Structured event memory: a neuro-symbolic model of event cognition

Nicholas T. Franklin

Harvard University

Kenneth A. Norman

Princeton University

Charan Ranganath

University of California, Davis

Jeffrey M. Zacks

Washington University in St. Louis

Samuel J. Gershman

Harvard University

Author Note

We are grateful to Momchil Tomov for comments on an earlier draft of the paper. This work was supported by a Multi-University Research Initiative grant (ONR/DoD N00014-17-1-2961).

Abstract

Humans spontaneously organize a continuous experience into discrete events and use the learned structure of these events to generalize and organize memory. We introduce the *Structured Event Memory* (SEM) model of event cognition, which accounts for human abilities in event segmentation, memory, and generalization. SEM is derived from a probabilistic generative model of event dynamics defined over structured symbolic scenes. By embedding symbolic scene representations in a vector space and parametrizing the scene dynamics in this continuous space, SEM combines the advantages of structured and neural network approaches to high-level cognition. Using probabilistic reasoning over this generative model, SEM can infer event boundaries, learn event schemata, and use event knowledge to reconstruct past experience. We show that SEM can scale up to high-dimensional input spaces, producing human-like event segmentation for naturalistic video data, and accounts for a wide array of memory phenomena.

Keywords: event cognition; memory; latent structure; Bayesian non-parametric models; neural networks

Structured event memory: a neuro-symbolic model of event cognition

Introduction

Although sensory input arrives continuously, our subjective experience is punctuated by the perception of events with identifiable beginnings and ends (Radvansky & Zacks, 2014). These event representations are abstract and schematic, transcend specific sensory details and allow us to generalize our knowledge across time and space. By distilling the latent structure underlying the sensory stream, event representations can be used to reconstruct the past, comprehend the present, and predict the future.

Despite the centrality of events in human cognition, there has been a notable dearth of formal models. This speaks in part to the scope and complexity of the modeling challenge; in some ways, a theory of event cognition is a theory of cognition writ large. In this paper, we take on the challenge, developing an integrated model that explains how humans segment, remember and generalize events. We frame event cognition in the language of statistical learning theory and argue that events serve to organize and structure continuous experience. This structure, and the generalization it affords, explains many of the empirical phenomena observed in event cognition. Along the way, we touch upon broader issues in cognitive science, including the need for structured representation, probabilistic reasoning, and parallel distributed processing. Our approach synthesizes ideas from symbolic and neural network modeling traditions, demonstrating how these ideas can work together to produce human-like competence in event cognition.

In what follows, we first summarize key empirical findings in the human event cognition literature and discuss existing theoretical treatments. We then lay out a general computational-level analysis, framing event cognition in terms of a common probabilistic generative model of events that can be inverted for different computations. This analysis forms the basic architecture of our model, which we then elaborate with a number of specific assumptions about the underlying dynamics, representations, storage

and retrieval processes, and inductive biases. In the Results, we put the model to work, simulating a range of empirical phenomena. In the Discussion, we consider the strengths and weaknesses of our modeling framework, how it compares to previous theoretical treatments, and directions for future work.

Theoretical and empirical background

A comprehensive theory of human event cognition must address the following questions:

- *Segmentation*: How do people identify event boundaries from the continuous sensory stream?
- *Learning*: How do people acquire knowledge about the internal structure of events from experience?
- *Inference*: How do people use their knowledge of events to make inferences about unobserved properties of the world?
- *Prediction*: How do people use their knowledge of events to make predictions about the future?
- *Memory*: How do people use their knowledge of events to reconstruct the past?

Here we briefly review the empirical data and theoretical ideas pertaining to each of these questions.

Segmentation

Event segmentation has been primarily studied using subjective judgments about event boundaries in movies and text. These judgments are consistent between subjects (Newtson & Engquist, 1976; Zacks, Tversky, & Iyer, 2001) and tend to track feature changes (Hard, Tversky, & Lang, 2006) or important statistical (Baldwin, Andersson, Saffran, & Meyer, 2008; Schapiro, Rogers, Cordova, Turk-Browne, & Botvinick, 2013) and causal boundaries (Baldwin et al., 2008; Radvansky, 2012). This process appears to

happen automatically and is identifiable in fMRI signals even in the absence of an explicit segmentation task (Baldassano et al., 2017; Speer, Zacks, & Reynolds, 2007).

According to *Event Segmentation Theory* (EST; Radvansky & Zacks, 2011; Zacks, Speer, Swallow, Braver, & Reynolds, 2007), people maintain an active representation of the event structure (the *current model*) that they use both to interpret the current situation and to predict what will happen within each event. When these predictions are violated, a prediction error signals the occurrence of an event boundary (Zacks, Kurby, Eisenberg, & Haroutunian, 2011; Zacks et al., 2001). This idea has been implemented in a neural network model of event segmentation (Reynolds, Zacks, & Braver, 2007), which used a context layer within a recurrent neural network to represent an event model. The activity of the context layer modulates the activity of the hidden layers of the network, thus varying the predicted dynamics by events. This architecture is similar to gating mechanisms long used in recurrent network models (Hochreiter & Schmidhuber, 1997). When simulated on a 3-D motion capture data set consisting of several short movements of a single person (e.g. chopping wood) concatenated together, prediction error in the model corresponded with transitions between events and was used to signal updates of the context layer.

Other models of event dynamics have used recurrent neural networks with the task of predicting the next item in a sequence. Schapiro and colleagues (2013) examined whether transition uncertainty in the environment was necessary for an event boundary. That is, one prediction of the Reynolds model is that if a transition between two scenes is highly unpredictable then this will correspond to an event boundary. Conversely, Schapiro and colleagues proposed that human event boundaries respect an alternate form of structure, graph community structure, that is unrelated to the predictability of scene transitions. They designed a task in which stimuli were drawn from a graph with multiple communities and, crucially, transitions between communities were equally probable as all other transitions. Thus, if people learn the underlying transition dynamics than transitions between transitions will not generate a larger prediction error crossing a community boundary than other transitions. They used a recurrent neural

network as a model and found that the network represented stimuli from the same graph community as more similar to each other, potentially signaling an event boundary between communities through a decrease in similarity. This corresponded to human-rated event boundaries, which tended to respect the graph community structure. In a third model of event dynamics, Elman and McRae (2019) used a simple recurrent network (Elman, 1990) with localist scene representations of the the scene to model event knowledge of individual events. This model was able to learn the temporal dynamics of events as well as co-occurrence statistics and predict sequential patterns of activities.

Learning

The process by which people learn event structure is not well understood, though several related lines of research offer some suggestive clues. One such line of research is statistical learning, originally proposed to explain how language learners come to delineate word boundaries in continuous speech through unsupervised observation (Aslin, Saffran, & Newport, 1998; Saffran, Aslin, & Newport, 1996). In a typical statistical learning task, subjects are presented with a sequences of objects and tested on successively or simultaneously presented objects without feedback. The probability distribution over sequences is potentially an important cue for learning latent temporal structure, and children and adult learners acquire this temporal structure across a wide range of linguistic and non-linguistic stimuli (Fiser & Aslin, 2001, 2002; Saffran et al., 1996; Saffran, Johnson, Aslin, & Newport, 1999). A drawback of these statistical learning tasks is that, while they are dynamic, the regularities embedded within them tend to be fairly unstructured (e.g., simple Markov processes on discrete symbols). In contrast, we would expect realistic events to contain a high degree of structure that may be unpredictable from low-level features (Richmond & Zacks, 2017).

We contrast statistical learning with relational learning, which tends to consider structure but not dynamic processes (Halford, Wilson, & Phillips, 1998; Hummel & Holyoak, 2003; Kemp, Tenenbaum, Niyogi, & Griffiths, 2010). Events are thought to

contain relational structure (Tversky, Zacks, & Hard, 2008), a feature that is often lacking in computational models. Bayesian models have been developed to explain relational structure learning (Kemp & Tenenbaum, 2008; Kemp et al., 2010), and we develop a variant of these models for learning event schemata.

Neural networks also provide convenient models for learning the types of sequential dependencies that constitute events. Typically, the sequential dependencies that constitute an event are modeled with recurrent networks (Elman, 1990; Hochreiter & Schmidhuber, 1997), which have strong theoretical guarantees in terms of what they can learn (Siegelmann & Sontag, 1991, but see Geman, Bienenstock, and Doursat 1992), and have been empirically effective in computational domains with long-term sequential dependencies, such as language (LeCun, Bengio, & Hinton, 2015) and reinforcement learning (Mnih et al., 2016). Because neural networks rely on distributed representations, they provide a basis for representing similarity though vector similarity (Hinton, McClelland, & Rumelhart, 1986), and are well suited for smooth generalization. These distributed representations can support symbolic structure via algebraic manipulation (Doumas & Hummel, 2005; Hummel & Holyoak, 2003; Plate, 1995; Smolensky, 1990), which allows them to encode the higher order symbolic processes the constitute events.

Inference

Event structure can inform inferences people make about parts of their environment they did not experience. As we move through our natural world, we do not always have immediate perceptual access to relevant features of our immediate surroundings. For example, if a person hears a door open behind them, they can make several inferences about the current scene that are relevant to the ongoing event, even though they do not have full perceptual access to the each feature.

Inferences about events have historically been investigated with reading and memory tasks in which a subject reads a narrative and either reading time (Altmann & Mirković, 2009; McKoon & Ratcliff, 1992) or memory (Bower, Black, & Turner, 1979;

Bransford, Barclay, & Franks, 1972) is used as a probe for their inferences about unstated aspects of the event. One of the main themes in the reading comprehension literature is that only a subset of inferences are made online, specifically those that are necessary for satisfying some notion of coherence in the text, although there is disagreement about what exactly this entails (Graesser, Singer, & Trabasso, 1994; McKoon & Ratcliff, 1992; Trabasso & Van Den Broek, 1985). In the constructionist account (Graesser et al., 1994), elements of the event model are used to fill in aspects of the event when they are likely to be relevant, and these inferred portions of the event are more quickly accessed and less surprising when they subsequently occur. This implies an averaging effect where inferences about the unstated aspects of an event reflect commonly experienced configurations. This is conceptually similar to adaptive statistical biases seen in other domains, such as the estimation of spatial location (Huttenlocher, Hedges, & Duncan, 1991).

Prediction

Event structure shapes predictions about the future. This can be seen in serial reaction time tasks, in which subjects responds to cues more quickly when they are generated from repeated, predictable patterns than when they are generated as a random sequence (Nissen & Bullemer, 1987). This form of prediction has been argued to be implicit (Reber, 1989; Robertson, 2007) and is consistent with earlier ideas that people learn 'scripts' to guide their actions (Lashley, 1951; Schank & Abelson, 1977).

There is also a long history of studying prediction in language comprehension. While reading, subjects fixate less often and for shorter durations on highly predictable words (Ehrlich & Rayner, 1981) and are slower to process unexpected words (Schwanenflugel & Shoben, 1985). Words completing sentences in a nonsensical way elicit an N400 event-related potential, thought to signify a surprising or unexpected stimulus (Kutas & Hillyard, 1984). Event knowledge specifically appears to influence language comprehension (McRae & Matsuki, 2009); individual words cue event-based knowledge (Altmann & Kamide, 1999; Hare, Elman, Tabaczynski, & McRae, 2009;

McRae, Hare, Elman, & Ferretti, 2005), and combinations of words narrow the scope of perceived events (Matsuki et al., 2011).

Neural measures also offer support for the proposal that people continually leverage sequential structure to make online predictions about upcoming experiences (Cohn, Jackendoff, Holcomb, & Kuperberg, 2014; Schiffer & Schubotz, 2011). These predictions are influenced by event structure. Behaviorally, people make better predictions within an event than across event boundaries (Zacks et al., 2011) and unpredictability across event boundaries is associated with prefrontal cortex, striatum, and hippocampus (Axmacher et al., 2010; Lisman & Grace, 2005; Ranganath & Rainer, 2003; Zacks et al., 2011). These findings are consistent with the hypothesis that failure in prediction is used to signal event boundaries (Reynolds et al., 2007; Zacks et al., 2007).

