ANIM five . | SUBJ gives OBJ_ANIM a hug .</td></tr>
<tr><td>SUBJ</td><td>→</td><td>HUMAN</td></tr>
<tr><td>VERB_GEN</td><td>→</td><td>sees | loves | holds | bites | washes</td></tr>
<tr><td>VERB_ANIM</td><td>→</td><td>kisses | tickles | hugs</td></tr>
<tr><td>OBJ_GEN</td><td>→</td><td>HUMAN | THING | ANIMAL | FOOD</td></tr>
<tr><td>OBJ_ANIM</td><td>→</td><td>HUMAN | ANIMAL</td></tr>
<tr><td>HUMAN</td><td>→</td><td>mummy | daddy | Samko | Helen | Mia | grandma | grandpa</td></tr>
<tr><td>THING</td><td>→</td><td>ball | book | balloon | toy | doll | block | crayon | pen</td></tr>
<tr><td>ANIMAL</td><td>→</td><td>dog | kitty | duck | bunny | rabbit | cow | pig | bug | puppy | bee | monkey | teddy bear | Winnie the Pooh</td></tr>
<tr><td>FOOD</td><td>→</td><td>cookie | banana | apple | cheese | cracker | ice cream | bread | pizza | french fries</td></tr>
</table>

Mummy eats carrot. Mia loves ice cream. Helen tickles Winnie the Pooh. Grandpa gives grandma a hug. Daddy kisses teddy bear good bye.

Single concepts were always represented by single tokens even if they were expressed by multi-word phrases, for example ICECREAM for *ice cream*. The meaning of a discontinuous verbal idiom was represented by a single token bound to the ACTION role, for example ACT: FAREWELL for *kisses NP good bye* or ACT: HUG for *gives NP a hug* (meaning of the NP was represented by another token bound to the PATIENT role).

4. Training and testing

We created five different instances of the neural architecture with different initial weights, to represent five 'model subjects'. Each model subject was trained on the next-word prediction task (see Table 2), with its own training set (500 randomly selected sentences from the artificial language) and tested with its own test set (1000 unseen sentences from the same language). We used a simple back-propagation training algorithm Rumelhart et al. (1986) with entropy cost function, learning rate 0.1 and no momentum.

In order to evaluate production performance, *meanings* of the 1,000 test sentences (together with SB) were presented to the network with frozen weights one by one after each training epoch. In each time step, the word to produce was again chosen stochastically and then fed back to the current word input. Sentence meaning stayed unchanged until the network predicted SB, when the next sentence meaning was delivered to the semantic input (SB was forced if not predicted within 20 steps). Thus the test was whether SRN can produce a correct sentence for a given novel meaning.

Table 2. Part of the training sequence for the next word prediction with semantics.

Time step	Semantic input	Current word	Target
1	AG: DADDY, ACT: SEE, PAT: TEDDYBEAR	SB	daddy
2	AG: DADDY, ACT: SEE, PAT: TEDDYBEAR	daddy	sees
3	AG: DADDY, ACT: SEE, PAT: TEDDYBEAR	sees	teddy
4	AG: DADDY, ACT: SEE, PAT: TEDDYBEAR	teddy	bear
5	AG: DADDY, ACT: SEE, PAT: TEDDYBEAR	bear	SB
6	AG: GRANDMA, ACT: HUG, PAT: MIA	SB	grandma
etc. . .			

