New Ideas in Psychology 50 (2018) 61–72

Contents lists available at ScienceDirect

New Ideas in Psychology

journal homepage: www.elsevier.com/locate/newideapsych

"How do humans make sense?" multiscale dynamics and emergent meaning

Rick Dale [a, *], Christopher T. Kello [b]

[a] Department of Communication, University of California, Los Angeles, USA
[b] Cognitive & Information Sciences, University of California, Merced, USA

ARTICLE INFO

Article history:
Received 28 January 2017
Received in revised form
5 September 2017
Accepted 16 September 2017
Available online 9 October 2017

ABSTRACT

The challenges posed by the composite nature of sense-making encourage us to study how that composite is dynamically assembled. In this paper, we consider the computational underpinnings that drive the composite nature of interaction. We look to the dynamic properties of recurrent neural networks. What kind of dynamic system inherently integrates multiple signals across different levels and modalities? We argue below that three fundamental properties are needed: dynamic memory, timescale integration, and multimodal integration. We argue that a growing area of investigation in neural networks, reservoir computing, has all these properties (Jaeger, 2001). A simple version of this model is then created to demonstrate "emergent meaning," using a simplified model communication system.

1. Introduction

In a classic paper of cognitive science, Herbert Simon (1992) asks us to imagine a situation in which people are having a conversation: three women in a Singapore cafe, speaking casually in Tamil. One of them has the floor, and is vibrantly sharing details about her prior day. She gestures actively, shifts social gaze, and modulates her vocal system to render sounds. Simon asks us to consider what it would be to "explain" this scenario. There are many ways to answer this question. One way is to explain how linguistic behavior is operating in this context. In this language-centered approach, a critical piece of an explanation is the "meaning" conveyed between interlocutors. The speaker and listeners understand each other, somehow. If the conversation is going right, then the speaker is "making sense."

"Sense-making" is often used as a technical term by some theorists of language, communication, and social systems (e.g., Di Paolo, 2005). It has been used to describe the complex and dynamic process of communicating in context (De Jaegher & Di Paolo, 2007; Tylén & Allen, 2009). This is the sense of "sense" that we mean in the title of this paper. Sense-making is a *process*, a rich kind of meaning that humans create together. *Beyond* words, *beyond* sentences, there's a complex co-creation taking place when two people talk. Humans speak or gesture (or both) to gain attention, identify joint goals, develop a plan, issue instructions, inform one another, and more.

This is a process of great complexity, because language performance unfolds on many different levels. Consider natural spoken language, as in Simon's imagined vignette. When talking, humans linearize sounds into patterns, reflecting what cognitive scientists approximately describe as "words" and "sentences" and "topics of conversation." At the same time, humans often utilize an array of non-verbal tools. As sounds are assembled, gestures render emphasis or structure conceptualization. Humans use social gaze to maintain attention of their conversation partner, or to track it in case something goes awry. The sounds themselves are subject to modulation. Loudness, frequency, and speech rate may all be modulated on any word or phrase to change emphasis, or win back some attention.

By Enfield's (2013) description, language performances are radically composite:

> *"A typically multimodal, multidimensional utterance will consist of numerous signs in a unified composite*, e.g., *words and morphemes, some morphosyntactic arrangement of these, some configuration of the hand, some movement of the arm in a certain direction and at a certain speed, some deployment of the artifactual environment, and much more besides."* (Enfield, 2013, p. 65)

* Corresponding author. Department of Communication, Rolfe Hall, University of California, Los Angeles, USA.
E-mail address: rdale@ucla.edu (R. Dale).
URL: http://rdale.bol.ucla.edu

https://doi.org/10.1016/j.newideapsych.2017.09.002
0732-118X/

This compositeness is sometimes, with justification, ignored. For example, in one influential tradition in the language sciences, it is a methodological convenience to investigate these levels separately. This tradition, at least in some sectors of cognitive science, remains the dominant approach. It inspired a whole generation of cognitive scientists who assume these processes are independent and strictly encapsulated — traditional "modules." Such a simplifying assumption licenses a useful but narrow focus. This focus has no doubt revealed important properties of many processes.

Yet the signals that make meaning "composite" occur in rapid sequences, and often simultaneously, during language performance. To put it simply: Cognitive scientists may have the luxury of modularity, but the human performer does not. In matters of milliseconds, the brain-body-environment system must work with a wide array of signals. These signals are weaved together into one coherent performance. If they are not weaved quickly and coherently, sense-making fails, or leads to unintended consequences. So how do humans do it? How do humans make sense?

The challenges posed by the composite nature of sense-making encourage us to study how that composite is dynamically assembled. In this paper, we consider the computational underpinnings that drive the composite nature of interaction. We look to the dynamic properties of recurrent neural networks. What kind of dynamic system inherently integrates multiple signals across different levels and modalities? We argue below that three fundamental properties are needed: dynamic memory, timescale integration, and multimodal integration (Dale, Kello, & Schoenemann, 2016). We argue that a growing area of investigation in neural networks, reservoir computing, has all these properties (Jaeger, 2001, p. 13). We showcase a simple reservoir computing model to demonstrate "emergent meaning," using a highly simplified "toy" communication system. It reveals the emergence of multi-level statistics. The model is a kind of existence proof, showing that even simple systems that have the right kind of multiscale dynamics can accommodate a simplified form of multilevel structuring seen in language.

In the next section, the concept of multiscale and multimodal dynamics is summarized in more detail. Here we develop key requirements that, we argue, are needed for a cognitive system to sustain a complex performance such as sense-making. Following this, we introduce reservoir computing and describe a simplified simulation that illustrates multiscale dynamics.

2. Cognitive desiderata: multiscale, multimodal dynamics

Let us revisit the hypothetical scenario from the Introduction. Herbert Simon offered a hypothetical cognitive analysis of his imagined human interaction (Simon, 1992), in which one person is speaking Tamil vibrantly to others. Simon asks what it is to "explain her behavior." There are many ways to answer this question. One could ask, as Simon does, why Tamil and not some other language? Why in that cafe and not in some other cafe or even another city or country? This more distal explanatory question may be the subject of anthropology and cultural history, or simply the series of coincident events unique to this person (the so-called "social band" timescale: Newell, 1992).

A cognitive scientist may seek an answer from studying how she is speaking. What gestures are useful for her to convey meaning? How does she organize her eye movements? What words is she choosing and what emotional tone is she conveying? This more intermediate timescale is based on principles of cognitive mechanism and process (Newell's "cognitive band").

There is of course a timescale finer than this. Imagine tracking her brain activity during the conversation. Using functional connectivity analysis, one could investigate the neural circuits that seem to be involved in conveying a word, or a phoneme. The scientist could investigate how this circuit is modulated by the context around her, such as other words or the behaviors of her interlocutors. Here is a process at the scale of milliseconds (Newell's "neural band").

All of these are possible routes to an "explanation." They also exemplify the problem of multiscale dynamics — the speaker must organize her behavior at different timescales in rather systematic ways in order to succeed. Sounds must weave into words which must weave into sentences which, of course, are presented in ways more or less culturally and contextually licensed.

Timescales are not the only way in which language is complex, however. In the face-to-face context, humans coordinate a whole suite of multimodal signals to support meaning (e.g., Louwerse, Dale, Bard, & Jeuniaux, 2012). There are many areas in the language sciences that emphasize this multimodal structure. Such a review is outside the scope of the present summary. It is, for the present purposes, a granted premise that natural language involves such a suite of signals. Why would humans use multimodal signals? Imagine again this conversation in Singapore that Simon envisions. The speakers are gesticulating to highlight certain conceptual structures, sequencing social gaze in a manner meant to maintain and track attention, modulating voice in ways to convey affect or emphasis, and so on — and in fact she may be *combining* them in coherent ways to lend greatest emphasis.

So Simon's hypothetical speaker can be explained at different timescales, and through a variety of seemingly different — but combined — processes. The first property reflects *multiscale* organization, and the second *multimodal* organization. What kind of cognitive system supports multiscale and multimodal dynamics?

From classic approaches to meaning in language, we might simplify the problem and begin our analysis at the word. The simplification is valuable. The focus on word meaning reflects a fundamental importance of the lexical level of analysis. Building discourse systems for combining these words has also led to much progress. This has been especially useful in building systems for multimodal discourse generation (e.g., Kopp et al., 2006) and other automated interactive systems (e.g., Graesser, 2011). But to gain a more complete explanation of Simon's anecdote, given the sheer complexity that sense-making involves, we need more.

Consider, for example, processes that even extend beyond the individual cognitive agents themselves: the environment in which they are communicating. The environment can support events that take place at a longer timescale. Simon's conversational scenario is illustrative once again. The very fact that the same people will be sitting across from our imagined speaker over many minutes means that the environment can support memory for the social and perhaps even topical landscape of the conversation (Spivey, Richardson, & Fitneva, 2004).