Memory

Memory can be both aided and impaired by knowledge of event structure. Event boundaries induce selective trade-offs in memory that depend on the exact study design and memory measure. Sequential recall and temporal order memory are worse across event boundaries than across boundaries (DuBrow & Davachi, 2013, 2016; Heusser, Ezzyat, Shiff, & Davachi, 2018), but memory for specific items has been found to be higher at event boundaries (Heusser et al., 2018) and the presence of an event boundary can increase overall recall (Pettijohn, Thompson, Tamplin, Krawietz, & Radvansky, 2016). Moreover, individuals with better event segmentation perform better on subsequent memory tests (Sargent et al., 2013; Zacks, Speer, Vettel, & Jacoby, 2006). An ongoing event appears to have a privileged role in memory as well, as memory for items within an ongoing event is often better than immediately following an event boundary in a way that is not recovered by returning to the original context (Radvansky & Copeland, 2006; Radvansky, Krawietz, & Tamplin, 2011).

The *Event Horizon Model* (Radvansky, 2012; Radvansky & Zacks, 2014) offers a conceptual explanation for these findings. According to this model, people track the

causal structure of events and use this structure to aid memory retrieval. This causal structure leads to better memory for items overall but can lead to memory interference under certain conditions, such as when recall depends on retrieval of a single event model and the presence of multiple events can introduce noise. The Event Horizon Model further hypothesizes that working memory maintains privileged access to the current model, which thus results in better memory retrieval. According to this account, sequential recall might be impaired across event boundaries because it relies on two competitive event models, whereas overall recall would be improved because it is non-competitive.

Event knowledge has also been implicated in false memory paradigms (Bower et al., 1979; Bransford et al., 1972). In these paradigms, subjects report remembering unstated details of a story that were nonetheless consistent with the narrative. This effect is parametric, such that more experiences with similar stories increases script-consistent false memories (Bower et al., 1979). This suggests that people use event knowledge to fill in the gaps of their memory and are consistent with 'script theory' in which people organize sequential processes in terms of an abstract schema, or script, that organizes perception and influences memory (Schank & Abelson, 1977). Conceptually, this is similar to the idea of 'pattern completion', where a partial memory trace is reconstructed with reference to learned pattern of activity (McClelland, McNaughton, & O'reilly, 1995; Norman & O'Reilly, 2003), and may be adaptive if it facilitates inference about unobserved aspects of an event.

Limitations of existing theoretical accounts

Most previous theoretical accounts of event cognition have been non-computational (Radvansky, 2012; Radvansky & Zacks, 2014; Zacks et al., 2007). While these accounts provide valuable theoretical insight into event perception and memory, they do not offer the same level of detailed prediction as a computational model. Computational accounts of events have typically focused either on providing a normative account without considering experimental data, or have provided an

explanation for a specific set of phenomena. Several models have focused on learning event dynamics and detecting boundaries as a statistical property of the stimuli. Reynolds et al. (2007) proposed a recurrent neural network model to account for event segmentation, arguing that prediction error could be used in learning event dynamics and arbitrating between event models. While their model was able to develop useful event representations in a 3-d motion capture dataset, it was not directly compared to human behavior. In contrast, the model proposed by Schapiro et al. (2013) was used to account for their empirical finding that humans are sensitive to community structure when delineating event boundaries, even when transition uncertainty is matched when staying within vs. transitioning across communities.

Neither of these models incorporate structured event representations, limiting their ability to explain the data on script memory and text comprehension. Furthermore, it has been argued that structured event representations are simpler to learn and generalize (Goodman, Ullman, & Tenenbaum, 2011; Kemp & Tenenbaum, 2008; Richmond & Zacks, 2017). The recent model developed by Elman and McRae (2019) avoids structured representation by using localist representations within a recurrent neural network model of events. Their model is able to generalize (infer) the co-occurrence of different fillers in their appropriate roles and shows generalization between events. Nonetheless, the representation used by Elman and McRae (2019) is problematic. Propositional roles were encoded with separate vectors in the input space, and specific fillers were encoded with one-hot representations in this vector space. This is a representationally greedy embedding space: each encoded item increases the dimensionality of the space by one and is completely orthogonal to every other item. Such a representation is unlikely to scale to naturalistic datasets, as learning the correlational structure needed to generalize is subject to the curse of dimensionality.

In all of these computational models, events are modeled as a dynamical process that unfolds over time. To our knowledge, how these processes interact with memory has not been addressed with computational modeling. The *Temporal Context Model* (Howard & Kahana, 2002) and the related *Context Maintenance and Retrieval* model

(Polyn, Norman, & Kahana, 2009) treat memory as a dynamical process in which an evolving context induces a temporal organization in memory. These models suggest dynamics similar to what we would expect when learning events but do not provide a mechanism for partitioning events or generating predictions of the future. Related models of memory support some forms of inference (e.g., Nelson & Shiffrin, 2013; Shiffrin & Steyvers, 1997), but these models tend not to incorporate the dynamical processes that define events.

Finally, no prior computational model of event cognition has attempted to explain naturalistic data (e.g., real videos). While the use of naturalistic data is not a theoretical challenge to any prior model of event segmentation, it is nonetheless an important practical consideration. In order to validate our computational theories of cognition, we should strive to show that they scale to real-world problems. Otherwise, it is not clear whether our models of cognition work only when constrained to artificial tasks. More broadly, while several computational models have addressed parts of event cognition; to our knowledge, no model has attempted to address all aspects of the problem.

The Structured Event Memory model

We propose a model, *Structured Event Memory* (SEM), that systematically addresses the 5 desiderata for understanding event cognition (segmentation, learning, inference, prediction, memory), overcoming the limitations of prior models.

Our model is a computational-level analysis of event segmentation and memory that frames these tasks in terms of probabilistic reasoning. At a high level, our model assumes that people simultaneously segment 'scenes' into events. By scenes, we mean a description of the environment containing a relevant set of objects and the relations between them. For example, for a person watching a movie, a scene might be the experience of sitting in a dark room eating popcorn, or it might describe the ongoing actions of the movie, the characters involved, and its setting. In the context of a psychological study, a scene might describe stimuli currently on a screen and the context

of sitting in front of a computer. Each individual event token is assumed to be unique to a sequence of contiguous scenes, and we assume each event token is associated with a generalizable event type that describes the dynamics of all tokens that share a type. We use a nonparametric prior over event segmentations that allows both the number of types and the boundaries between tokens to be inferred based on the scene inputs.

SEM simultaneously learns the sequential dependencies over scenes for each event type (analogous to a 'script' or 'event schema') for the purpose of generalizing these dynamics to new instances of the event type. Due to the challenge of generalizing dynamics over logical expressions, and in order to facilitate the neural plausibility of the model, we embed the logical descriptions in a vector space using holographic algebra (Plate, 1995). The event dynamics can then be parametrized as continuous functions of these vectors, and the parameters can be learned using gradient descent.

Finally, we assume that the memory encoding and retrieval process loses information about the features of the scene. Inferred event structure is combined (via Bayes' rule) with a noisy memory trace to reconstruct past scenes. This gives rise to effects of event semantics and boundaries on memory performance.

A prior distribution over events

We make the following assumptions about events:

1. Humans do not know *a priori* how many distinct event types there are; this must be discovered from the data.

2. Humans attempt to reuse previously learned event models whenever possible. This constitutes a "simplicity" bias in the sense of Ockham's razor. It facilitates generalization between events, a key computational goal.

3. Events are temporally persistent, such that there is a high probability that any two consecutive moments in time are in the same event.

4. Events define dynamical systems, generating predictions of successor scenes as a function of the current scene.

5. Events have latent structure: each instance of an event (and the scenes within events) may be unique with respect to specific percepts but nonetheless shares a latent structure (such as the relationships between objects) with similar events.

6. Events are used in memory retrieval to regularize a noisy memory trace. By this we mean that event knowledge can compensate for missing information in a memory trace by introducing knowledge of a typical event in a reconstruction process.

Assumptions 1, 2 and 3 relate to the process of assigning each moment in time to an individual event. We can satisfy these assumptions using a simple Bayesian nonparametric process known as the sticky Chinese restaurant process (CRP).[1]

The sticky CRP sequentially assigns time points to events according to past event frequency (higher frequency events are more likely to be repeated) and recency (the most recent event is more likely to be repeated) while maintaining some probability that a new event will be generated at each moment in time (cf. assumption 1). Formally, at time n the next event is drawn from the following distribution:

$$\Pr(e_n = k|e_{1:n-1}) \propto \begin{cases} C_k + \lambda \mathbb{I}[e_{n-1} = k] & \text{if } k \leq K \\ \alpha & \text{if } k = K + 1 \end{cases} \tag{1}$$

where e_n is the event assignment, K is the number of distinct event types in $e_{1:n-1}$, $\mathrm{I}[\cdot] = 1$ if its argument is true (0 otherwise), and C_k is the number of previous timepoints assigned to event k. The concentration parameter $\alpha > 0$ determines the simplicity bias (cf. assumption 2); smaller values of α favor fewer distinct events (Figure 1). The stickiness parameter $\lambda \geq 0$ determines the degree of temporal autocorrelation (cf. assumption 3); higher values of λ favor stronger autocorrelation.

The remaining assumptions relate to the nature of event schemata (assumptions 4 and 5) and memory retrieval (assumption 6). In the following sections, we discuss how these assumptions can be instantiated in a computational model. First, we discuss the representational space in which event models, as dynamical systems, operate (cf.

[1] For an explanation of this culinary metaphor, see Gershman and Blei (2012).

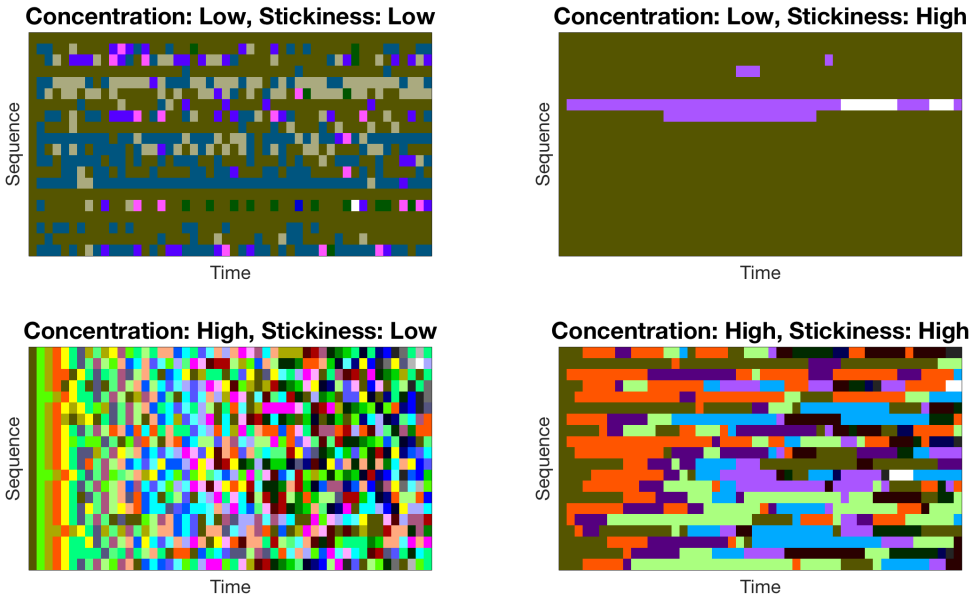

Figure 1. **Samples from the generative process of events under different parameter regimes.** Each row in each panel is a single draw from the process across time. Colors indicate when different event schemata are active. High values of the concentration parameter α (bottom row) leads to more unique events. High values of the stickiness parameter λ (right column) leads to longer event durations.

assumption 4). Representation is critical for generalization, and to that end, we discuss a structured representation in vector space that facilitates generalization via continuous functions. We then return to our model to discuss event dynamics and how they are learned, how event segmentation occurs in the model, and finally, we present our model of event memory.