4.0.0.1 Results

Sentence generation in our network was driven by the message on the semantic input. A generated sentence was considered correct if it was correct both syntactically (i.e., it belonged to the language) *and* semantically (i.e., concepts in all three roles of the given message were expressed by appropriate words/phrases in appropriate syntactic positions). For some meanings, multiple correct sentences were possible due to synonymy. As we can see in Figure 1right, the network quite quickly achieved a high level of production performance. Out of 1,000 sentences generated after 50 training epochs, 96.7% were correct (SD=0.4). The generated sentences contained on average 206.4 idioms (SD=11.6), out of which 93.7% were correct (SD=1.7). Here are examples of sentences produced for given messages:

AG: GRANDPA, ACT: BITE, PAT: CRAYON → *Grandpa bites crayon.*
AG: HELEN, ACT: HUG, PAT: BEAR → *Helen gives Winnie the Pooh a hug.*
AG: GRANDMA, ACT: FAREWELL, PAT: RABBIT → *Grandma kisses rabbit good bye.*
AG: GRANDMA, ACT: HUG, PAT: BEAR → **Grandma hugs Winnie the Pooh a hug.*

We can conclude that the SRN has learned quite well to generate sentences which conform both syntactically and semantically to the rules of the training language.

The task performed by the network can be analysed as having several different components. The task of ‘completing a continuous idiom’ can be seen as a purely syntactic one. Once we have started generating an idiom for a given meaning, subsequent words neither add any new semantic content to the idiom’s whole meaning, nor require additional semantic information, so a regular Elman network with no semantic input should be able to accomplish this task comfortably. The task of predicting words in a regular transitive sentence requires a semantic input in each step to disambiguate next-word prediction. At each time step, the updating context units select words which are syntactically appropriate in the current context, while the semantic input units provide a tonic bias towards words which are semantically appropriate. The task of generating a discontinuous VP idiom appears the most difficult: the network must start generating a VP idiom, then put this process ‘on hold’ while generating an intervening

NP, before finally resuming the 'syntactic' task of idiom completion. (Note that generating the intervening NP itself requires a mixture of semantic and context information, and may even contain a nested idiomatic structure of its own.) However, these apparently heterogeneous tasks are all readily learned by a fairly basic SRN architecture.

5. State space analysis

The state space of an Elman network is a N-dimensional space, where N is the number of neurons in the hidden layer. Each *state* – a particular configuration of activities of the hidden-layer neurons is a point in this space and a temporal sequence of states can be thought of as a trajectory in the hidden space Elman (1991). Because of its recurrent connections, the hidden layer has a special function of encoding previous history in the dynamics of the network Tino et al. (2004).

If the activation functions of the output layer are linear (as is the case of our model), each output neuron computes a scalar product of its weight vector with the hidden layer activity configuration, hence effectively responds to those configurations that are most similar (have the least angle) with its weight vector. If each output neuron represents one of the possible next words in the dictionary, it is desirable that the sequences (partial sentences) that can be completed with a particular word be represented by similar hidden layer configurations. The difference between an untrained and a trained SRN is that, during training, the hidden space reorganizes so that categories of sequences that should yield a similar outputs are clustered together Cernansky et al. (2007).

In order to analyze the internal organization of the hidden space, we froze the weights of a selected model individual after 50 epochs of training. The complete training set was swept through the selected network word by word, and the hidden layer activity patterns were recorded. We plotted the network's hidden layer representations of words in 2-dimensional projections of the hidden space created by Principal Component Analysis (PCA), using a colour scheme to identify words with particular grammatical roles and different positions within idioms. Different colours were used for SBs, for all words in subject position, whether idiomatic or not, and for all verbs. Different colours were also used for the 1st, 2nd and 3rd words of idiomatic objects, and for the 2nd and 3rd words of discontinuous verbal idioms. The results are shown in Figure 2 (left). We can see that the first two principal components clearly distinguish these different structural positions. Higher components mostly reflect semantic information. For instance, Figure 2 (right) plots word representations using a different scheme, where different colours are used to represent different complete semantic messages. In this scheme, all the words in a sentence are plotted in the same colour; words from sentences with different meanings appear in different parts of the hidden space. The figure shows results from several sentences, some of which report the same episode in different ways (e.g. *Mummy hugs rabbit*, *Mummy hugs bunny*, *Mummy gives rabbit a hug* and *Mummy gives bunny a hug*). Although the words making up these sentences appear in various different structural positions, and some of them are parts of idioms while others are not, we can see that their activities cluster quite nicely in the plane of the principal components shown here.