In this paper, we choose to focus on the relevant "internal" cognitive mechanisms: We consider neural computation that supports multiscale and multimodal processing. Cognitive science has long involved a strong influence of so-called "internalism," the view that the locus of explanation for cognition is inside the individual. Despite the continued debate about internal vs. external features to explain the mind, we take for granted here that exploring dynamics of a model system provokes useful questions. Whatever one's perspective on this ongoing theoretical debate, the human brain is certainly a critical component in this multiscale,

multimodal system.[1]

So what are the needs of cognitive dynamics in supporting sense-making? We present three desiderata here. We do not claim that these are fully sufficient for linguistic meaning, but they do seem to be, at least, necessary conditions: (i) dynamic memory, (ii) timescale integration, and (iii) multimodal integration.

2.1. Cognitive desideratum: dynamic memory

A "sense-making" system fundamentally involves activities and processes that unfold over time in an organized fashion (De Jaegher & Di Paolo, 2007). In addition, a sense-making system needs to *sustain* its processes over various timescales. Research on language and memory finds that we can tax our memories when related words are spanned too distantly within sentences (e.g., Gibson, 1998). Memory is taxed when we consider anaphora, in which subtle cues such as "he" or "she" or "they" mark a prior mention that can span several sentences. But overlaid upon this is a sustenance of topic that must work to have coherent sense-making. Humans preserve a sense of a goal of their conversation, and maintain a topic across an entire exchange, which could last many minutes. A dynamic memory must be able to rapidly introduce structure, hold it in abeyance, and potentially return to it, over the course of seconds and minutes and perhaps even longer.

2.2. Cognitive desideratum: timescale integration

Dynamic memory is not itself enough. The processes supporting that memory need to be *temporally integrative* across a range of timescales relevant to language and communication. Words, when chosen, fit inside some broader topic of conversation. Sounds must, of course, be initiated and situated within a particular context and performance envelope to work. Sentences must be dynamically introduced that recognize prior mentions, and introduce new information to keep interaction flowing. The most cognitively expensive way of doing this integration is to compute each step independently, and then put them together afterwards, keeping a track to make sure errors are not made. Slightly less computationally costly is to build up from sound to meaning, in a serial order of construction. An even more efficient means of doing all this is to *constitutively integrate* — the processes are not computed independently, but interdependently. We give a further example of what we mean by this below, but the general problem is this: Processes at varied timescales must make sense relative to one another, they must rapidly (and sometimes simultaneously) obey structured and potentially predictive relations to sustain natural language usage (Pickering & Garrod, 2013).

2.3. Cognitive desideratum: multimodal integration

Finally, integration must also occur across "space" as well as time. In terms of face-to-face interaction, the elements in space concern the observed behaviors that are integrated into the performance itself: gestures, eye gaze, and so on. These disparate signals serve valuable functions in sense-making. A subtle gaze in one direction, rather than another, can mark blame or concern or some other conversationally significant intention (cf. Langton, Watt, & Bruce, 2000). This multimodality is central to our primary mode of making meaning in face-to-face contexts. When it is dampened it can seem odd or distracting (Kuhlen & Brennan, 2010). Getting it right matters. This is more than saying simply that conversation is embodied. The term "modality" can stand in for a variety of processes, beyond the standard sensors and effectors that the term "multimodal" connotes. Affective charge of a sentence can be considered a kind of modality, which could be resolved into two separate dimensions itself, such as valence and intensity. In short, the system must be capable of integrating diverse signals across and within timescales.

In the next section, we describe a computing system that can satisfy these desiderata. Reservoir computing emerged in the early 2000's, and is an already influential paradigm for modeling dynamic systems of various sorts. The construction of a reservoir computing system can be quite simple, and yet yield results that inform complex phenomena like sense-making.

3. A primer on reservoir computing

Reservoir computing is a style of neural network modeling. These models can be surprisingly effective as learning devices, even with few theoretical commitments on the structure of the nervous system itself or how it learns. They require some basic assumptions of nonlinear dynamics in a "reservoir" of neurons, and a statistical method for extracting memory and computation from those neural dynamics. The reservoir is, from a certain vantage point, simply a random assemblage of neurons that activate and deactivate as inputs impinge upon the network over time. The reservoir is "recurrent" because its neurons are connected to each other, and this recurrent connectivity can support a self-sustaining dynamic of activity within the reservoir.

Reservoir systems of the kind we introduce here have two properties that distinguish them from more traditional neural network modeling. One is that the connections among neurons in a reservoir are not adjusted by any kind of learning mechanism. The connections are typically randomized at the start of a simulation, and activation of their neurons flows across the system. The second property derives from how those initial random connections are made. Under particular rules for the random initialization, the reservoir networks have an "echo state" property, in that activation flows over the network continually even when it is not "perturbed" by external input (see the Appendix).

The computational implications of a reservoir's connectivity are critical here. The structure provided by even random interconnectivity and the way activation flows over neurons has a time signature that can be used for computation. These architectures can solve some classic problems in cognitive modeling, such as the XOR problem (Plunkett & Elman, 1997), which we describe further below. They can also solve more applied problems such as aspects of speech recognition and locomotion (Caluwaerts, D'Haene, Verstraeten, & Schrauwen, 2013). Reservoir computing also raises interesting questions about the emergence of cognitive processes, including crucial processes like linguistic recursion (Frank, 2006), which we consider further below, and which has figured centrally in linguistic theory.

There are two general classes of reservoir computing system, with subtly distinct histories (see Jaeger, 2007). Whether a network is in one class or another depends on how its model neurons are programmed. This is beyond the scope of the present discussion, but we include an Appendix that describes these two classes, and offers more details about the version we develop here. For the

[1] Despite strong claims to the contrary (e.g., Noë, 2009), there are many reasons for exploring internal processes, even when admitting the structuring role of body and environment in the previous paragraph. Injury to the nervous system can systematically alter cognitive processing, and conversation clearly illustrates this. It is known that certain brain damage can result in a loss of capacity to track the conversation and even the prior identity of a conversation partner when, for example, a conversation partner in Simon's scenario steps away for even less than a minute (Squire & Wixted, 2011). This is not to say that these brain areas are the "seat" of such processes; but they are obviously critically involved in them (cf. Yoshimi, 2012).

present primer, we discuss the essential ingredients of reservoir computing. To begin, let's consider the way a reservoir system is constructed, and how it handles a classic assay of neural network performance, the XOR problem.

3.1. Structure of a reservoir system

Like all neural network models, a reservoir computing system takes a pattern of activation as its input, and transforms it to produce output — another pattern of activation over simple simulated neurons. In reservoir computing, this transformation has two steps. First, the input pattern of activation is fed into the "reservoir," a large network of randomly connected neurons that are iteratively influencing each other, time and time again. The reservoir is allowed to "run," and in many cases can be said to have its own intrinsic dynamics: Activation flows across the network continually, even when it is not perturbed by input. The reservoir is thus intrinsically recurrent. The interesting bit is what happens when the input is presented to the network — the reservoir is influenced, and its dynamics change.

The reservoir's neurons are connected to a set of output nodes, or a "readout." These connections, from reservoir to output, are the only ones modified during learning. The reservoir's internal connections to itself, in most studies, are not trained. They are initialized once and remain unchanged throughout a given simulation. The random structure, and the flow of activation, is enough to instill a generic memory in the reservoir dynamics that can be used for learning at the output — in other words, reservoir dynamics carry their history forward in time. This memory permits the network to learn about patterns sequentially presented in the input, without any special assumptions about the internal structure of the system at all. An illustrative reservoir system is portrayed in Fig. 1, left.

3.2. Dynamics of a reservoir system

To understand how random connections can facilitate processing of information, consider the case of a simplified reservoir system for computing the XOR logical operator, shown in Fig. 1, right. This is a classic task that revealed the limitations of certain classes of neural networks (Plunkett & Elman, 1997). XOR or "exclusive or" is simply a function that takes two propositions, and returns true if only one proposition at the input is true (not both, or neither). The function is used as an assay of neural network learning because it requires a nonlinear division of the input space.

The full account of training this reservoir system is described in the Appendix, and is based on a thorough introduction to the modeling framework found in Lukoševičius (2012). To train the network, we first present an input pattern, such as proposition 1 as true and proposition 2 as false. The XOR operator over these two truth values should produce 1, because one (but not both) of these propositions is true. After the input is presented to the reservoir, we allow its dynamics to cycle, processing that perturbation (in Fig. 1, right, we show this occurring $k = 20$ times). The random connectivity of the reservoir leads to distinct signatures, produced by this input perturbation. These distinct signatures can then be correlated with the desired output, using linear regression. After presenting all such patterns to the network, many times, that correlation technique results in a trained network that can now solve the problem — using only random interconnectivity in the reservoir.