Scene representations

We assume that event schemata define dynamical processes over scenes, in which the event model is used to generate a prediction about the next scene given the recent history of scenes. Formally, we define a function that takes in a scene $s \in \mathcal{S}$ and returns a successor scene s'. We further assume that for each scene s, there exists a distributed (vector) representation $\mathbf{x} \in \mathbb{R}^d$. Whereas a scene can be thought of a ground-truth description of the external world, $\mathbf{x}$ can be thought of as a representation of the features

of s relevant to an agent.

A core assumption is that the vector representation of the scene is distributed and encodes features in a similarity space (Goodfellow, Bengio, & Courville, 2016). For example if "cat" and "dog" have meaningful representations, they may share a number of features encoded in the space, such as "isSmall" and "isFurry". As pure symbols, "cat" and "dog" are as distinct from each other as any other pair of symbols. Here, we assume the features of each term in this similarity space are linearly composable with vector addition, a property common in distributed systems and learnable with unsupervised training (Mikolov, Sutskever, Chen, Corrado, & Dean, 2013; Radford, Metz, & Chintala, 2015). As a consequence of this similarity space, similar scenes have similar vectors such that the vector representation of "dog" is close in Euclidean space to the vector representation of "cat".

A primary motivation of using distributed representations is that they facilitate smooth generalization. As we will discuss in the next section, defining scenes in vector space allows us to parametrize event dynamics over arbitrary scenes. We will assume that these event dynamics are smooth, such that if f is a function that represents the event dynamics and $\mathbf{x} \approx \mathbf{y}$, then f will generally have the property $f(\mathbf{x}) \approx f(\mathbf{y})$ (Goodfellow et al., 2016). An embedding space that encodes similar scenes with similar vectors will facilitate this type of generalization naturally with parameterized functions; by contrast, were we to use a tabular representation, we would have to define transitions over an intractably large discrete space that does not permit smooth generalization.

Distributed representations have the further advantage that they are representationally compact. A purely symbolic representation encodes a dimension for each binary feature, whereas a distributed representation can take advantage of the correlational structure to encode scenes in a lower-dimensional space (Goodfellow et al., 2016). This is related to dimensionality reduction, in which a representation is projected into a low-dimensional space that preserves relevant features and discards irrelevant features. This property is important for computational reasons, as it is computationally intractable to estimate dynamical systems in high dimensional spaces

(Bellman, 1961) (such as the raw perceptual space) and event dynamics are thought to be smoother in higher-level (i.e. low-dimensional) representations (Richmond & Zacks, 2017). As a practical matter, we need a principled way to construct a low-dimensional scene representation to model naturalistic data sets. Later in the paper we will present simulations with video data and show that convolutional neural networks are well suited to this task for visual domains (Fukushima, 1980; Kietzmann, McClure, & Kriegeskorte, 2018; LeCun et al., 2015). Representational spaces as described above can be learned in visual data with unsupervised methods (Hou, Shen, Sun, & Qiu, 2017; Radford et al., 2015). In our simulations, we use a variational autoencoder (VAE; Kingma & Welling, 2013), an unsupervised convolutional neural network, to learn the representational space (see Appendix A for detail). We note that while the representation of scenes is important for our theoretical model, we are agnostic to the specific details by which it is learned in the brain.

In addition to these unstructured features of scenes, we further assume that scenes encode relational structure. By this we mean that scenes contain relational information about how particular fillers are bound to particular roles (Radvansky & Zacks, 2011). For example, if a person is holding a phone, the scene would contain not just a reference to the objects "person" and "phone", but also a specific binding between the objects and their roles in the relationship "holding". A faithful vector representation of structured scenes must therefore also contain this structure when relevant. Formally, this requires variable binding, a prerequisite for symbolic computation. There are several ways variable binding can be implemented in vector spaces, such as using tensor products (Smolensky, 1990), holographic reduced representations (Plate, 1995), or binding by synchrony (Hummel & Holyoak, 2003).

In this paper, we will focus on holographic reduced representations (HRRs), which use circular convolution (a compressed form of tensor product) as the binding operator, and vector addition as the conjunction operator. This produces vectors of a fixed dimensionality (unlike tensor products) regardless of the complexity of the underlying formula. Concretely, consider the expression "dog chases cat". This takes the logical

form "chase(dog, cat)", where dog occupies the agent role and cat occupies the patient role. If we have vectors representing both roles and fillers (where our fillers are "dog", "cat" and "chase" and our roles are "agent", "patient" and "verb"), then the scene vector is constructed according to:

$$\mathbf{x} = \text{dog} \circledast \text{agent} + \text{chase} \circledast \text{verb} + \text{cat} \circledast \text{patient}$$

where $\circledast$ denotes circular convolution. The underlying components can be approximately decoded using circular correlation, a simple algebraic operation, directly from the composed scene vector (see Plate, 1995, for details).

The HRR is convenient because both the encoding operation and approximate decoding operation can be accomplished by efficient algebraic manipulations. Furthermore, the operations of HRRs preserve the similarity of the composed terms (see Appendix B for detail), meaning that structural properties can be encoded as features of the embedding space. For example, we can decompose a term as a linear combination of structured and unstructured features, such as

$$\text{cat} = \text{acceptAgent} + \text{acceptPatient} + \text{isSmall} + \text{isFurry} + ...$$

where the features "acceptAgent", "acceptPatient" correspond to valid relational roles "isSmall", "isFurry" are unstructured features. If the term "dog" corresponds to a similar vector with shared composed features, than a scene composed with "dog" will be similar to a scene composed with "cat". Concretely, the scene "chase(cat, mouse)" will be close in vector space to the scene "chase(dog, mouse)" because of the structure similarity of the fillers. As noted above, this facilitates smooth generalization of event dynamics by parameterized functions.

Scene dynamics

As previously noted, we assume that event schemata are stochastic dynamical processes that takes in a scene s and output a successor scene s' with probability $\Pr(s_{n+1}|s_n, e)$, where s_n is the scene at time index n and $e \in \mathbf{Z}$ is an event label. In SEM, we assume this probability distribution is defined with a smooth function f over

the embedded scenes and parameterized by θ, such that

$$\Pr(\mathbf{x}_{n+1}|\mathbf{x}_{1:n}, e, \theta) = \mathcal{N}(\mathbf{x}_{n+1}; f(\mathbf{x}_{1:n}, e, \theta), \beta \mathrm{I}) \tag{2}$$

where $\mathbf{I}$ is the identity matrix, and $\mathbf{x}_{n+1}$ and $\mathbf{x}_{1:n}$ are the vector embedding for the successor scene s_{n+1} and all of the previous scenes, respectively. Equivalently, we are assuming that the position of the successor scene in vector space is defined by the function f, such that

$$\mathbf{x}_{n+1} = f(\mathbf{x}_{1:n}, e, \theta) + \epsilon \tag{3}$$

where ϵ is zero-mean Gaussian noise.

We are agnostic to the shape of the function f and only require that θ is learnable from observation. In our simulations, we model f as a recurrent neural network (Figure 2). Except where noted, we used a fully connected, four-layer network with gated recurrent units (GRUs; Cho, Van Merriënboer, Bahdanau, & Bengio, 2014) with a leaky rectified linear activation function ($\alpha = 0.3$; Maas, Hannun, & Ng, 2013) as a non-linearity and 50% dropout for regularization (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014). A second, non-recurrent hidden layer with a linear output is used to produce the output. The networks were implemented in Keras (Chollet et al., 2015) and were trained with batch updates of cached observations using the Adam stochastic optimization algorithm (Kingma & Ba, 2014). Parameter values for this optimization are listed in table D2. We chose a recurrent network so that the network would be sufficiently flexible to learn the event dynamics. Previous theoretically modeling suggest event dynamics can represented with recurrent neural networks (Elman & McRae, 2019; Reynolds et al., 2007; Schapiro et al., 2013), and similar recurrent neural networks are thought to be biologically plausible (O'Reilly & Frank, 2006). Furthermore, Reynolds et al. (2007) found that recurrence improved learning event dynamics when compared to otherwise identical feed-forward networks. Nonetheless, we do not believe that any of these specific implementation details are critically necessary, and other variants may make similar predictions.

The noise over scene transitions in equation 2 is assumed to be Gaussian with a diagonal covariance matrix, such that the covariance matrix is the product of the

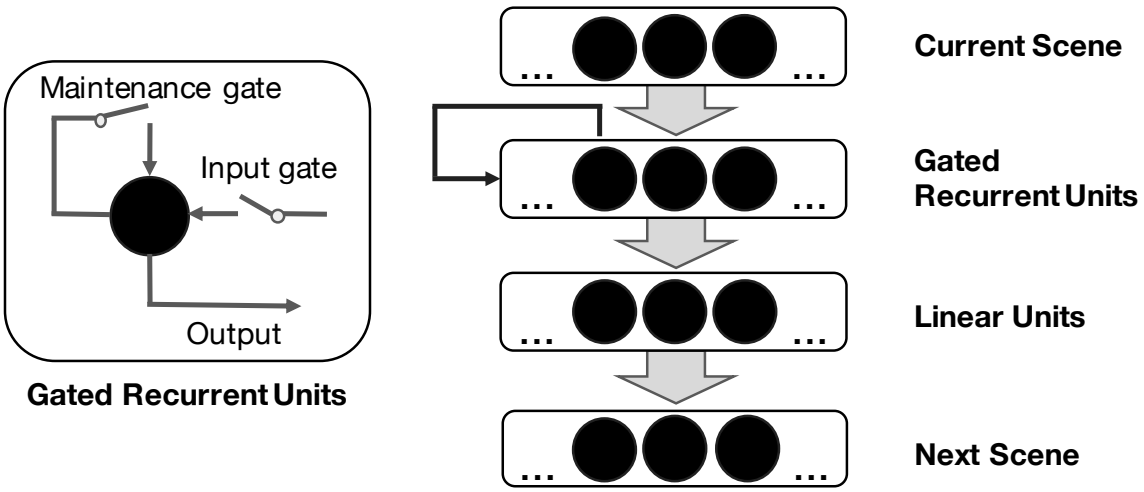

Figure 2. **Event Dynamic Neural Network** A four-layer network took in the current scene as the input and predicted the next scene as the output. The network has two hidden layers: one with gated recurrent units (GRU; Cho et al., 2014) and a leaky rectified linear output and second layer of fully connected, linear units. GRUs provide recursion by maintaining an internal state of the network. This state is controlled by two gates, an 'input' gate and a 'maintenance' gate.

identity matrix and the vector of noise parameters $\beta \in \mathbb{R}_+^d$. This is done for mathematical convenience, as estimating a full covariance function is a difficult problem with limited data. The vector β is estimated using maximum *a posteriori* estimation, assuming an inverse-χ^2 prior parametrized by ν degrees of freedom and a scale of s^2 (Gelman et al., 2013).

We separately define an initial condition f_0 for the function f, which is estimated from the data with a uniform prior over f_0. The transition probability for scene s_{t+1} following an event boundary is thus:

$$\Pr(s_{n+1}|s_n, e_{n+1} \neq e_n) = \int \mathcal{N}(\mathbf{x}_{n+1}; f_0, \beta \mathrm{I}) \Pr(f_0) df_0 \tag{4}$$

Importantly, this initial condition probability is different for experienced and novel events. The prior over f_0 is important for novel events, as it allows us to define a scene transition probability for an unseen event by integrating over the prior. When the event

e_{n+1} is a previously unseen event, e_{new}, the transition probability in equation 4 reduces to a constant. For experienced events, we simplify this probability function by ignoring the prior $\Pr(f_0)$ and instead use a point estimate of f_0. As we discuss in the following section, we use the probability $\Pr(s_{n+1}|s_n, e_{new})$ to infer the event boundaries and therefore require a definition of this term for unseen events.