The PCA analysis just discussed led us to the hypothesis that different subspaces of the hidden space encode different types of information during sentence generation. Sentences with similar structures will follow similar trajectories when projected to a subspace of components

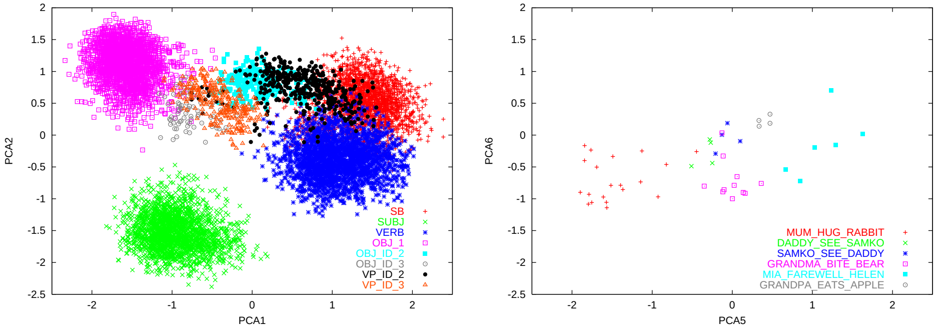

Fig. 2. Left: Activity patterns elicited by all words in the training sequence (*not* averaged across different contexts) shown in PCA1-2 plane. Words with identical syntactic positions in sentences share the same color codes. Syntactic clustering is clearly visible. Right: Activity patterns of all words generated for selected sentence meanings shown in PCA5-6 plane. All words of all synonymic variants of a sentence share the same color codes regardless of their syntactic position. Semantic clustering (by sentence meanings) is visible.

that encode mostly structural information; sentences with similar semantics (i.e. sharing some semantic roles) will have similar trajectories when projected to components that encode semantic information. Recall that we can think of the network's computation as a process in which each output neuron representing a particular candidate next word compares the angle of its weight vector with the hidden-layer activity pattern vector. The activity pattern vector is equal to a linear combination of its principal components (with earlier principal components having the most influence)—which means that syntactic and semantic aspects jointly contribute to the decision. As we will show in the next section, representation of semantic and structural aspects can be selectively damaged by lesioning different parts of the network.

6. Effects of lesions on hidden layer representations

We will denote the connections from the semantic input to the hidden layer as *semantic connections* and those from current word and context units to the hidden layer as *syntactic connections*. To study effects of lesions, we took the trained model individual described in Section 5 and selectively lesioned (zeroed) a certain ratio of (randomly chosen) semantic or syntactic connections. Then we tested the performance of the network on the sentence generation task, and also examined how its hidden layer representations changed in response to the lesions.

6.0.0.2 Performance of the lesioned network

The SRN with seriously lesioned syntactic connections was unable to produce syntactically correct sentences. It often produced or repeated just one or two words. However, the produced words were nearly always semantically appropriate. Conversely, when we progressively damaged semantic connections, the network maintained a high degree of syntactic correctness, but at the same time the sentences became less and less semantically relevant to the meanings that should have been expressed (e.g. the network might generate *Mummy kisses puppy.* for AG: MUMMY, ACT: EAT, PAT: APPLE).

6.0.0.3 Hidden space organization in lesioned networks

In Section 5 we saw that syntactic and semantic information is visible in different principal components of the space of hidden layer neuron activations. Recall that Figure 2 shows clustering of syntactic categories in PCA1-2 (left) and semantic categories in PCA5-6 (right) subspace. By way of comparison, Figures 3 and 4 show activation patterns for exactly the same stimuli and with the same color coding, but for networks obtained from the original one by lesioning 100% of semantic connections (left) and syntactic connections (right).