How does the network do it? The network uses a correlation of the reservoir's dynamics with the desired output. Not all the reservoir's neurons need be involved. As long as some cue is present in the reservoir's activation pattern, then there are correlations to be discovered. Because the reservoir's dynamics are nonlinear (they are based on a complex flow across the network), they are capable

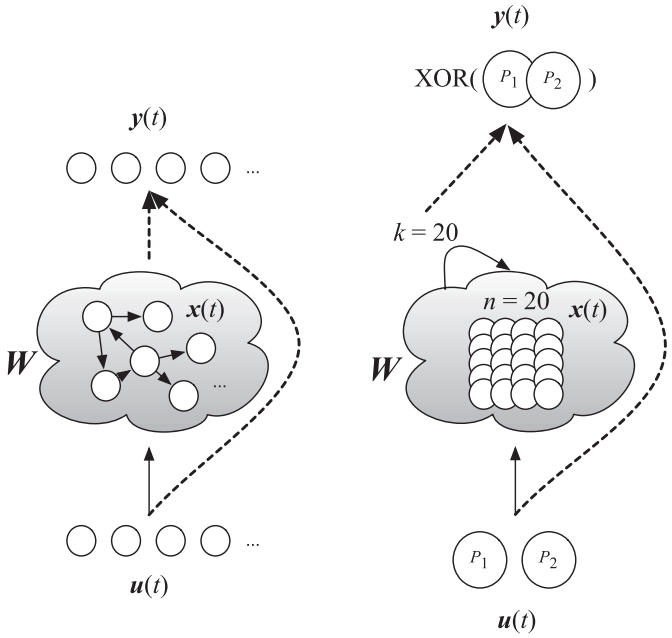

Fig. 1. Left: A basic reservoir computing system, referred to as an "echo state network." The network receives input (u(t)) into the reservoir (x(t)), which has a flow of activation over a bank of neurons. These activations feed into an output y(t) (along with the input pattern itself), and the connections at the output (dotted line) are trained. Right: An example model implemented to solve a task. In this case, we use 20 reservoir neurons and two inputs (truth/falsity of two propositions) to solve the XOR problem at the output (one neuron).

of "separating" solutions to the XOR problem. To find out what signature the network is using, we can perform an analysis on the reservoir's activities. We use principal component analysis (PCA) on these 20 neurons. This analysis finds the dominant patterns of variability that are driven by the input. Though it is outside the scope of our simplified demonstration, it is important to note that each principal component reflects a combination of neuron activities; the neurons that participate in a given XOR solution are said to statistically "load" onto these components. By using PCA we obtain a smaller number of components, and so we can visualize the network's behavior to illustrate how it is separating the XOR function results (true vs. false).

We extracted the two primary "components" or signatures that correlate with the solutions to XOR. This is shown in Fig. 2. Because the reservoir is dynamic, we can plot these two components over time. This can be interpreted as a low-dimensional visualization of the reservoir's dynamics that solve XOR. Fig. 2 shows the network transitioning across its $k = 20$ cycles from one region to another, in a clear pattern such that "true" outputs from XOR are in one region of this space, and "false" another.

This section serves as a general introduction to reservoir computing. We implemented a well-known version of this model (Jaeger, 2007; Lukoševičius, 2012), and showed that it can implement a classic logical operator (XOR). The reservoir's dynamics create a flow that produces distinct signatures which help the model correlate the reservoir with the desired outputs. This was all done without changing random weights on the reservoir — one need only a certain kind of nonlinear flow to make it work. Reservoir networks are different from simple recurrent networks and other more traditional connectionist models because complicated learning mechanisms like backpropagation are unnecessary. Indeed, they need nothing more than linear regression to learn nonlinear functions like XOR.

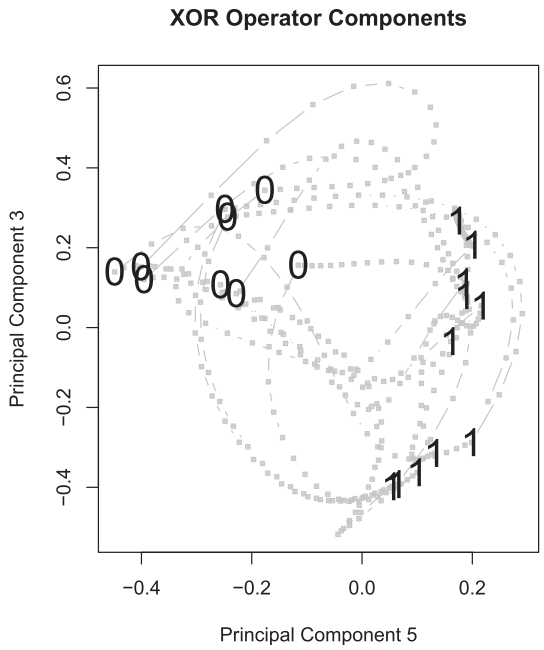

Fig. 2. PCA analysis in this particular model shows that components 5 and 3 are most closely related to the XOR solution. These components are measures of the underlying "signature" in the reservoir's dynamics. We can replot the dynamics of the reservoir on these two components. The gray lines are the reservoir's dynamics, transitioning across this two-dimensional signature. The numbers are points at which the network was asked to solve XOR. A clear separation between solutions is observed, reflecting its ability to identify "true" (1) or "false" (0) output of the XOR operator.

How do reservoir networks help us formalize and examine principles of emergent meaning? We apply this model to a simple, toy language system in the next section, showing that with these minimal commitments it can extract multiscale properties of a simplified communication system.

4. Demonstration in a simple model of language processing

Reservoir computing entails integrative dynamics. Whatever the structure of our input, a *single* reservoir of neurons integrates that input, and changes its dynamics. The dynamics of the reservoir generates signatures that can then be used at the output. An interesting property of this system is that it can solve complex dynamical problems. This is desideratum (i) noted above: Reservoir computing systems have memory, in the sense of carrying their history forward, by virtue of integrative, recurrent dynamics. In addition, the reservoir resonates across multiple timescales. This resonance is not demonstrated by the XOR problem because it does not span multiple timescales. To show that reservoir computing can also achieve desideratum (ii), we develop a simple example here, in a toy communication system.

Elman (1990) used a simple grammatical system to present a well-known recurrent network framework. It is common to use simplified input to demonstrate the behavior of model systems. We draw from this inspiration here, and demonstrate that the reservoir computing architecture can also integrate timescales across three levels. These levels are only represented in a highly simplified "toy" fashion. Nevertheless, the network is easy to build and interpret, and shows that it can solve desideratum (ii) above: timescale integration from phonemes to words to topics. We do not address desideratum (iii), multimodal integration, in the current simulation, but return to it in the General Discussion.

4.1. Simplified language system

Our simplified model "language" generates a sequence of symbols that feeds into a reservoir that is listening to a "monologue." The reservoir system predicts a series of inputs from someone "talking" to it. If we take these symbols to be a simplified sound system, we can consider the model to be an *addressee* predicting sound to sound to sound as input unfolds. This is of course only one side of dialogue, and it should be acknowledged that dialogue likely has important properties that cannot be captured in such a simplified simulation (cf. Pickering & Garrod, 2004). It will serve our purposes here, however, because it will show that a single model can satisfy the basic cognitive conditions described above.

Because we wish to represent multiple timescales, we created input that has three levels of temporal organization: sounds, words, and topics. These are, of course, just interpretive glosses on a highly simplified structure. Nevertheless the structure does have three layers of organization. These are shown in Fig. 3 below. On the left, we represent these three layers of organization: 2 topics, 3 words, and 5 sounds. Each topic is linked to two words (e.g., topic *a* is linked to words *i* and *ii*; *b* to *ii* and *iii*). Each word itself invokes a deterministic sequence of sounds. For example, word *i*, as shown, is generated by activating three sounds: 1, 2, and 3. Topics, like conversational topics, are "perseverative." This system will tend to stay on topic *a* if the prior word selected was in topic *a*. However, there's a certain probability that it might switch to topic *b*.

The reservoir model will only "hear" the sounds of this toy system. How we generate input is demonstrated in Fig. 3, right. A probabilistic algorithm first chooses a topic (e.g., *a*), samples from two words (*i* or *ii*), then generates a sequence of sounds. At the next time step, the "speaker" either stays on or switches from that topic, with a tendency to remain in the same topic. Occasionally, the speaker generates input to the network that "switches" topic (e.g., *b*). The speaker will then sample from words *ii* and *iii*, and will have a tendency to linger on that topic again. We can generate a large sequence of "sound" inputs from this toy system by running this sampling procedure over many iterations.

4.2. Example simulation

We investigated whether the network shows emergence of dynamic patterns, patterns that lie above the "sounds" input to the network. Can the reservoir detect topics, and can we observe the topic transitions in its dynamics, as we saw in the XOR case? In addition, how are words "represented" in this system? Because word *ii* is possible in both topic *a* and topic *b*, we would expect that the reservoir's dynamics should recognize that word *i* and word *iii* are the most semantically distant, while word *ii* represents a "border" word, a semantically related sequence, that may reflect transitions in both topic *a* or topic *b*.