Event segmentation

Having defined the generative process for events, the representational space for scenes, and the scene dynamics, we can now pose questions for the computational model. A primary challenge for the model presented above is how it can learn and segment events without an external training signal. In terms of statistical estimation, event segmentation is an unsupervised learning problem. A key claim of the model is that it can learn event dynamics while simultaneously segmenting scenes into events. In order to solve both of these problems simultaneously, we perform inference over the generative model.

As we assume that events are not directly observable, their identity and boundaries must be inferred (segmented). Given a history of scenes $\mathbf{s} = \{s_n\}_{n=1}^N$, Bayes' rule stipulates the posterior over events $\mathbf{e} = \{e_n\}_{n=1}^N$ is:

$$\Pr(\mathbf{e}|\mathbf{s}, \theta) = \frac{1}{Z} \Pr(\mathbf{s}|\mathbf{e}, \theta) \Pr(\mathbf{e}) \tag{5}$$

where $\Pr(\mathbf{s}|\mathbf{e}, \theta)$ is the likelihood of the scene history under a hypothetical event segmentation, and $\Pr(\mathbf{e})$ is the prior probability of the event sequence (i.e., the generative process for events, Eqn. 1) and Z is the normalizing constant. The likelihood can be decomposed according to:

$$\Pr(\mathbf{s}|\mathbf{e}, \theta) = \prod_{n=1}^{N} \Pr(s_{n+1}|s_n, e_n, \theta) \tag{6}$$

where $\Pr(s_{n+1}|s_n, e_n, \theta) = \Pr(\mathbf{x}_{n+1}|\mathbf{x}_{1:n}, e, \theta)$ is given by Eq. 2. The likelihood can be viewed as a way of encoding (inverse) "prediction error", which has been suggested as a key determinant of event segmentation (Reynolds et al., 2007; Zacks et al., 2011). A low likelihood indicates a high prediction error, favoring the inference of an event boundary.

To make this point more explicit, we can express the log-likelihood for a single transition $s_n \rightarrow s_{n+1}$ as follows:

$$\log \Pr(s_n | s_{n+1}, e, \theta) = -\frac{1}{2\beta} ||\mathbf{x}_{n+1} - f(\mathbf{x}_{1:n}; e, \theta)|| + \text{const.} \tag{7}$$

Thus, the log probability is inversely proportional to the prediction error.

In principle, event segmentation requires inference over the intractably large discrete combinatorial space of partitions. To comply with the cognitive constraint that inference is carried out online (i.e., without re-segmenting past experiences), we employ a "local" maximum a posteriori (MAP) approximation (Anderson, 1991; Gershman, Radulescu, Norman, & Niv, 2014):

$$\Pr(e_{n+1} | \mathbf{s}_{1:n}, \theta) = \sum_{\mathbf{e}_{1:n-1}} \Pr(e_n | \mathbf{s}_{1:n}, \mathbf{e}_{1:n-1}, \theta) \tag{8}$$

$$\approx \Pr(e_n | \mathbf{s}_{1:n}, \hat{\mathbf{e}}_{1:n-1}, \theta) \tag{9}$$

where $\mathbf{s}_{1:n}$ denotes the sequence of scenes observed from time 1 to n and $\hat{\mathbf{e}}_{1:n-1}$ is a point estimate of the prior event segmentation defined recursively as follows:

$$\hat{e}_n = \underset{e_n}{\text{argmax}} \Pr(e_n | \mathbf{s}_{1:n}, \hat{\mathbf{e}}_{1:n-1}, \theta) \tag{10}$$

In other words, we approximate the intractable summation over $\mathbf{e}_{1:n}$ with a single high probability hypothesis about the prior segmentation.

Event Memory

We now turn our attention from the problem of event segmentation to consider the encoding of items into memory. We first define the generative process of encoding items into memory and then return to discuss how event dynamics can be used to improve memory in a reconstructive process. Consider the paired sequences of embedded scenes $\mathbf{x} = \{\mathbf{x}_1, .., \mathbf{x}_N\}$ and events $\mathbf{e} = \{e_1, ..., e_N\}$ that define the world dynamics of our generative model. Each scene vector $\mathbf{x}_n \in \mathbb{R}^d$ is a real-valued vector encoding the features of the scene at time n, while $e_n \in \mathbb{Z}$ is a label corresponding to the event model at the same time.

Implicitly, both $\mathbf{x}$ and $\mathbf{e}$ encode time via position because they are ordered sequences. In order to make this representation explicit, we define an unordered set of memory 'items' $y = \{y_1, ..., y_N\}$ that have a one-to-one correspondence to the scene vectors $\mathbf{x}$ and events $\mathbf{e}$. Each element y_i is defined as a 3-tuple of the features of the scene, the event label, and its time index, such that $y_i = (\mathbf{x}', e', n)$ where $\mathbf{x}'$ is the vector of features, e' is the event label and n is the time index. Equivalently, $\mathbf{x}' = \mathbf{x}_n$ and $e' = e_n$ for the scene $y_i = (\mathbf{x}', e', n)$.

We assume memory is a lossy encoding and retrieval process such that all of the components of the memory items are corrupted. Specifically, we assume an encoding process $\Pr(\tilde{y}|y)$ creating the corrupted (encoded) memory trace $\tilde{y}_i = (\tilde{\mathbf{x}}', \tilde{e}', \tilde{n})$ where $\tilde{\mathbf{x}}' = [\tilde{x}_1, ..., \tilde{x}_d]$, $\tilde{e}'$ and $\tilde{n}$ corresponds to the corrupted memory traces of the scene features, event label, and time index, respectively. The assumption that memory traces are corrupted versions of an original stimulus is common in computational models of memory (Hemmer & Steyvers, 2009; Huttenlocher et al., 1991; Shiffrin & Steyvers, 1997) and is analogous to a capacity-limited compression (Brady, Konkle, & Alvarez, 2009; Nassar & Frank, 2016).

For convenience, we will assume that the corruption process for each component of each scene vector is independent, such that

$$\Pr(\tilde{y}|y) = \Pr(\tilde{\mathbf{x}}|\mathbf{x}) \Pr(\tilde{\mathbf{e}}|\mathbf{e}) \Pr(\tilde{n}|n). \tag{11}$$

We assume Gaussian noise over the features, such that

$$\Pr(\tilde{\mathbf{x}}|\mathbf{x}) = \mathcal{N}(\tilde{\mathbf{x}}; \mathbf{x}, \tau \mathrm{I}) \tag{12}$$

where the parameter τ corresponds to the degree of corruption noise of the feature. For event tokens, we assume that the corruption process is an asymmetric channel similar to a Z-channel (MacKay, 2003), such that the event token is either correctly encoded or that the event label is completely lost:

$$\Pr(\tilde{e}|e) = \begin{cases} \epsilon_e & \text{if } \tilde{e} = e \\ 1 - \epsilon_e & \text{if } \tilde{e} = e_0 \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

where e_0 is a 'null' event label corresponding to no event model (representing a loss of the event label), and where the parameter ϵ_e defines the probability of retaining the event label in memory. Thus, when $\epsilon_1 = 0.0$, the event label is completely lost from the memory trace. Corruption noise over the time index n is defined as a discrete uniform over the interval $[n - b, n + b]$, such that

$$\Pr(\tilde{n}|n) \sim \mathrm{U}[n - b, n + b]. \tag{14}$$

We will typically assume small values of b, allowing for items in memory to be flipped, but not allowing large jumps in time. Taken with the corruption of event labels, time index corruption has several consequences, including the corruption of the relative order of items within each event and the corruption of the (implicitly represented) boundary location. We note that while we have committed to these independent corruption processes, other forms of information loss in memory are plausible, including event-label switching and non-uniform time corruption among others. The full memory trace over n scenes is defined as the unordered set of corrupted traces $\tilde{y} = \{\tilde{y}_1, ..., \tilde{y}_N\}$.

Reconstruction. Given a memory trace $\tilde{y}$ and learned event dynamics (f, θ, e), the task of the memory model is to reconstruct y. We note that this is different from many models of memory, in that we assume a corrupted memory trace and ask how event dynamics can be used to denoise this trace, whereas many memory models are concerned with the process by which a memory trace is generated (e.g., Norman & O'Reilly, 2003). To reconstruct y, we define the generative process of the corrupted memory traces and do inference over this process to reconstruct the original items prior to encoding. Knowledge of the event dynamics is helpful in memory reconstruction because they are assumed to be the generative process of the original scenes. As such, the event dynamics can regularize the reconstructive memory process. A similar approach by Huttenlocher and colleagues (1991) was used to model human memory judgments of the spatial location of dots. They argued that a category prior can be used to reduce the variance of a corrupted memory trace. Here, we argue that event dynamics occupy the same role, reducing variance of the corrupted memory trace by regularizing (that is, adding an adaptive statistical bias).

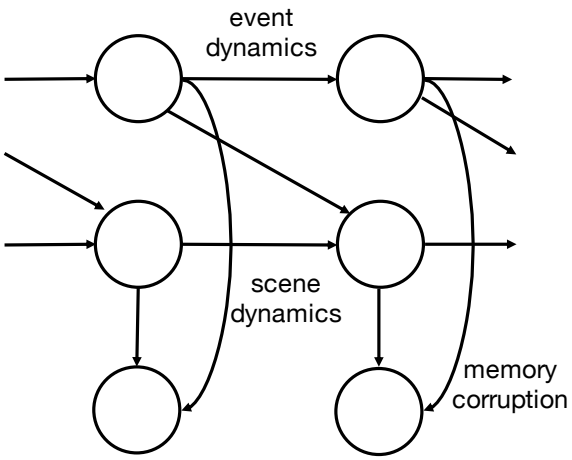

Figure 3. **Memory corruption.** A lossy memory encoding procedure stores the features of the original scene $\mathbf{x}$, its time index n and event index e in corrupted form $\tilde{y} = (\tilde{\mathbf{x}}, \tilde{e}, \tilde{n})$. Memory reconstruction inverts this generative process to infer the original memory item.

We first consider the generative process of $\tilde{y}_i$ (and its corrupted features $\tilde{\mathbf{x}}_i$ and time index $\tilde{n}$). This is defined:

$$\Pr(\tilde{y}_i|f,\theta) = \sum_{\mathbf{e}} \int_{\mathbf{x}} \Pr(\tilde{y}_i|\mathbf{x},\mathbf{e}) \Pr(\mathbf{x}|\mathbf{e},f,\theta) \Pr(\mathbf{e}) d\mathbf{x} \tag{15}$$

where $\mathbf{x} = \mathbf{x}_{1:n}$ and $\mathbf{e} = e_{1:n}$ are the sequences of scenes and event labels, respectively. The three probability distributions, $\Pr(\tilde{y}_i|\mathbf{x},\mathbf{e})$, $\Pr(\mathbf{x}|\mathbf{e},f,\theta)$, and $\Pr(\mathbf{e})$ correspond to the encoding process, transition dynamics and prior over events, respectively.

The goal of memory retrieval is to estimate the original scenes $\hat{y}$ using a reconstruction process over the generative model (equation 15). Because the posterior has no closed form expression, we employ Gibbs sampling to draw a sample of reconstructed memory traces. A complete description of our Gibbs sampling algorithm is detailed in Appendix C, but at a high level, Gibbs sampling takes advantage of the conditional independence properties of the generative model, which can be seen in the graph structure (Figure 3). At each point in time, the generative process for the memory trace, the dynamics over scenes and the dynamics over events can be expressed with three conditionally independent functions (equations 11, 2, and 1, respectively). As such, we can draw samples of the memory traces, reconstructed scenes and event

labels one variable at a time by conditioning on the other variables in the process.