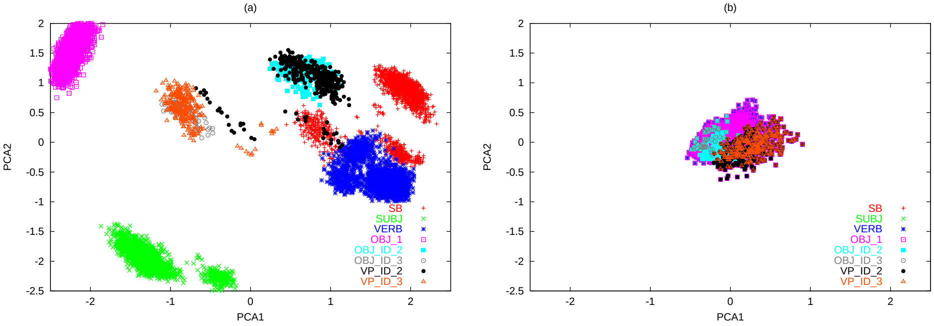

Fig. 3. Activity patterns of words in the training sequence color-coded by syntactic roles in the network with 100% semantic (left) and syntactic (right) connections lost shown in the PCA1-2 plane. Clusters are preserved in the former case and destroyed in the latter (compare Figure 2 left).

As we can see, clustering of syntactic categories is well preserved if we only lesion *semantic* connections (Fig. 3a). The loss of semantic information causes the clusters for SB and SUBJ to be divided into smaller subclusters that represent the influence of the previous sentence. (This influence would be eliminated in unlesioned network by arrival of a new meaning on the semantic input.) The clusters are now smaller in size, but they generally remain in areas within the original clusters of the unlesioned network when projected to principal components that encode mostly syntactic information (refer back to Figure 2 left). If, on the other hand, we lesion syntactic connections, clusters of syntactic categories collapse and are intermingled (Fig. 3b).

Now we consider the effects of lesions to hidden space organization along principal components that mostly encode semantic information. If we lesion semantic connections, clustering of semantic categories is destroyed (Fig. 4). If we lesion 100% of syntactic connections, original clusters reduce to points for the first produced word (as without a context signal, the SRN cannot get past producing the first word of a sentence). However, these 1-point activation patterns across synonymous sentences that express the same meaning (e.g. *Mummy hugs rabbit*, *mummy hugs bunny*) are highly consistent.

Taken together, these results suggest that the SRN has developed largely independent representations of syntactic and semantic knowledge in its state space, which can be selectively damaged by lesioning connections in the respective parts of the network.

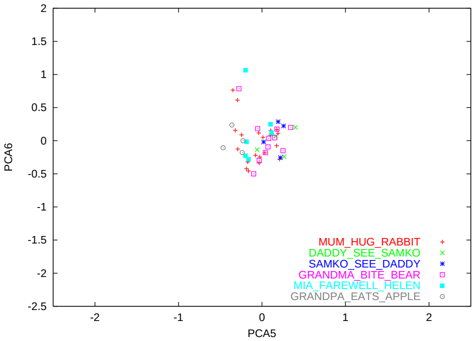

Fig. 4. Activity patterns of words for selected meanings color-coded by sentence meanings in the network with 100% semantic connections lost shown in the PCA5-6 plane. Clusters are destroyed (compare Figure 2 right).

7. Discussion

We have shown that an SRN enhanced with a tonically active semantic input layer can produce syntactically well-formed and semantically correct sentences for given meanings, reflecting a simple grammar. The network was able to learn to produce both idioms and regular sentences. Detailed analysis of the representational space of the network's hidden layer showed that different subspaces encoded different aspects of the sentence generation process: some encoded structural information (regardless of content), others encoded content (regardless of structure). Because a hidden layer activity pattern can be expressed as a linear combination of its components, syntactic and semantic aspects encoded in different subspaces jointly influence the generation process.