4.2.1. Parameter settings

It is well known that this kind of reservoir system is sensitive to the size of the reservoir, with the general observation that computational power (in particular, memory span) corresponds roughly with the number of neurons in the reservoir. We therefore increased the reservoir size to 500 (though this global parameter can vary without impacting results). Unlike the XOR model, we did not "cycle" the reservoir for each sound presented ($k = 1$, where for XOR, $k = 20$). We generated a sequence of approximately 1300 of these sounds for training, using the rules specified in Fig. 3 (using a probabilistic phrase-structure grammar interpreter: Dale, 2007). These were converted into a pattern of activity, where a bank of neurons would have one of its neurons activated (set to 1) for each sound. The input was set to 0 for the inactive sounds. This is

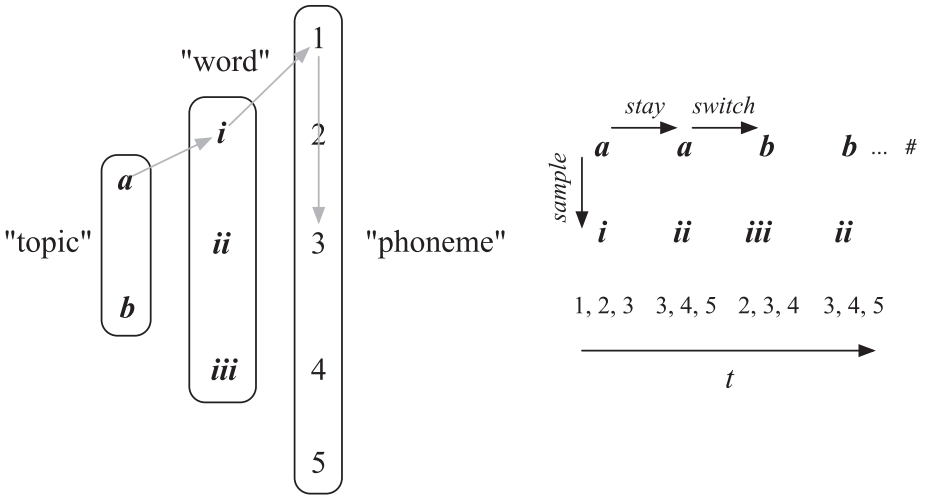

Fig. 3. Left: The structure of the toy system, from topics to sounds. The gray lines show an example sequence of sounds being generated by choosing a topic, sampling a word, and linearizing sounds. Right: An example sequence of input (sounds) being generated. The "hidden" variables of topic and word are not seen by the model, only the basis for generating the sounds. Note that topics will tend to be preserved by some "stay" probability. With some "switch" probability, the topic will change from a to b (or vice versa). After a period of time, the system ceases "talking" and generates a "end" marker, represented here by the hash character.

depicted in Fig. 4. The task of the network, as seen here, is to predict the next sound generated by the toy language model (Elman, 1990).

Neurons in the reservoir have continuous activations ranging from −1 to 1, and change in that range over time. All weights are initialized randomly in the reservoir to be between −0.5 and 0.5. The reservoir's weights are then scaled to ensure that the network will reveal what is termed the "echo state" property. As noted above, this property means that the network will not saturate (all neurons constantly active) or die out (all neurons off). It has an intrinsic, internal dynamic. More details on the configuration of the network are described in the Appendix (including links to source code).

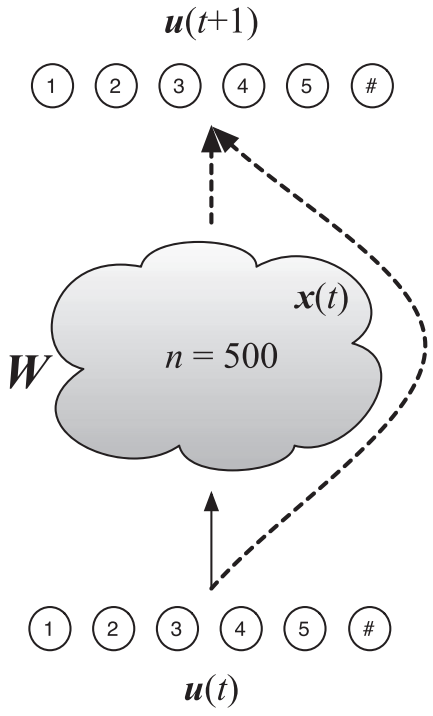

Fig. 4. The reservoir model for the simplified communication model is shown here. It has the same structure as the prior models, with a few exceptions. It has more reservoir neurons (500). It has 6 input and output nodes, which will be used to predict "sounds" that are part of the simulated language input. There are 5 sounds, and a final neuron that represents an end to the monologue. We generated thousands of sound inputs in order to train the network to predict.

4.2.2. Training of the model

The model is trained in precisely the same way as described above, and in more technical detail in the Appendix. The reservoir is perturbed by the input, and we track the reservoir's dynamics. These dynamics are then correlated using linear regression at the output. This linear regression's coefficients are used to set the weights (connections) from reservoir and input to output. This statistical method picks up the signature of the reservoir's activations that successfully generate prediction.

4.2.3. Testing the model

In order to test how the model is performing, we constructed a test sequence in which 21 sounds occurred in topic *a* (words *i* and *ii*) then switched to topic *b* (words *ii* and *iii*), for 42 total time steps. We again use PCA to assess how the reservoir's internal dynamics are working. If these dynamics contain signatures of multiple levels, then there should be components from the PCA, much like the XOR demonstration, that separate topic and word, even though the network is not receiving these as input. The dynamics of the reservoir are integrating the sequences of "sound," and these sequences give raise to signatures organized at larger timescales than the sounds themselves.

4.2.4. Analysis of network behavior

By testing the model's predictions with regression, as described above, we find that the model can predict the sequence of sounds very effectively, but does it learn anything about the topic sequence, and how words are related to each other in that topic space? To check this, we ran a PCA analysis as we did for XOR. We identified which components best predicted topic *a* vs. topic *b*, and then observed the dynamics of the reservoir system for a test set of sounds, when there is a transition occurring between topic *a* and topic *b*. We plotted the reservoir's dynamics based on which components best supported word prediction together with topic, on the same kind of PCA plot to determine the relative relationship between words *i*, *ii*, and *iii*. If the network is detecting the topic and

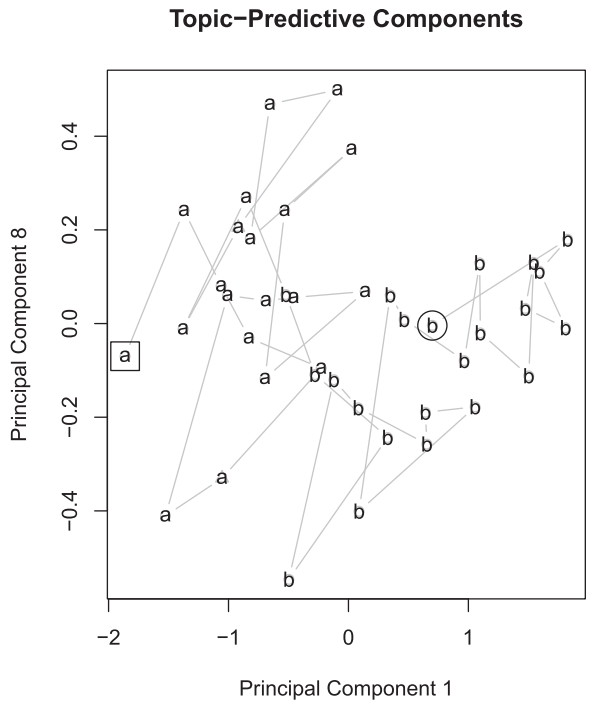

Fig. 5. After training this model on approximately 1300 sounds from this toy "monologue," we present approximately 40 sounds, with the first 21 from topic *a* and the next 21, after a switch, from topic *b*. The network starts in topic *a* (large square), and shows a distinct dynamic as it processes the words, predicting a shift into topic *b* (large circle).