To capture the possibility of memory failure, we augment the set of corrupted memory traces $\{\tilde{y}\}$ with a 'null' memory y_0 and define a special case of the corruption process conditioned on a time index

$$\Pr(y_0|y) \propto \epsilon \tag{16}$$

where ϵ is a free parameter that controls forgetting in the model, as we discuss below. This choice of corruption process is convenient, as it has the interpretation of integration over memory items, $\int_y \Pr(\tilde{y}_i|y_i)dy \propto \epsilon$ for any arbitrary value of ϵ.[2] We assume that y_0 can be repeatedly sampled, as it is does not correspond to a specific memory trace, but the absence of one. Finally, we also assume that the learned event models and parameters (f, θ) are known because they have already been learned.

Simulations

The parameter values for all of the following simulations are listed in table D2 in the appendix. The code for all simulations is available in our Github repository: `https://github.com/ProjectSEM/SEM`.

Human-like segmentation on naturalistic stimuli

A key test of the model is whether it generates human-like segmentation on naturalistic tasks. Operationally, event boundaries in human studies are often defined by having subjects mark them in a naturalistic dataset, for example, while watching a video (Baldassano et al., 2017; Hanson & Hirst, 1989; Newtson & Engquist, 1976; Zacks et al., 2006, 2001). Historically, this has posed a challenge for computational models due to the difficulty of dealing with unannotated raw video data. The computational model proposed by Reynolds et al. (2007) attempted to circumvent this problem by

[2] Implicitly, this corruption process assumes a uniform prior over features, $\Pr(\mathbf{x}) \propto 1$. Alternatively, we could assume a generative process $\mathbf{x} \sim \mathcal{N}(0, \epsilon \mathrm{I})$, but this will lead to a similar result so long as the norm of corrupted memory traces are similar, and ϵ would likewise parameterize the degree of forgetting in the model for reasons discussed below.

using a carefully collected, low-dimensional motion capture dataset. Reynolds et al. (2007) evaluated their recurrent neural network model of event processing on a set of motion capture timeseries, each of which contained 18 points in a 3-dimensional coordinate space measured across time with a 3Hz sampling rate. While this model was able to provide valuable theoretical insights, such as the feasibility of updating event models with prediction error, it is difficult to fully validate the model without comparing the model directly to human data.

However, because of the rapid improvement in computer vision in recent years (LeCun et al., 2015), this computational issue can now be tackled; we can directly evaluate SEM on the same naturalistic dataset used in human studies. In the following simulations, we evaluate SEM on three video datasets previously used in a pair of studies by Zacks et al. (2006, 2001) to probe human event segmentation. We use the model to generate event boundaries and compare its predictions to human behavior reported in Zacks et al. (2006). We do so with fully unsupervised training, estimating an unstructured scene representation with a variational auto-encoder (see Appendix A for details). Consequently, in a video of a person making a bed, the model is not told what a person is or a bed is but has access only to what can be learned from pixel data. We are not claiming that people only use unstructured information while performing this task, but we nonetheless believe it is important to validate the model's performance in an end-to-end, fully unsupervised process on naturalistic data to argue that SEM scales up. We will return to the role structure plays in segmentation with later simulations.

Results. The stimulus set in Zacks et al. (2006, 2001) consists of five videos of a single person completing an everyday task, such as washing dishes, shot from a single, fixed camera angle with no edits. At a resolution of 240x320x3 pixels, each frame has more than 230,000 dimensions, and we used a variational auto-encoder to generate an unstructured scene representation of 100 dimensions. We evaluated the model on each of the three videos used as stimuli in experiment 1 of Zacks et al. (2006). The model was trained on each video independently with the parameter values listed in table D1. Figure 4 shows a qualitative comparison of human and model behavior in the "washing

dishes" video as well as a quantitative comparison over all three videos. Both human and model event boundaries have been binned in 1 second intervals, as was reported in Zacks et al. (2006). Qualitatively, there is good agreement between the model's MAP estimates of boundaries and the population of human subjects, with several of the major peaks in the group data corresponding to a model boundary. We assess this quantitatively with the point-biserial correlation between the model's MAP boundary estimates and the grouped subject data. Point-biserial correlation is a similar metric to the one used by Zacks et al. (2006) to asses the segmentation of each individual subject. The point-biserial correlation of the model was 0.143, both highly significant under permutation testing (95% CI of permutation test: [-0.068, 0.034]) and well within the range of observed human values (range: [0.054, 0.412], middle 75% of observations: [0.139, 0.400]). As the model produces a boundary probability at each time step, we also compared the model's log boundary probability to the aggregated human data directly, which was significant ($r^2 = 0.067$, 95% CI of permutation test: [0.000, 0.005]).

Event boundaries and community structure

While naturalistic tasks provide ecological validity, in the simulations above it is difficult to tell which features of the dataset people and the model rely on to segment events. Thus, while we can make descriptive comparisons, the video dataset does not provide a particularly diagnostic test of the model. A stricter test of the model is to compare human and model segmentation on specific features that can potentially drive segmentation.

One such principle that has previously been identified is prediction error (Reynolds et al., 2007). Humans show a decrease in predictive accuracy across an event boundary in naturalistic tasks (Zacks et al., 2011) and generally respect statistical structure (Avrahami & Kareev, 1994; Baldwin et al., 2008). Moreover, surprising occurrences in human tasks influence event perception (Newtson, 1973), suggesting that the predictability of scenes is important for segmentation. Given the formulation of event segmentation in our model as probabilistic inference (equation 5), the property

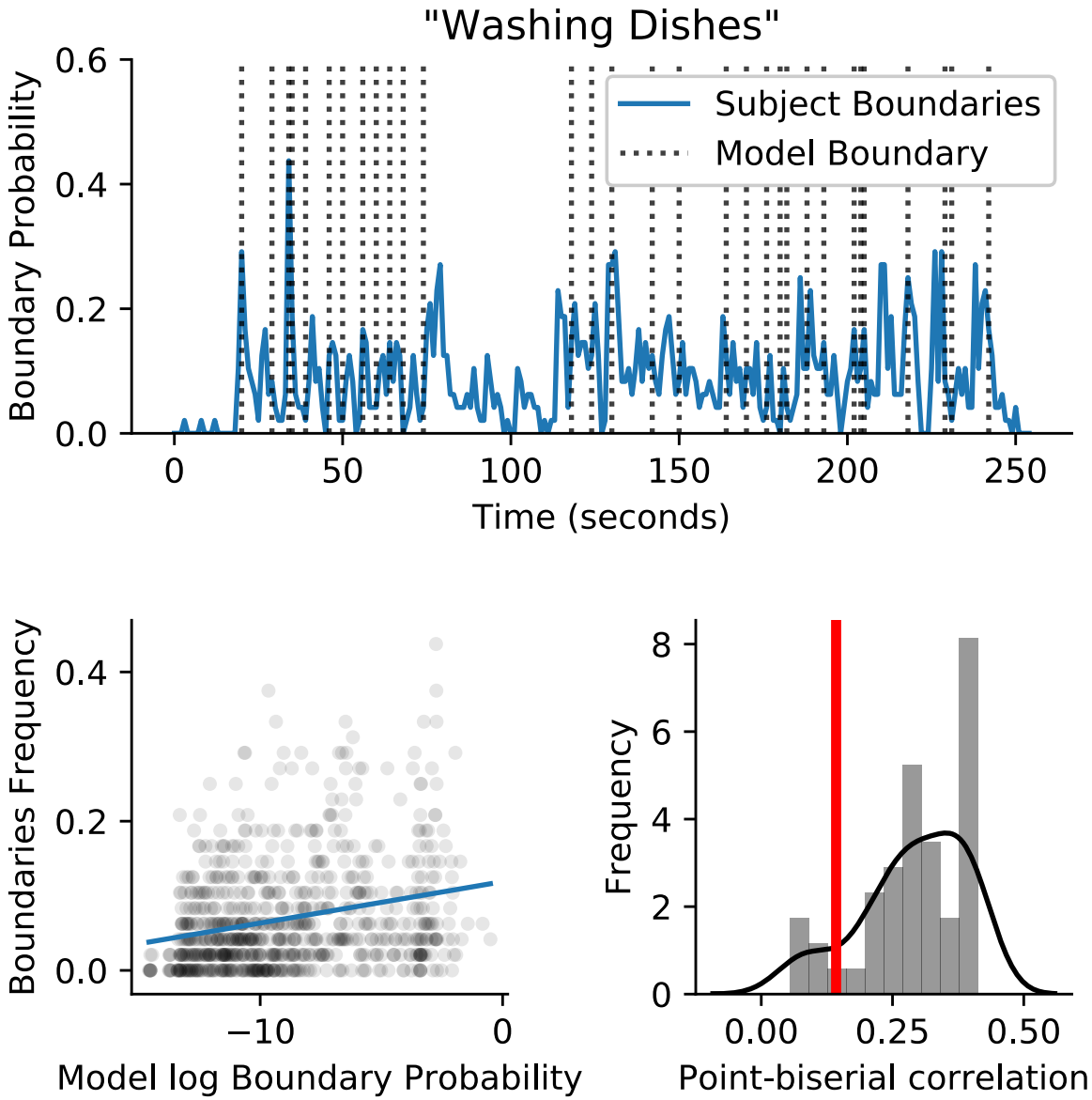

Figure 4. **Video Segmentation**. *Top*: SEM generates human-like boundaries. The model MAP estimate of boundaries is shown for the "washing dishes" video and compared to human segmentation frequency. *Bottom Left*: Model log boundary probability compared to human segmentation frequency at each moment in time. Each grey dot is represents a 1s interval and the regression line is shown in blue. *Bottom Right*: Model point-biserial correlation (red line) compared to individual subjects across all three videos (grey bars).

that surprising or unpredicted stimuli will produce event boundaries is axiomatic. The model is sensitive to both prediction error and the uncertainty of prediction, a property that can be derived analytically (see Eqn 7).

Less apparent, however, is how the model responds to other types of statistical structure. At first glance, the role of predictive inference in event segmentation might suggest that humans rely strictly on surprising outcomes to drive segmentation. However, Schapiro et al. (2013) identified community structure as a feature that drives human event segmentation, even when controlling for predictability. In this context, a community refers to a group of nodes in a graph that are densely interconnected. Schapiro et al. (2013) showed subjects a sequence of stimuli drawn from a graph that had multiple communities, and subjects were asked to note transition points between stimuli. An important feature of the task is that transitions were equated for probability, such that a community transition was no more or less likely than any other individual transition. This was done to rule out the possibility that subjects rely solely on unpredictability in the environment to identify event boundaries.

Nonetheless, subjects preferentially marked event boundaries at community transition points, respecting the graph structure. Schapiro and colleagues further demonstrated that a recurrent neural network trained on the same sequence of stimuli developed internal representations that were similar for community members. Within the network, representational similarity between sequential items decreases at event boundaries, potentially acting as a signal that a boundary has occurred.

Like the Schapiro model, SEM is also sensitive to community structure. To demonstrate this, we simulated the model on 1400 stimuli generated by a random walk on the graph used in Schapiro et al. (2013) in a sequence exposure phase, followed by 600 stimuli from randomly drawn Hamiltonian paths in a parsing phase.[3] Consistent with the previously reported human behavior and the Schapiro model, SEM has higher boundary probability for a community transition than a non-community transition, across both the training trials and in the Hamiltonian paths of the parsing phase

[3] A Hamiltonian path is a path through the graph that visits each node exactly once.

(Figure 5b).