Our lesion study showed that by lesioning different parts of the network, we could reproduce some symptoms of patients with Broca's aphasia (grammatical errors, when lesioning connections of the basic sequencing SRN) and Wernicke's aphasia (semantic errors, when lesioning connections from the semantic input layer to the hidden layer of the sequencing SRN). Analysis of the hidden layer showed that these lesions indeed had effects in the corresponding subspaces representing content and sentence structure. The network learned to distinguish the separate contributions of word-sequencing and semantics to the task of sentence generation, and to represent them separately in the hidden layer.

As stated at the outset, we certainly do not want to suggest that a simple word-sequencing SRN augmented with static representations of episodes will suffice as a model of human sentence production. But it is useful to explore thoroughly what a network of this kind *can* do: that is our main goal in the current paper. One interesting finding has been that a word-sequencing network is good at learning idiomatic structures in language. Idioms are constructions which make reference to the surface structure of language as well as to its syntactic structure Jackendoff (2002). A word-sequencing network is ideally suited for learning the surface components of idiomatic constructions. Indeed, almost by definition, it is better suited for this task than a network specially configured to learn patterns that abstract away from surface structure. At the same time, a simple word-sequencing SRN learns reasonable representations of the grammatical contexts where idioms can appear, of discontinuous idioms, and even of nested idioms. In addition, the network also learns a good model of the *semantics* of idioms. As we showed in Sections 5 and 6, the network learns semantic representations which are largely separable from its representations of sequential structures in language. These representations are very well suited to serve as the meanings of idiomatic word sequences. During generation of a word sequence, they provide a tonic bias towards one idiom rather than another, while the network's recurrent representations control how the words within an idiom are sequenced. The representations learned by the network combine to implement something rather like a 'superlemma'—the structure envisaged by Levelt and colleagues to be responsible for the storage of idioms in the mental lexicon Levelt & Meyer (2000); Sprenger et al. (2006). A superlemma is a unit of lexical organisation which maps a single semantic concept onto a sequence of individual word lemmas. In Levelt *et al.*'s treatment, superlemmas are localised units, which have connections to concepts on the one hand, and to particular word lemmas on the other. In our scheme, on the other hand, superlemmas are distributed entities, stored in the learned synaptic weights which organise the network's hidden layer representations. The learned weights achieve an effect similar to that brought about by the activation of a superlemma: when the meaning of a given idiom becomes active in the semantic input layer, the network is biased to generate the component words of the idiom, and separately biased to produce them in the right order.

In summary, we envisage that a simple word-sequencing SRN augmented with semantic inputs might have a useful role to play in a complete model of sentence production. Given that most current models assume some element of recurrency in the network which generates sentences, the word-sequencing network provides a parsimonious account of the mechanism which forces generated sentences to conform to constraints defined in the surface structure of language—i.e. which produces idioms.

Of course, we must still address the question of how the word-sequencing network is integrated with a network implementing more abstract syntactic constraints on the structure of generated sentences. This is a matter of ongoing research. Our current suggestion about the nature of abstract syntactic constraints [reference witheld] starts from the proposal that the semantic representation forming the input to the sentence generation process is not presented tonically to the network, but rather in the form of a sequence with its own internal structure, ultimately reflecting sequential constraints in the sensorimotor processes through which episodes in the world are apprehended. In this scheme, the abstract syntactic structure of a sentence is due, at least in part, to the sequential structure of the underlying input message. We have already shown that that a network taking semantic input in the form of a sequence can learn very abstract syntactic generalisations, making no reference at all to particular words or surface forms [reference witheld]. It remains to be seen whether this network can be combined with the word-sequencing network described in the current paper, to implement both grammatical and surface structure constraints.

8. References

Cernansky, M., Makula, M. & Benuskova, L. (2007). Organization of the state space of a simple recurrent neural network before and after training on recursive linguistic structures, *Neural Networks* **20**(2): 236–244.

Chambers, L. (1995). *Practical Handbook of Genetic Algorithms*, Vol. 2, CRC-Press, Boca Raton, FL.