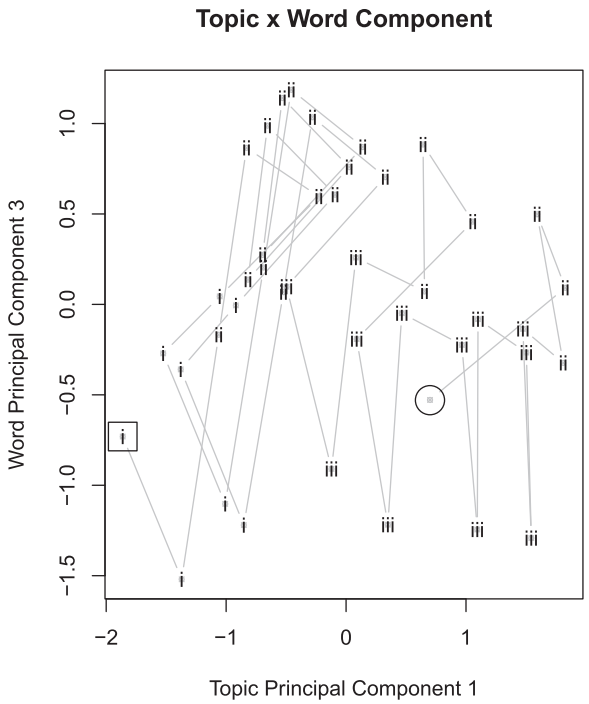

Fig. 6. We extracted the dynamic signatures in the reservoir activations that best predicted topic (x-axis) and word identity (y-axis). The neural network's readout can predict sound-to-sound patterns by using the structure of activation through the two higher-level patterns in the toy data set. Topic facilitates next prediction of sounds for word *i* and *iii*. Word identity is organized around topic too; when the signature (principal component) for topic is high, suggesting topic *a*, then the network predicts both words *ii* and *iii*, and does so by using the second dimension shown on the y-axis.

word semantics, then we should observe a distinct clustering within that PCA space, as we saw for the XOR solution.

4.2.5. Results

Fig. 5 shows the PCA results for one iteration of our toy simulation. Principal components 1 and 8 reflect the dynamic signatures in the reservoir that most strongly correlate with the network being in topic *a* or topic *b*. It is important to note that the network never sees topic *a* or topic *b*. Nevertheless, there are dynamic signatures present in the network, and learned at its output, that reflect this scale. In Fig. 5, we show how the network processes "sound" input from topic *a* which then switches at a few words in, to topic *b*. The PCA plot shows that the network separates these topics in its activation, and also has a coherent "flow" from one topic to another. In short, topic structure has "emerged" in how the network is perturbed from input.

Fig. 6 shows the network's behavior during testing in a different way. On the x-axis, we show the strongest component that reflects topic shifts. On the y-axis, we take the strongest component that predicts word identity. The plot is labeled with *i*, *ii*, or *iii*, depending on which word is being input to the network. Word *iii* is on the far right of the plot, most distant from word *i*, on the left. These reflect the same regions of space relevant to the topic structure. Importantly, word *ii*'s location is intermediate, and can sometimes be strongly present on the left side of the plot or the right, conditioned on topic. The dynamical pattern relating to word ii changed as a function of its topical context. In this sense, the network encodes a semantic gradient influenced by the topic being processed.

In a follow-up analysis of the network's performance, we discovered that the behavior of the reservoir's dynamics tended to follow a regularity in the way that levels were integrated. Topic tended to dominate the dynamics of the network, with words next, and then sounds — the only elements the network "heard" — dominating the later components. This analysis suggests that the network substantially organized its performance around topical structure, with word structure nested within topical structure, and sound structure nested within word structure.

To further examine the hierarchical organization of reservoir dynamics, we again took the principal components from the PCA analysis, and tested which were best accounted for by the relevant scales: sounds, words, or topics. In PCA, the components (the "signature" in the reservoir) are ordered by strength, with earlier PCA components having higher relevance than later components.[2] Fig. 7 shows this effect. On the x-axis we order the signatures (principal components) according to their strengths. On the y-axis is the extent to which a given level of analysis, such as topic, predicts the values on that signature. A general trend we obtain is that topic best predicts earlier principal components, and sound, the finest-grain level, better predicts the later components. This suggest that the reservoir's largest source of variation can be used to predict topic, and this can be used to organize the lower-level dynamics of the system, which themselves are instantiating these higher-level patterns through smaller fluctuations in the reservoir as it is perturbed by each sound.

The results reported here represent a single instantiation of our model. The code is available publicly to those who wish to test these

[2] There is an important simplification to note here. "Strength" is used as a convenient stand in for "variance." Principal components are ordered by the variance in the reservoir data that they account for. This does not mean that principal component one is necessarily more impactful. It depends on how those components relate to other factors and measures of performance.

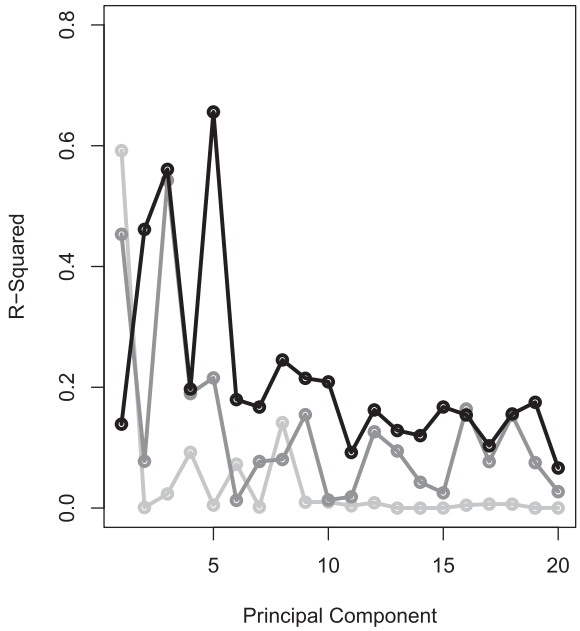

Fig. 7. How well different levels of analysis predict the scores on principal components (the signatures in the 500-dimensional reservoir system). Topic tends to dominate the lowest component, and sounds the later components. The network has dynamics that resolve distinct levels of analysis, all while integrating them as suggested by desideratum (ii) in section 4. R-Squared reflects the percentage variance in principal component explained by multinomial factors. Light gray: topics, darker gray: words, black: sounds.

patterns under different parameterizations or inputs.[3] We ran 100 iterations of our model, and combined results into one plot. This is shown in Fig. 8, and the general pattern holds robustly. Topic and words have higher relationships to first few components, and sounds to the later components.

4.3. Discussion

In our simple simulation, we see that a relatively simple reservoir computing model produces multiscale dynamics, without explicit representations for each level of language structure. Topic of conversation, and word identity, "emerge" from the dynamics of sound and their surrounding context.

Using PCA on the reservoir activation patterns, we find that the network can solve this problem by combining two strategies. On one dynamic signature in the reservoir, it responds to the broader topical context (Fig. 6, x-axis). On another signature, it marks whether word *ii* is occurring rather than *i* or *iii* (Fig. 6, y-axis). By combining these two sources of information at the readout, the network has predictive dynamics that resonate across all three timescales: topic, word, and sound prediction. The network does this through what we termed in section 4 as *constitutive integration*. The model handles all levels simultaneously. This solves desiderata (i) and (ii).

These results are related to what Elman (1990) demonstrated with words and letters (words emerge from sequential, lower-level statistics), and what Botvinick and Plaut (2004) showed with hierarchical action sequences (a neural model can learn higher-level action hierarchies from exemplar statistics). Our simple model further supports these and other demonstrations showing that what is often deemed "abstract" or "symbolic" can naturally emerge from a probabilistic system, processing information in time (cf. Christiansen & Chater, 1999).

What is unique about the present model is that it does so simply on the basis of the "resonant" dynamics of the reservoir computing system. No abstract, complex rule system with hierarchical encapsulation is required. In fact, the weights between neurons in the reservoir are not learned. The model integrates the statistics of fast-changing sounds. These statistics are organized at higher levels (words, topics), and these higher levels are reflected in the system's dynamics without any further specification. The model has theoretical implications which we elaborate in the General Discussion. Put in pointed terms: Our results demonstrate that a multidimensional dynamic system can integrate activation across several timescales and thereby "resonate" with patterns generated by a simple communication system; no special machinery is required.

5. General discussion

We began this paper with a question: What kind of cognitive system can produce the *composite* nature of natural language? How do humans make sense? Necessary conditions, we argued, are (i) dynamic memory, (ii) timescale integration, and (iii) multimodal integration. Herbert Simon's speaker in the Singapore cafe serves as a striking illustration of these issues. Natural language performance invokes interlocked dynamic patterns that reflect an ebb and flow at differing timescales — from quick bursts of gestural or articulatory activity, to slow motion bursts that constitute one turn in a conversation, or even an entire lecture. Phonemes unfold in milliseconds, words in a second or less, but thoughts and topics in a conversation change more smoothly, fluidly, over seconds and minutes.

We presented a model to show that recurrent dynamics provide a rich medium for fulfilling these computational needs, even in randomly structured networks. Our model is admittedly simple, but it reveals this basic fact — special "equipment" is not needed as much as our mind needs a capacity to weave the timescales together through nonlinear dynamics.