It is important to note that boundary probability and prediction error are related in SEM, such that more surprising scenes are more likely to lead to an event boundary (Figure 5c). Consequently, the average log probability of each successive scene, a Bayesian measure of prediction error,[4] is lower at community transitions than at other transitions, meaning that community transitions are more "surprising" than non-community transitions (Figure 5d). This might be seen as surprising giving the equating of transition probability in the task. However, predictability in the environment is not the same as predictability from the point of view of an agent, and SEM's generative model is not equivalent to the generative process of the task. Furthermore, the recurrent neural network model proposed by Schapiro and colleagues relies on similar computational principles as the recurrent neural network we used to model event dynamics. Thus, the two should be sensitive to similar features of the task. As such, we don't view the prediction error account of event segmentation (Reynolds et al., 2007) as incompatible with that of temporal community structure.

Generalizing structure

Above, we hypothesized that events capture an underlying structure that is to some extent invariant to the surface features of scenes. This is motivated both by the computational goal of generalization, as well as the empirical observation that people tend to remember the surface features of an event less well across time while nonetheless maintaining an accurate memory of the situation (Kintsch, Welsch, Schmalhofer, & Zimny, 1990; Radvansky & Zacks, 2014).

As a preliminary demonstration of the model's ability to generalize structure, we present here a simple example: imagine an event structure that is defined by (1) a person asking a second person a question, followed by (2) the second person responding to the first person. Symbolically, we can represent this event as $Ask(Tom, Charan) \rightarrow Answer(Charan, Tom)$, where we have named the two people

[4] Formally, this is defined with the density function $\Pr(\mathbf{x}_{t+1}|\mathbf{x}_{1:t}, \hat{\theta}) = \sum_{e_t} \Pr(\mathbf{x}_{t+1}|\mathbf{x}_{1:t}, e_t, \hat{\theta})$.

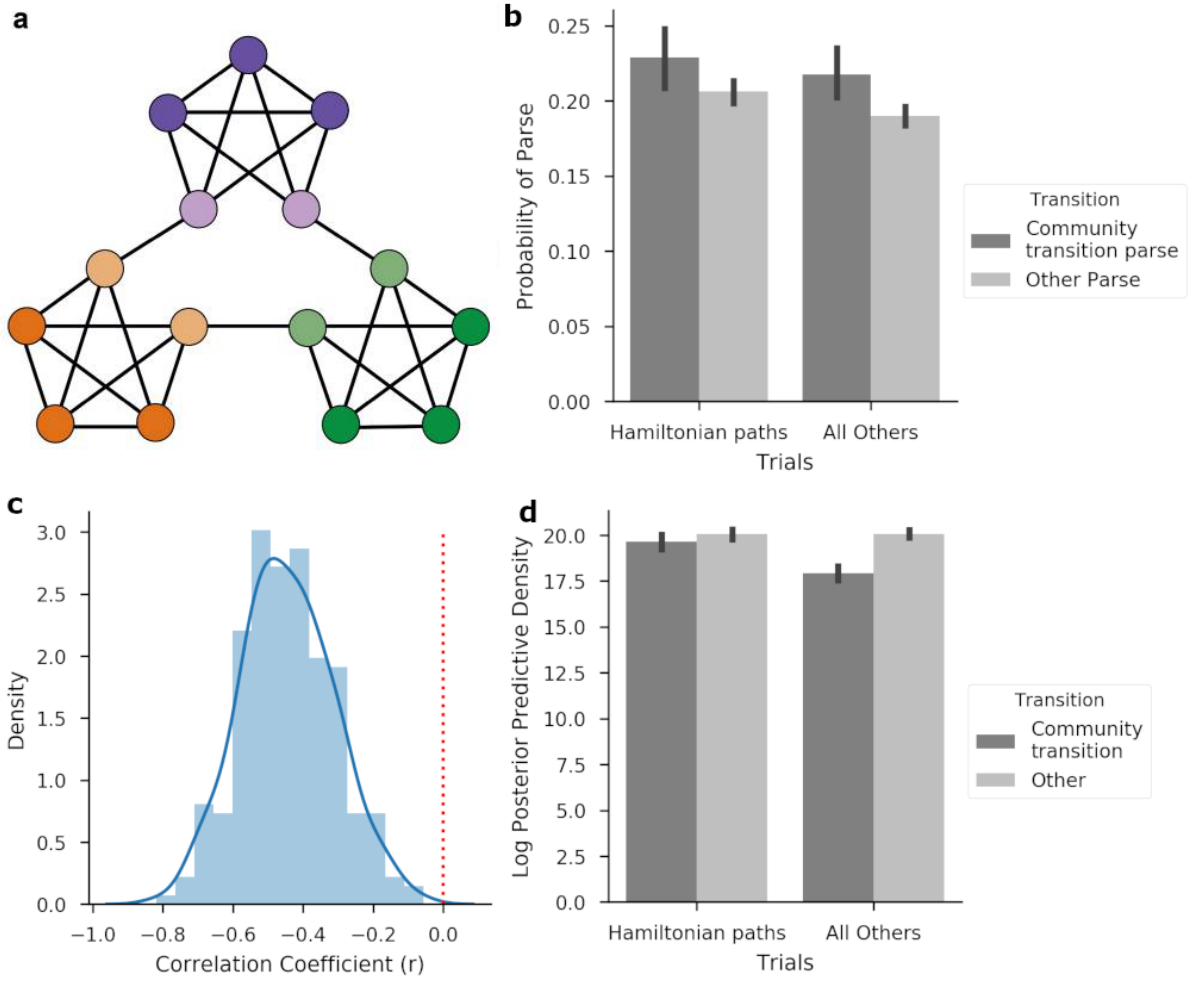

Figure 5. **Event boundaries at graph community boundaries**. (a) The graph structure of the transition matrix, adapted from (Schapiro et al., 2013). Sequences of stimuli were generated by drawing from walks through this graph. (b) The probability of a parse (event boundary) is shown between pairs of items that reflect a community transition (dark grey) or within a community (light grey) in all trials in the simulation (left) and segregated for Hamiltonian paths through the graph (right). (c) Pearson's r coefficient between the log posterior predictive density of each scene and the boundary probability is shown for the sample of simulations. (d) The log posterior predictive density for each scene. Lower values correspond to greater surprise under the model.

"Tom" and "Charan" and the expression means that Tom asks the question and Charan answers (Figure 6).[5] One question we can ask is whether SEM can learn this structure and generalize it to new fillers (i.e., other people) that it has not encountered. Specifically, we can asses whether the model is sensitive to the structural form of the event, without regard to role/filler assignments.

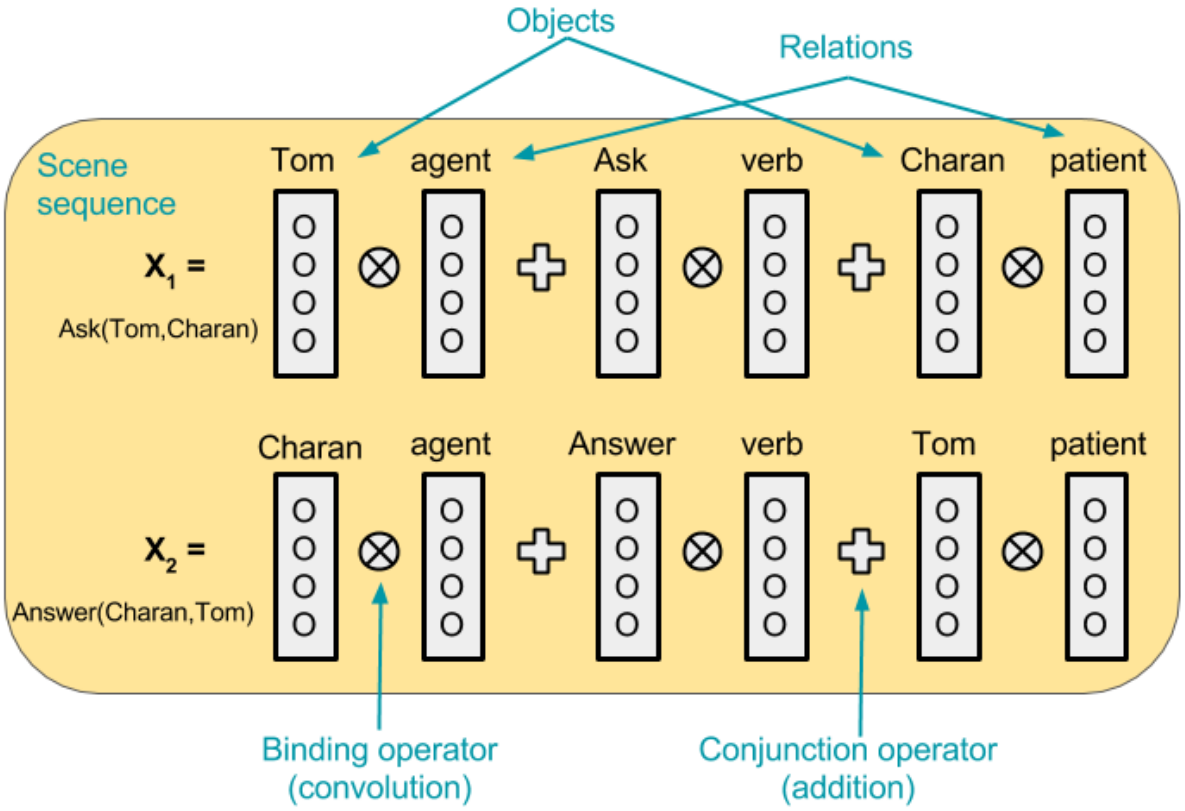

Figure 6. **HRR of the question/answer task**. Each filler is bound to its role by circular convolution. Scenes are composed of role/filler bindings with vector addition.

To probe this, we examined the probability of an event boundary between $Ask(A, B)$ and $Answer(B, A)$ for arbitrary fillers of A and B. As we are concerned here with the use of relational structure to delineate event boundaries, we pre-trained SEM with examples of the sequence $Ask(A, B)$, $Answer(B, A)$. This simulates event structures that are already known. SEM was pre-trained on 3 unique sequences of $Ask(A, B)$, $Answer(B, A)$ with 6 unique fillers. Each scene vector was composed using

[5] We can interpret this structure linguistically but we do not have to – this symbolic representation is valid whether we are watching the event occur or reading about it in a text.

the HRR as previously described. Each independent feature was represented with a spherical, zero-mean, Gaussian random vector ($\mathbf{x} \sim \mathcal{N}(0, d^{-2}\mathrm{I})$) and similarity between fillers was encoded with a shared component. Due to the simplicity of the event dynamics, we estimated the event dynamics without recursion and replaced the GRU layer in our function approximator with a non-recurrent, but otherwise equivalent, layer.

We simulated SEM on four test sequences, probing the event boundary probability between the first and second scenes. We also probe the probability that the second scene belongs to a new event (as opposed to a new instance of the previously experienced event). The first sequence, $Ask(Tom, Charan) \rightarrow Answer(Charan, Tom)$, was included in the pre-training sequences and acts as a negative control, as SEM should assign a low event boundary probability. A second sequence $Ask(Tom, Charan) \rightarrow Chase(Dog, Cat)$ deviates from the event structure and acts as a positive control. As expected, SEM assigns a low boundary probability to the probe $Answer(Charan, Tom)$ (Fig 7) and a high probability of both an event boundary and a new event for the probe $Chase(Dog, Cat)$.