Chang, F. (2002). Symbolically speaking: a connectionist model of sentence production, *Cognitive Science* **26**(5): 609–651.

Chang, F., Dell, G. S. & Bock, K. (2006). Becoming syntactic, *Psychological Review* **113**: 234–272.

Christiansen, M. H. & Chater, N. (1999). Toward a connectionist model of recursion in human linguistic performance, *Cognitive Science* **23**(2): 157–205.

Cowie, A. (ed.) (1998). *Phraseology: Theory, analysis and applications*, Clarendon Press, Oxford, UK.

Dell, G. S., Juliano, C. & Govindjee, A. (1993). Structure and content in language production: A theory of frame constraints in phonological speech errors, *Cognitive Science* **17**(2): 149–195.

Dominey, P., Hoen, M. & Inui, T. (2006). A neurolinguistic model of grammatical construction processing, *Journal of Cognitive Neuroscience* **18**(12): 2088–2107.

Elman, J. L. (1990). Finding structure in time, *Cognitive Science* **14**(2): 179–211.

Elman, J. L. (1991). Distributed representations, simple recurrent networks, and grammatical structure, *Machine Learning* **7**: 195–224.

Fenson, L., Dale, P., Reznick, J. S., Thal, D., Bates, E., Hartung, J., Pethick, S. & Reilly, J. (1994). Variability in early communicative development, *Monographs of the Society for Research in Child Development* **59**(5): i–185.

Goldberg, A. E. (1995). *Constructions: A construction grammar approach to argument structure*, University of Chicago Press, Chicago.

Jackendoff, R. (1995). *The boundaries of the lexicon*, Lawrence Erlbaum, Hillsdale, NJ, pp. 133Ű–166.

Jackendoff, R. (2002). *Foundations of Language: Brain, Meaning, Grammar, Evolution*, Oxford University Press, Oxford, UK.

Konopka, A. & Bock, K. (2009). Lexical or syntactic control of sentence formulation? Structural generalizations from idiom production, *Cognitive Psychology* **58**(1): 68–101.

Levelt, W. J. M. & Meyer, A. S. (2000). Word for word: Multiple lexical access in speech production, *European Journal of Cognitive Psychology* **12**(4): 433–452.

Levelt, W. J. M., Roelofs, A. & Meyer, A. S. (1999). A theory of lexical access in speech production, *Behavioral and Brain Research* **22**(1): 1–75.

McClelland, J. L. & Kawamoto, A. H. (1986). Mechanisms of sentence processing: assigning roles to constituents, *in* J. L. McClelland & D. E. Rumelhart (eds), *Parallel distributed processing: explorations in the microstructure of cognition*, Vol. 2: Psychological and biological models, MIT Press, Cambridge, MA, pp. 272–325.

Pulvermüller, F. & Knoblauch, A. (2009). Discrete combinatorial circuits emerging in neural networks: A mechanism for rules of grammar in the human brain, *Neural networks* **22**(2): 161–172.

Rumelhart, D. E., Hinton, G. E. & Williams, R. J. (1986). Learning internal representations by error propagation, *in* D. E. Rumelhart & J. L. McClelland (eds), *Parallel distributed processing: explorations in the microstructure of cognition*, Vol. 1: Foundations, MIT Press, Cambridge, MA, USA, pp. 318–362.

Sprenger, S. A. (2003). *Fixed expressions and the production of idioms*, PhD thesis, University of Nijmegen, Netherlands.

Sprenger, S. A., Levelt, W. J. M. & Kempen, G. (2006). Lexical access during the production of idiomatic phrases, *Journal of Memory and Language* **54**: 161–184.

Tino, P., Cernansky, M. & Benuskova, L. (2004). Markovian architectural bias of recurrent neural networks, *IEEE Transactions on Neural Networks* **15**(1): 6–15.

Tomasello, M. (2003). *Constructing a Language: A Usage-Based Theory of Language Acquisition*, Harvard University Press, Cambridge, MA.