5.1. Limitations

Our simulations addressed desiderata (i) and (ii), but what about (iii), multimodal integration? The general-purpose nature of reservoir computing provides a ready answer. Networks of random connectivity will integrate information over any kinds of inputs signals, regardless of whether they originate from different modalities with the different intrinsic structures. Indeed this is the beauty of reservoir computing: A reservoir with rich nonlinear interactions will naturally produce a large array of nonlinear conjunctions and disjunctions of their inputs. As the size and complexity of reservoir dynamics grows, it becomes increasingly likely that some of the nonlinear combinations will be useful for learning in many tasks and contexts. Therefore, no special circuitry is needed to perform multimodal integration by taking into account the particular structures inherent to particular modalities.[4] A multimodal simulation falls outside the scope of this study, but previous studies have demonstrated multimodal integration in reservoir computing models applied to robot navigation and control tasks that rely on multiple sensor and actuator signals (Antonelo, Schrauwen, Dutoit, Stroobandt, & Nuttin, 2007; Antonelo, Schrauwen, & Stroobandt, 2008; see also Heinrich,

[3] https://github.com/racdale/emergent-meaning.

[4] Multimodality may be supported by distinct input systems, which could approximate modular structure, at least in early stages of processing before reaching a highly integrative, multimodal "cortex."

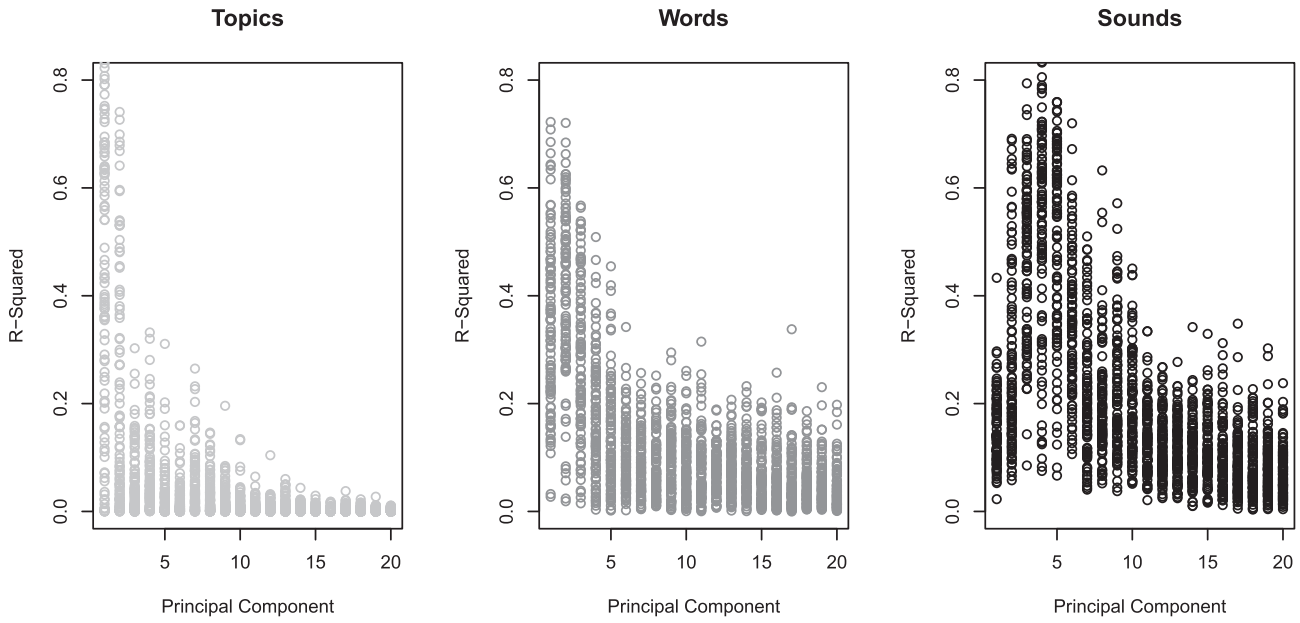

Fig. 8. The same data shown in Fig. 7, but with 100 simulations, and separated by timescale. The trend in the example demonstration holds up robustly. The largest PCA components account for the longest timescales; finer-grained signatures (subsequent PCA components, along the x-axis) reflect words, then sounds.

2016; Heinrich, Magg, & Wermter, 2015).

The simplicity of our model may raise another concern about whether this architecture can scale up to more complex and hallmark properties of language. For example, syntactic recursion is often regarded as a central and requisite feature of human language skill, perhaps even an innate and defining feature of human language (e.g., Hauser et al., 2002; Watumull, Hauser, Roberts, & Hornstein, 2014).[5] Encouragingly, several researchers have already demonstrated syntactic capacities — and generalization — in reservoir systems. Frank (2006) has shown that reservoir computing can solve basic recursive problems from the most stringent concepts of so-called "systematicity" (see Fodor & Pylyshyn, 1988). Tong and colleagues show that reservoir computing can carry out generalization in a language model, and outperforms other neural network architectures (Tong, Bickett, Christiansen, & Cottrell, 2007; see also Farkaš & Crocker, 2008). Indeed Dominey (2013, 1995), who originated reservoir computing to a great extent, recently argued that certain configurations of reservoir computing can extract structural features of language (e.g., syntactic aspects) as an emergent property of integrating semantic features of the world (Dominey, 2013, p. 11; see also Schoenemann, 1999 for related theoretical discussion):

> *Based on the notion that infants can parse visual scenes by detecting sequences of perceptual primitives …, we developed an event parser that could detect actions including take, take-from, give, push and touch. … What we will find, is that as the cognitive systems of robots become increasingly sophisticated, they will naturally afford richer language. For example, as mental simulation capabilities develop, the need for verb aspect to control the flow of time in these simulations will naturally arise.*

How do the results of our simple demonstration relate to prior work on recurrent neural networks? What we have demonstrated is indeed consistent with prior work, such as Elman (1990), and later Elman (2004), who demonstrates that prediction and contextual facilitation occur in simple recurrent networks. Sometimes referred to as "shading," a recurrent network will show subtle fluctuations in its activation in response to contextual factors (cf. Botvinick & Plaut, 2006; in serial memory).[6] These fluctuations will "shade" the meaning generated by a word presented to a network. This modulation by context would lie at a longer timescale, akin to the "topics" in our toy simulation.

Inspired by this prior work, what we have shown is an existence proof that a communication structure can be processed at multiple timescales. We simulated three layers of structure meant to reflect the kind of structuring of language (albeit in highly simplified form). With sufficient dimensionality to work with, a complex stream of information resonates at multiple timescales. So, as reviewed above, while reservoir computing systems can "solve syntax," we may also say that these systems can "solve synergies," in the sense that they are able to synergize multi-level statistics by virtue of their nonlinear dynamics alone (see Dale, Kello, et al., 2016; Fusaroli, Rączaszek-Leonardi, & Tylén, 2014; for related discussion).

There are important implications to this basic observation. Imagine presenting a series of words under a given topic. Because of the recurrent property of the network, this will establish a particular signature in the activity of the reservoir neurons. At this point, when a new word is sequentially presented, the network changes under two important constraints. First, the word acts to

[5] Other aspects of the model to expand the language system could include the distributional features of the structural levels (sound-to-word-to-message ratios should better reflect natural language reuse). In addition, admittedly, phonological neighborhoods of words could also be simulated more faithfully.

[6] In fact, effects of shading in prior demonstrations have been relatively subtle, appearing as minor shifts in the trajectories when PCA is used to illustrate them. Here, it may be interpreted as a rather larger effect as seen in Fig. 6. This may be an illusion of our analysis: The principal components here are showing the most extreme statistical patterns predicting word or topic identity. In addition, the principal component for word (y-axis) shows that words are relatively stable along some components regardless of context. Reservoir computing may thus serve as an exploratory arena to examine what aspects of context may more or less radically impact the processing *dynamics* of words in context. This is outside the scope of the present discussion, but we thank a reviewer for noting this fact.

"perturb" the system in a manner that will be consistent with that word's prior presentations. In other words, hearing "d-o-g" will perturb a language comprehension system in some consistent ways — projecting in particular patterns, resonating in the network. However, these perturbations will not be occurring in a vacuum. The prior pattern of activation over the reservoir will *modulate* those perturbations, as the reservoir has a memory. The word is thus "folded" into an ongoing topic, thereby allowing the network to predict sounds in a contextually facilitated way — words and topics serve to structure the dynamics in a systematic way even as the network only "receives" sounds.

The current simulation is obviously limited in assuming language processing is a one-way process of "monologue." Our model only captures a unidirectional flow of information, as if one reservoir system, acting as a "learner," is extracting the dynamics of an external input source (an "adult" speaker). It would be easy to scale this up. Exploring the properties of interactive neural models is an important future step in this domain (see, e.g., Dale, Fusaroli, et al., 2016 for an example). Interactive models could allow investigation of "mutual emergent meaning," in two systems that do not yet have the capacity to "make sense," in linguistic terms. Indeed, there are many such evolutionary models that use agent-based and other simulations (see Cangelosi & Parisi, 2002; Christiansen & Dale, 2003; for early reviews of this work).