Of particular interest is whether SEM generalizes the structure of the event to arbitrary fillers. To test this, we provided SEM with the sequence $Ask(Dick, Bruce) \rightarrow Answer(Bruce, Dick)$, which used fillers ("Dick" and "Bruce") that were held out of the pre-training set for this purpose. Here, SEM assigns a low, but non-zero, probability both that an event boundary has occurred (18.0%) and that the probe belongs to a new event (12.2%). Equivalently, this corresponds to a high probability that an event boundary has not occurred (82%). This is because, although SEM does generalize the structure of the event, it is sensitive to the non-structural features as well. Non-structural features are novel, and the probability of the previously learned event under the model is relatively lower. SEM makes the empirical prediction that people rely on both structural and non-structural features for segmentation, but as the structure of the event is thought to be more predictable across time (Richmond & Zacks, 2017), we would expect the event structure to dominate for familiar event types.

Finally, we look at how SEM reuses events multiple times. SEM uses Bayesian

inference to identify events and supports re-using an event model following an event boundary. To illustrate this, we provide SEM with the sequence $Ask(Tom, Charan) \rightarrow Ask(Tom, Charan)$, in which the first item is repeated. Here, SEM correctly infers an event boundary but does not assign the second item to a new event. Instead SEM infers that the original event has been restarted. Interestingly, this is a property of the structure of the event, and SEM will reuse an event following a boundary based on this structure. This can be shown by providing SEM with the sequence $Ask(Tom, Charan) \rightarrow Ask(Bruce, Dick)$, which reuses the event with a pair of novel fillers. SEM infers an event boundary between the two items and re-uses the previous event (i.e., infers a low probability of a new event), again showing structure sensitivity. Taken together, these simulations predict that people group events with the same relational structure. If subjects were trained on an event with relational structure, we would expect subjects to delineate the same boundaries for a sequence of events with the same relational structure but novel fillers.

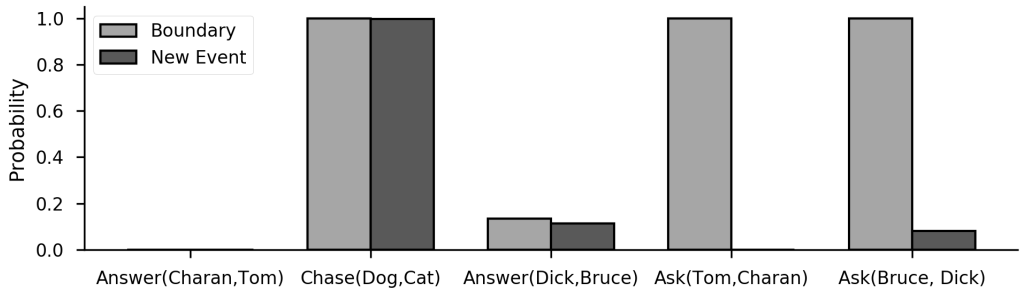

Figure 7. **Generalization task**. The probability assigned by the model for an event boundary (light grey) and a new event (dark grey) are shown for four test events. Each event begins with the scene $Ask(Tom, Charan)$ and ends with the denoted scene, except for the event $Ask(Bruce, Dick) \rightarrow Answer(Dick, Bruce)$, From left to right, the probabilities are shown for an event in the training set, a structurally dissimilar event, a structurally similar event with novel role/filler bindings, a control where the original event was restarted, and the beginning of an structurally similar event with novel role/filler bindings.

Test Case	1st Scene	2nd Scene
Training example	Ask(Tom, Charan)	Answer(Charan, Tom)
Structure violation	Ask(Tom, Charan)	Chase(Dog, Cat)
Novel role/filler binding	Ask(Bruce, Dick)	Answer(Dick, Bruce)
Structure Re-use	Ask(Tom, Charan)	Ask(Tom, Charan)
Structure Re-use with novel fillers	Ask(Tom, Charan)	Ask(Bruce, Dick)

Table 1

Stimuli for generalization task. *For each of the five test cases, the probability of an event boundary was measured between the 1st and 2nd scenes.*

Structured memory inference

We now turn to the memory predictions of the model. We first focus on false memories, which have long been thought to be a consequence of reconstructive memory (Bartlett, 1932; Roediger & McDermott, 1995), and are a natural prediction of the reconstructive memory model.

In a classic finding, Bower et al. (1979) found that subjects who read multiple similar stories drawn from the same "script" would falsely recall portions of the story that weren't present in the original story. For example, subjects might read a story in which a character "John" goes to the doctor and reads a magazine before seeing the doctor, and a second, similar story in which a character "Bill" goes to the dentist and has to wait. Given these two stories, which shared a common event structure and were drawn from the same script, a subject might falsely recall the detail that "John" has to wait to see the doctor, a likely inference that was nonetheless not stated in the original story. This tendency to recall unstated memory items increased with the number of stories subjects read from the same script, suggesting that subjects were recombining elements of stories with a shared event representation in the recall process. This process may be adaptive to the degree that it reflects inferences about unexperienced scenes that nonetheless occurred.

We use the reconstructive memory model to generate structured false memories

Script	1st Scene	2nd Scene	3rd Scene	4th Scene
1	Goto(John, Doctor)	Checkin(John)	Read(John, Magazine)	Treat(Doctor, John)
	Goto(Bill, Dentist)	Read(Bill, Magazine)	Wait(Bill)	Treat(Dentist, Bill)
	Goto(Natasha, Chiropractor)	Checkin(Natasha)	Wait(Natasha)	Treat(Chiropractor, Natasha)
2	BuyTicket(Jill, Movie)	FindSeat(Jill)	Buy(Jill, Popcorn)	Watch(Jill, Movie)
3	Wakeup(Sarah)	GetDressed(Sarah)	Eat(Sarah, Toast)	Leave(Sarah)

Table 2

Stimuli for Bower task. *Each row denotes a story with comprised of four scenes, representing a simplified narrative containing relational structure. Each of the stories in script 1 contain common structural elements, with the same beginning and endings and a shared 2nd or 3rd scene, but with different fillers in the roles.*

with a paradigm similar to the one presented in Bower et al. (1979). In the original stimulus set, stories with the same script had highly similar beginnings and endings, and had a combination of structurally similar sentences and distinct, story specific sentences in the middle that suggested a similar event trajectory. To model this, we defined a simplified set of five stories from three different 'scripts', each of which was comprised of four scenes (Table 2). Three stories belonged to the same script and shared a common structure but used different fillers within the structure. These sentences all shared the first and last scene, and any two of the three shared a third scene.

The manipulation of interest is whether providing SEM with multiple similar stories will increase the probability of a script producing semantically valid false memories under the reconstruction model. Concretely, we test for the reconstructive memory probability of the probe cue *Wait(John)*, a scene that does not incur in any story, but is syntactically correct and consistent with the *Goto(John, Doctor)* story in script 1 (Table 2).

We provided SEM with three stories, including either one, two or three stories from script one. In our simulations, we used the HRR as previously described to encode each scene but provided no pretraining or other semantic knowledge to the model. Verbs (e.g., *Goto*) are encoded with a common feature as are agents (e.g., *Natasha*) and objects (e.g., *Popcorn*). SEM was first provided the scenes for each story in order to

learn the event dynamics and infer the event labels. Because we are not examining segmentation with these simulations, all of the scenes from a single story were assumed to belong to the event and SEM was tasked with inferring a single label for them. It is worth noting the segmentation problem present in Bower et al. (1979) is trivial: stories were presented with clear external cues for the beginning and ends with intervening time between each story.

We then used the Gibbs sampling algorithm as previously described (see Table D1 for parameter values) to simulate reconstructive memory. We modeled a two alternative forced choice recognition memory test, comparing the script-consistent false memory probe *Wait(John)* with a syntactically valid but script-inconsistent memory probe *GetDressed(John)*. This memory score can be described as the expectation of the comparison

$$\mathbb{F}[\Pr(A|\tilde{y}) > \Pr(B|\tilde{y})] \tag{17}$$

under the memory model, where A and B are memory probes. Let $f(\mathbf{x}) = \Pr(\mathbf{x}|\tilde{y})$ be the recognition memory probability under the model, and the expectation above defined as an expectation over f. Our reconstruction samples $\hat{\mathbf{x}}$ (equation 19) are drawn from samples of f. Thus, if we assume a function $g(\mathbf{x}, \hat{\mathbf{x}})$ to be monotonically related to a sample of f, we can evaluate the ordinal comparison of the two memory probes on g and approximate the memory score with the average of N samples:

$$\mathbb{F}[\Pr(A|\tilde{y}) > \Pr(B|\tilde{y})] \approx \frac{1}{N} \sum_{\hat{\mathbf{x}} \in \{\hat{\mathbf{x}}\}_{1:n}} \mathbb{I}[g(\mathbf{x}_A, \hat{\mathbf{x}}) > g(\mathbf{x}_B], \hat{\mathbf{x}}) \tag{18}$$

where $\mathbb{I}[\cdot] = 1$ when its argument is true, and 0 otherwise. Here, we choose $g(\mathbf{x}, \hat{\mathbf{x}}) = \sum \exp\{-\gamma||\mathbf{x} - \hat{\mathbf{x}}||^2\}$.

Figure 8 shows the results of our simulations. The script-consistent false memory probe *Wait(John)* was compared to the script-inconsistent probe *GetDressed(John)*. The recognition memory probability for the script-consistent probe increased monotonically with the number of script instances in the stimuli, similar to the behavioral finding of Bower et al. (1979). For a single script instance, this was below chance, as the comparison probe *GetDressed(John)* was more reflective of the training

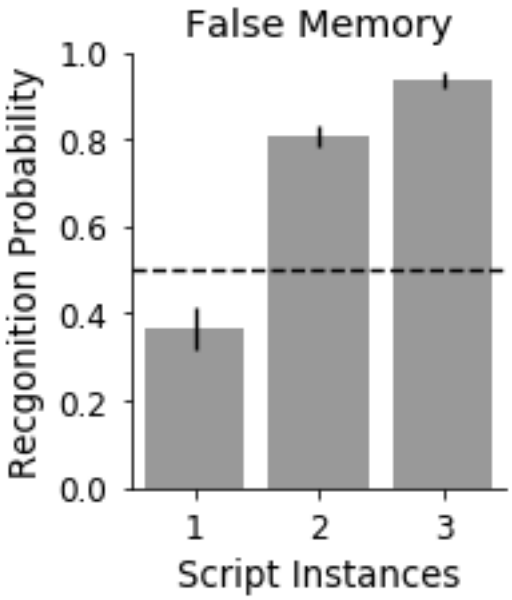

Figure 8. **False memory simulations**. The recognition memory probability for a script-consistent false memory probe is shown as a function of the number of stories of the same "script" during learning. Chance (50%) is denoted by a dotted black line.

set than the script consistent probe. Additionally, the reconstruction process improved as a function of the number of script instances. This can be seen in the frequency with which the original memory traces were included in the reconstruction sample, which increased monotonically with the number of script instances (1 script instance: 0.82, 2 script instances: 0.88, 3 script instances: 0.92). Under SEM, these two effects are related due to the regularizing effects of the reconstructive process.

Event boundaries and working memory

An empirical consequence of event boundaries is that items that occur in an ongoing event are better remembered than items that occur immediately prior to an event boundary (Pettijohn & Radvansky, 2016; Radvansky & Copeland, 2006; Radvansky et al., 2011; Radvansky, Tamplin, & Krawietz, 2010). A study by Radvansky and Copeland (2006) had subjects remember items in a virtual environment as they moved from room to room. In each room, subjects put an item they were carrying (but was not visible to them) on a table and picked up a second object, before

carrying it to another room. Subjects were given a memory probe either immediately after walking through a door (*shift condition*) or at an equidistant point in a larger room (*no-shift condition*). Overall, subjects remembered items better in the no-shift condition than in the shift condition, suggesting that the act of walking through a door interfered with the item memory. This appears to be distinct from context-change effects that have been observed in long-term memory (e.g. Godden & Baddeley, 1975), because returning to the original room in the shift condition did not eliminate the memory decrement (Radvansky et al., 2011).