5.2. Theoretical issues

Reservoir computing is still new to the behavioral and cognitive sciences, but it has been studied in computer science and neuroscience for over a decade. Here we review some of the theoretical issues that have arisen in this literature, and consider how they might relate to questions and phenomena in the behavioral and cognitive sciences. The theoretical issue that has received perhaps the most attention is how to construct a reservoir with dynamics that are generically useful for computation. The most common answer to this question is that reservoir dynamics tend to be most useful when they are balanced between convergence and divergence (Lukoševičius & Jaeger, 2009), or between order and chaos (Bertschinger & Natschläger, 2004). Kello (2013) showed that this balance can be achieved in a spiking neural network that dynamically regulates spike propagation to hover near a critical branching point (Beggs & Plenz, 2003). Put simply, the critical branching point is where spike propagation is conserved over time, such that spike rates remain stable on average (that is, rates neither converge to zero or diverge to infinity). Numerous studies have reported evidence for critical branching in a range of neural systems (Hahn et al., 2010), and theoretical analyses have shown that critical branching is beneficial to the computational (reservoir) capacity of spiking networks (Shew et al., 2011). Kello (2013) argued that the effects of critical branching can be seen even in human performance as power law distributions, including language performance (Kello, Anderson, Holden, & Orden, 2008). Taken together, these and other studies suggest that many neural and cognitive systems, including language systems, may be tuned and maintained for the purpose of reservoir computing.

Is reservoir computing neurophysiologically plausible? It's important to note that neural models, in general, suffer from weak plausibility. This is to be expected, given the scale and complexity of the central nervous system, and the still-nascent state of understanding how computation works at higher levels of the brain. For example, backpropagation has long been considered to be neurally implausible, but some are revisiting this dogma (Lillicrap, Cownden, Tweed, & Akerman, 2016).

There are reasons to suspect that reservoir computing reflects some general organizational principles of cognition as it is implemented in the mammalian brain. For example, the original proposal from Dominey (1995) was that cortico-cortical recurrent connections could support this kind of nonlinear dynamics, by implementing something akin to a working memory that may be especially relevant to prefrontal cortex. These dynamics could then be wired up with other circuits in the nervous system to support dynamic processing of a complex sort. For example, Dominey's original work (1995; Dominey, Arbib, & Joseph, 1995) demonstrated that cortical networks with recurrent nonlinear dynamics can help establish reward-based learning via corticostriatal projections which are trained, akin to the "readout" of the reservoir system.

More generally, the idea of recurrent connections serving as a computational foundation is consistent with new discoveries that the nervous system may require substantial experience-dependent sorting and plasticity during development. Epigenetic developmental trajectories during early human brain development cannot sort strict functional pre-specification of cortical regions (Buckner & Krienen, 2013). This organizational principle means that much of cortex may be setup to permit multimodal integration, so much so that some have argued that the mammalian neocortex is "essentially multisensory" (Ghazanfar & Schroeder, 2006). In fact, if it is true that even random projections in a growing brain permit more complex sequential processing, then the reservoir framework may be well situated to explore more systematic learning and sorting that may take place over those initially random projections. This may by itself support recursive processing akin to syntax (e.g., Frank, 2006). Curiously, some work on comparing species by brain volume finds that an evolutionary growth of absolute brain volume supports sequential memory (Stevens, 2014), as might be predicted from the learning dynamics of reservoir size. In addition, there may be "critical cognitive thresholds" in the global parameters of these models, providing a potential arena for exploring nonlinear thresholds in the evolutionary processes of brain and cognition (e.g., Herculano-Houzel, 2012; Schoenemann, 1999).

5.3. Conclusion

In a simple model of language processing, we demonstrated that a randomly connected reservoir can encode a dynamic memory and integrate multiple timescales, and we argued that inherent encoding capacity can be foundational to sense-making. The composite nature of language is captured by nonlinear dynamics that combine patterns across multiple timescales and modalities. The various levels of language processing may thereby interact quickly, continually, and sometimes effortlessly, to make sense from the sounds and sights that compose our acts of language. The "sense" humans make is a self-organizational property of the cognitive system's behavior, resonating in a manner that weaves fluctuations from sounds to topical context.

Appendix

The model in the paper is based on the reservoir computing architecture known as "echo state networks." These represent a major emerging tradition in this domain of neural modeling (Jaeger, 2001, 2007). They can be distinguished from another brand of reservoir computing called "liquid state machines" (Maass, Natschläger, & Markram, 2002). These models have similar properties, but are implemented slightly differently. Liquid state machines employ spiking neurons, whereas echo state networks (the architecture used here) use leaky integrate-and-fire neurons. These frameworks can also model interconnectivity in the reservoir in different ways. For example, echo state networks often have highly sparse connectivity in the reservoir (1% of reservoir neurons

connected). In general they are regarded as subtly different architectures co-discovered in the early 2000s, and have similar properties (Jaeger, 2001; originally these properties discovered by Dominey, 1995).

We use an implementation offered by Lukoševičius (2012) and implemented in R. All code that we use is derived from his initial demonstrations, and can be found here: https://github.com/racdale/emergent-meaning

The technical underpinnings of the framework are elegantly introduced by Lukoševičius (2012), and can be simply described here. A reservoir of neurons in the echo state network is represented as vector $\mathbf{x}(t)$, and its activation from time step to time step is a function of itself (multiplied by a set of connection weights) and randomly connected input, $\mathbf{u}(t)$:

$$\mathbf{x}(t) = (1\text{-}d)\,\mathbf{x}(t\text{-}1) + d\,\tanh(\mathbf{W'u}(t\text{-}1) + \mathbf{Wx}(t\text{-}1))$$

W' is a weight matrix, randomly initialized, between inputs and reservoir. d is a decay parameter reflecting how leaky the simulated neurons are. Connections between these layers of neurons are initialized using uniformly random values from −0.5 to 0.5. d was set to 0.3 (the simulation does not rely on any narrow range of these parameters). The reservoir weight matrix **W** is rescaled using its singular spectrum — in order to have stable dynamics under recurrent feedback, the maximum singular value of the weight matrix must be near one (Jaeger, 2001, p. 13; Lukoševičius, 2012). As specified by Lukoševičius, this is done by randomly initializing **W** then performing a rescaling:

$$\mathbf{W} := 1.25\,\mathbf{W} \,/\, \lambda(\max)$$

$\lambda(\max)$ is the maximum singular value of the matrix **W** once it is randomly initialized.

To explore the dynamics here, we use principal component analysis (PCA) on the history of reservoir activations. We then fit various linear models to the components of the PCA output to determine which component in the network resonates most closely with the input perturbation of interest (e.g., current topic, vs. current sound). PCA essentially rotates the activation matrix of the reservoir, and we take the rotated data using the component that best reflects the levels of interest. The new rotated data is then plotted over time to reflect the dynamics of that signature (e.g., XOR in section 3).

References

Antonelo, E. A., Schrauwen, B., Dutoit, X., Stroobandt, D., & Nuttin, M. (2007). Event detection and localization in mobile robot navigation using reservoir computing. In *International conference on artificial neural networks* (pp. 660–669).

Antonelo, E. A., Schrauwen, B., & Stroobandt, D. (2008). Event detection and localization for small mobile robots using reservoir computing. *Neural Networks, 21*(6), 862–871.

Beggs, J. M., & Plenz, D. (2003). Neuronal avalanches in neocortical circuits. *Journal of Neuroscience, 23*(35), 11167–11177.

Bertschinger, N., & Natschläger, T. (2004). Real-time computation at the edge of chaos in recurrent neural networks. *Neural Computation, 16*(7), 1413–1436.

Botvinick, M., & Plaut, D. C. (2004). Doing without schema hierarchies: A recurrent connectionist approach to normal and impaired routine sequential action. *Psychological Review, 111*(2), 395.

Botvinick, M. M., & Plaut, D. C. (2006). Short-term memory for serial order: A recurrent neural network model. *Psychological Review, 113*(2), 201.

Buckner, R. L., & Krienen, F. M. (2013). The evolution of distributed association networks in the human brain. *Trends in Cognitive Sciences, 17*(12), 648–665.

Caluwaerts, K., D'Haene, M., Verstraeten, D., & Schrauwen, B. (2013). Locomotion without a brain: Physical reservoir computing in tensegrity structures. *Artificial Life, 19*(1), 35–66.

Cangelosi, A., & Parisi, D. (2002). *Simulating the evolution of language*. Springer Science & Business Media.

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

Christiansen, M. H., & Dale, R. (2003). Language evolution and change. In M. Arbib (Ed.), *The handbook of brain theory and neural networks* (pp. 604–606). MIT Press.

Dale, R.. (2007). Random sentences from a generalized phrase-structure grammar interpreter. arXiv preprintcs/0702081.

Dale, R., Fusaroli, R., Tylén, K., Rączaszek-Leonardi, J., & Christiansen, M. H. (2016). A recurrent network approach to modeling linguistic interaction. In J. Trueswell, A. Papafragou, D. Grodner, & D. Mirman (Eds.), *Proceedings of the 38th annual meeting of the cognitive science society* (pp. 901–906). Austin, TX: Cognitive Science Society.

Dale, R., Kello, C. T., & Schoenemann, P. T. (2016). Seeking synthesis: The integrative problem in understanding language and its evolution. *Topics in Cognitive Science, 8*(2), 371–381.

De Jaegher, H., & Di Paolo, E. (2007). Participatory sense-making. *Phenomenology and the Cognitive Sciences, 6*(4), 485–507.

Di Paolo, E. A. (2005). Autopoiesis, adaptivity, teleology, agency. *Phenomenology and the Cognitive Sciences, 4*(4), 429–452.

Dominey, P. F. (1995). Complex sensory-motor sequence learning based on recurrent state representation and reinforcement learning. *Biological Cybernetics, 73*(3), 265–274.

Dominey, P. F. (2013). Recurrent temporal networks and language acquisition—from corticostriatal neurophysiology to reservoir computing. *Frontiers in Psychology, 4*, 500.

Dominey, P., Arbib, M., & Joseph, J.-P. (1995). A model of corticostriatal plasticity for learning oculomotor associations and sequences. *Journal of Cognitive Neuroscience, 7*(3), 311–336.

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

Elman, J. L. (2004). An alternative view of the mental lexicon. *Trends in Cognitive Sciences, 8*(7), 301–306.

Enfield, N. J. (2013). *Relationship thinking: Agency, enchrony, and human sociality*. Oxford University Press.

Farkaš, I., & Crocker, M. W. (2008). Syntactic systematicity in sentence processing with a recurrent self-organizing network. *Neurocomputing, 71*(7), 1172–1179.

Fodor, J. A., & Pylyshyn, Z. W. (1988). Connectionism and cognitive architecture: A critical analysis. *Cognition, 28*(1), 3–71.

Frank, S. L. (2006). Strong systematicity in sentence processing by an echo state network. In *International conference on artificial neural networks* (pp. 505–514).

Fusaroli, R., Rączaszek-Leonardi, J., & Tylén, K. (2014). Dialog as interpersonal synergy. *New Ideas in Psychology, 32*, 147–157.

Ghazanfar, A. A., & Schroeder, C. E. (2006). Is neocortex essentially multisensory? *Trends in Cognitive Sciences, 10*(6), 278–285.

Gibson, E. (1998). Linguistic complexity: Locality of syntactic dependencies. *Cognition, 68*(1), 1–76.

Graesser, A. C. (2011). Learning, thinking, and emoting with discourse technologies. *American Psychologist, 66*(8), 746–757.

Hahn, G., Petermann, T., Havenith, M. N., Yu, S., Singer, W., Plenz, D., et al. (2010). Neuronal avalanches in spontaneous activity in vivo. *Journal of Neurophysiology, 104*(6), 3312–3322.

Hauser, M. D., Chomsky, N., & Fitch, W. T. (2002). The faculty of language: What is it, who has it, and how did it evolve? *Science, 298*(5598), 1569–1579.

Heinrich, S. (2016). *Natural language acquisition in recurrent neural architectures*. Dissertation. Department of Informatics, Universitat Hamburg http://ediss.sub.uni-hamburg.de/volltexte/2016/7972/.

Heinrich, S., Magg, S., & Wermter, S. (2015). Analysing the multiple timescale recurrent neural network for embodied language understanding. In *Artificial neural networks* (pp. 149–174). Cham: Springer. https://link.springer.com/chapter/10.1007/978-3-319-09903-3_8.

Herculano-Houzel, S. (2012). The remarkable, yet not extraordinary, human brain as a scaled-up primate brain and its associated cost. *Proceedings of the National Academy of Sciences, 109*(1), 10661–10668.

Jaeger, H.. (2001). The "echo state" approach to analysing and training recurrent neural networks-with an erratum note. Bonn, Germany: German National Research Center for Information Technology GMD Technical Report, 148(34).

Jaeger, H. (2007). Echo state network. *Scholarpedia, 2*(9), 2330.

Kello, C. T. (2013). Critical branching neural networks. *Psychological Review, 120*(1), 230–254.

Kello, C. T., Anderson, G. G., Holden, J. G., & Orden G C, V. (2008). The pervasiveness of 1/f scaling in speech reflects the metastable basis of cognition. *Cognitive Science, 32*(7), 1217–1231.

Kopp, S., Krenn, B., Marsella, S., Marshall, A. N., Pelachaud, C., Pirker, H., et al. (2006). Towards a common framework for multimodal generation: The behavior markup language. In *International workshop on intelligent virtual agents* (pp. 205–217).

Kuhlen, A. K., & Brennan, S. E. (2010). Anticipating distracted addressees: How speakers' expectations and addressees' feedback influence storytelling. *Discourse Processes, 47*(7), 567–587.

Langton, S. R., Watt, R. J., & Bruce, V. (2000). Do the eyes have it? Cues to the direction of social attention. *Trends in Cognitive Sciences, 4*(2), 50–59.

Lillicrap, T. P., Cownden, D., Tweed, D. B., & Akerman, C. J. (2016). Random synaptic feedback weights support error backpropagation for deep learning. *Nature Communications, 7*.

Louwerse, M. M., Dale, R., Bard, E. G., & Jeuniaux, P. (2012). Behavior matching in multimodal communication is synchronized. *Cognitive Science, 36*(8), 1404–1426.

Lukoševičius, M. (2012). A practical guide to applying echo state networks. In G. Orr, & K. R. Mueller (Eds.), *Neural networks: Tricks of the trade* (pp. 659–686). Springer Berlin Heidelberg.
Lukoševičius, M., & Jaeger, H. (2009). Reservoir computing approaches to recurrent neural network training. *Computer Science Review, 3*(3), 127–149.
Maass, W., Natschläger, T., & Markram, H. (2002). Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural Computation, 14*(11), 2531–2560.
Newell, A. (1992). Précis of unified theories of cognition. *Behavioral and Brain Sciences, 15*(3), 425–437.
Noë, A. (2009). *Out of our heads: Why you are not your brain, and other lessons from the biology of consciousness*. Macmillan.
Pickering, M. J., & Garrod, S. (2004). Toward a mechanistic psychology of dialogue. *Behavioral and Brain Sciences, 27*(2), 169–190.
Pickering, M. J., & Garrod, S. (2013). An integrated theory of language production and comprehension. *Behavioral and Brain Sciences, 36*(4), 329–347.
Plunkett, K., & Elman, J. L. (1997). *Exercises in rethinking innateness: A handbook for connectionist simulations*. MIT Press.
Schoenemann, P. T. (1999). Syntax as an emergent characteristic of the evolution of semantic complexity. *Minds and Machines, 9*(3), 309–346.
Shew, W. L., Yang, H., Yu, S., Roy, R., & Plenz, D. (2011). Information capacity and transmission are maximized in balanced cortical networks with neuronal avalanches. *Journal of Neuroscience, 31*(1), 55–63.
Simon, H. A. (1992). What is an "explanation" of behavior? *Psychological Science, 3*(3), 150–161.
Spivey, M., Richardson, D., & Fitneva, S. (2004). Thinking outside the brain: Spatial indices to linguistic and visual information. In J. Henderson, & F. Ferreira (Eds.), *The interface of vision language and action* (pp. 161–189). New York: Psychology Press.
Squire, L. R., & Wixted, J. T. (2011). The cognitive neuroscience of human memory since HM. *Annual Review of Neuroscience, 34*, 259–288.
Stevens, J. R. (2014). Evolutionary pressures on primate intertemporal choice. *Proceedings of the Royal Society of London B: Biological Sciences, 281*(1786), 20140499.
Tong, M. H., Bickett, A. D., Christiansen, E. M., & Cottrell, G. W. (2007). Learning grammatical structure with echo state networks. *Neural Networks, 20*(3), 424–432.
Tylén, K., & Allen, M. (2009). *Interactive sense-making in the brain*. Enacting Intersubjectivity: Paving the Way for a Dialogue Between Cognitive Science, Social Cognition and Neuroscience (pp. 224–241).
Watumull, J., Hauser, M. D., Roberts, I. G., & Hornstein, N. (2014). On recursion. *Frontiers in Psychology, 4*, 1017.
Yoshimi, J. (2012). Active internalism and open dynamical systems. *Philosophical Psychology, 25*(1), 1–24.