One potential explanation of these effects put forth by Radvansky (2012) is that an ongoing event is preferentially stored in working memory. During a memory probe after an event boundary, two event models have to be represented at the same time to complete the probe, causing memory interference previously described as a fan effect (Anderson, 1974; Radvansky & Zacks, 2017). In our model, memory reconstruction acts on a degraded set of features and a degraded event token simultaneously. As such, noise from one event memory will affect the reconstruction of the other—i.e., memory interference similar to the fan effect. If there is preferential access to the ongoing event, then this reduces a source of noise in the reconstruction process by obviating the need to infer the correct event.

We simulated the model on a simplified variant of the task in Radvansky and Copeland (2006). In the original task, subjects navigated from room to room in a virtual environment. We simplified this to a stereotyped set of scenes, using a similar sequence of structured scenes as in the previous memory experiment and in the generalization simulations. To model the interactions in each room, we provided SEM with observations of (1) entering the room, (2) putting an object down (3) picking up the next object and (4) leaving/crossing the room. Each of these scenes is composed of a verb (e.g., "Enter") and context that corresponds to an individual room (Table 3). We assumed that the object to be picked up was observed in the first two scenes. For balance, we assumed that the object that was put down was observed in the second two scenes. We further assumed that the pickup object was bound to the verb "Pickup" in

the third scene. Each of these scenes was encoded using a HRR as outlined in table 3 with Gaussian random vectors for each of the features.

Scene	Features
1	Enter + Room + Object A
2	PutDown + Room + Object A
3	PickUp(Object A) + Room + Object B
4	Leave + Room + Object B

Table 3

Scene representation for Simulations of Radvansky and Copeland (2006).

Each scene was composed of vectors corresponding to the features in the scene.

The model was trained on a sequence of 15 rooms and given a memory test after each room. We assumed that event boundaries corresponded to entering and exiting a room, and constrained the model to infer an event boundary only at this time. The model was free to choose to reuse an event model or infer a new one for each room. Access to the current model was simulated by modulating the corruption noise, ϵ_e, associated with the event label. In the switch condition, ϵ_e was equivalent for all scenes in memory and set to a value of 0.25. In the no-switch condition, this corruption noise was lowered by increasing the value of $\epsilon_e = 0.75$ for the scenes associated with the most recent event, simulating privileged access to the current model. The reconstruction process was otherwise equivalent. Error was assessed by the probability that a corrupted memory trace was included in the reconstruction sample. Mirroring the results in the human studies, the model had lower error in the no-shift condition than in the shift condition (Figure 9).

Event boundaries improve overall recall

While these short term memory effects suggest that event boundaries interfere with memory, the relationship between event structure and subsequent memory is not always intuitive. Overall, extracting relevant event structure tends to improve memory,

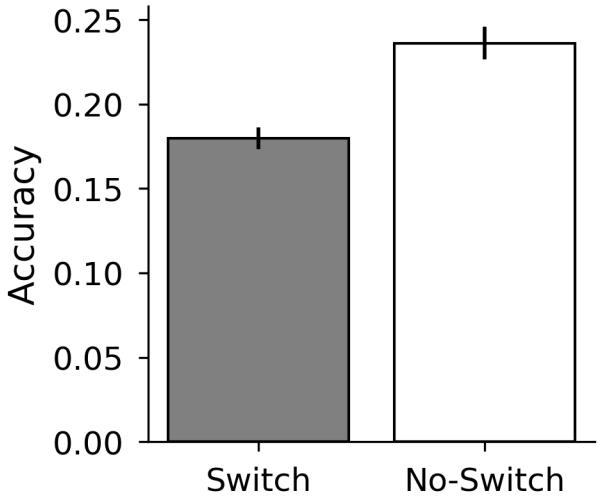

Figure 9. **Simulations of the task in Radvansky and Copeland (2006)**. Reconstruction memory accuracy is shown for items both after an event boundary (*Switch*) and before an event boundary (*No-Switch*).

as subjects with better segmentation judgements tend to have better subsequent recall (Sargent et al., 2013; Zacks et al., 2006). Similarly, studying list of items or video-taped lectures in multiple contexts leads to better overall memory (S. M. Smith, 1982, 1984; S. M. Smith & Rothkopf, 1984). Within the context of SEM, event structure plays an important role in the reconstruction memory process. Poor segmentation leads to a more noisy reconstruction process and thus worse overall memory.

Interestingly, the benefits of event boundaries within the reconstructive memory process extend to cases where the sequence of studied memory items are random and where there is no clear relationship between events and studied items. In a series of studies, Pettijohn and Radvansky (2016) demonstrated that when subjects were give a list of items separated by a physical or virtual event boundary, subsequent recall was higher overall. This suggests that segmentation itself influences memory is important for overall memory irrespective of environmental statistics.

Here, we probe the model for these effects by simulating experiment 1 in Pettijohn et al. (2016). In this experiment, subjects were given a list of 40 words to remember while moving between four locations in physical space, divided into 4 ordered sub-lists of 10 words each. Subjects read one sub-list (10 words), then moved to a new location

in space, either in a new room (*shift condition*) or a new space in the same room (*no-shift condition*) that was equated for physical distance, and read a second sub-list (10 words). Subjects were then given a distractor task, and finally asked to recall as many of the words as possible from both sub-lists. Subjects had higher recall accuracy in the shift condition than in the no-shift condition.

We simulated these effects by generating a list of 20 items, each as a Gaussian random vector. We trained the model on the list, either constraining the model to learn all items within a single event (no-shift) or assuming a single event boundary halfway though the list (shift). We then used the reconstruction procedure to create a reconstructed memory trace and probed memory recall as the probability that each corrupted memory item $\tilde{y}_i$ is included in the reconstructed trace, defined by equation 15. Overall, the accuracy is higher in the switch than in the no-switch condition (Fig 10), replicating the findings of Pettijohn et al. (2016). During the reconstruction process, the uncertainty about a single item propagates to its neighbors as a consequence of the dynamics of the event schema. In general, the event schemata reduce the overall reconstruction uncertainty by regularizing the process, but they nonetheless propagate uncertainty between noisy scene memories. Because the dynamics end at boundaries, boundaries prevent the uncertainty from spreading.

Sequential recall

Even as event boundaries improve memory performance overall, they introduce specific deficits to sequential recall. Given narrative texts that include a temporal shift, subjects are worse when remembering the next sentence immediately after a temporal shift than immediately before (Ezzyat & Davachi, 2011). This impairment of sequential order memory by boundaries occurs even in the absence of a naturalistic event structure (DuBrow & Davachi, 2013, 2016) and is not associated with an impairment of associative memory (Heusser et al., 2018). This suggests that the event structure that subjects learn (as opposed to the structure of the task) is responsible for this memory effect. In the context of the model, SEM continuously estimates the transition structure

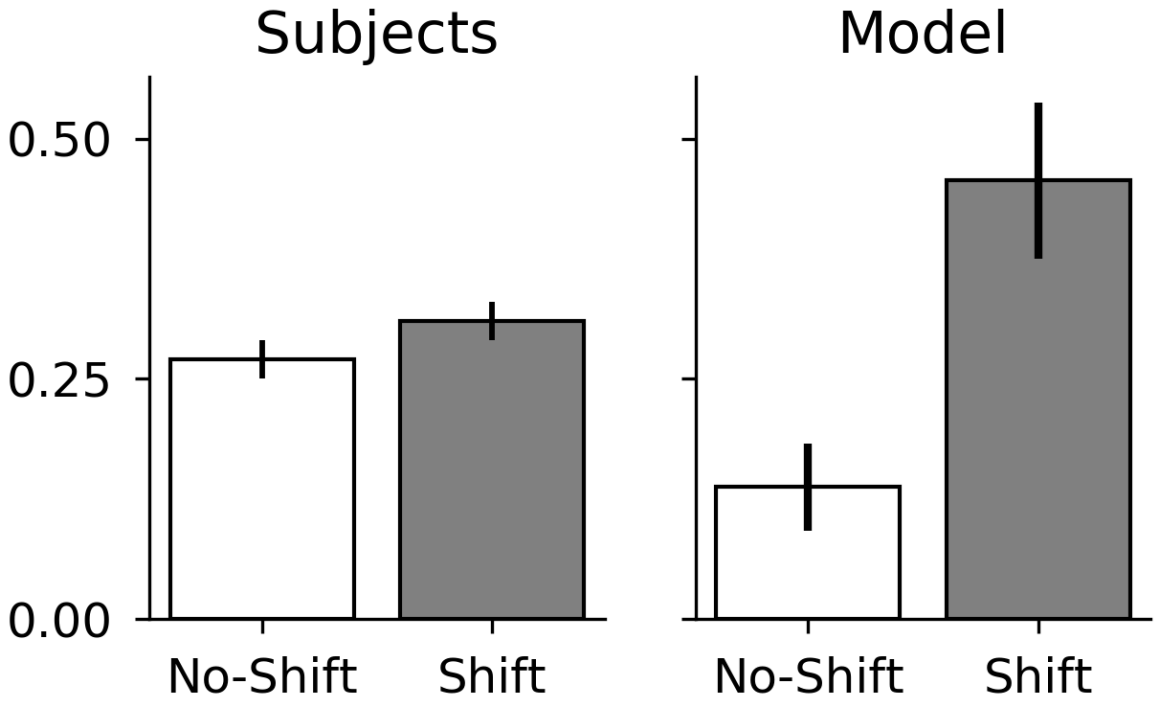

Figure 10. **Simulations from Pettijohn et al. (2016)**. *Left*: Human subjects have been recall in the shift vs no-shift conditions (data reproduced from Pettijohn et al., 2016). *Right*: Model has higher recall in shift vs no-shift condition.

between scenes as the event dynamics, irrespective of the regularity or unreliability of that structure. Introducing a boundary disrupts learning an association between successive scenes. This becomes apparent in the reconstruction process, where the presence of boundaries disrupt sequential memory, even as they aid reconstruction overall.

In order to demonstrate this, we simulated experiment 1 of DuBrow and Davachi (2013). In the original experiment, subjects were presented with 400 items sequentially (200 celebrity faces and 200 nameable objects) across 16 study-test rounds of 25 images each, while performing a task in which subjects either made a male/female judgement (faces) or a bigger/smaller judgment (nameable objects). Following each round, subjects were asked to recall each item they saw in order. Sequential recall accuracy, as measured by direct transitions between consecutive items, was higher for items immediately before a task/category switch than immediately after.

We simulated 5 events, alternating between two categories after the presentation of 5 items each. Each item was embedded by combining an item-specific factor (i.e., a random Gaussian vector) and a shared category feature (itself a random Gaussian vector) with addition in order to encode the similarity relationship between items of the same category. As in the previous memory simulations, the model inferred a single event label the scenes within a single events and event boundary locations were provided to the model as a simplifying assumption of the task. The reconstruction was performed as previously described, and SEM inferred a reconstructed memory trace and event label for each moment in time. Order memory was assessed by measuring the probability that two sequential probe items were reconstructed in the correct sequential order. Thus, if item "B" followed item "A" in the stimuli, it was scored as correct if it appeared in this order in the reconstruction, regardless of the position of these items in the reconstructed sequence.

Figure 11 summarizes the results of the model simulated on this task. As expected, the model has higher serial recall accuracy for items studied immediately prior to the category/task switch than after, mirroring previous empirical results. Mechanistically, this occurs because the model learns the dynamics over the scenes within an event, whereby each item is learned as a function of the previous item within the event, but there is no direct association between sequential items in separate events. It is worth noting that this occurs regardless of the structure of the task and reflects the hypothesis that people are continually and automatically learning about temporal structure. The reconstruction process, which regularizes the corrupted memory trace with the learned dynamics, thus aids the formation of sequences within events. Overall temporal information is not otherwise lost, as the identity of each event and the within-event dynamics help order the entire sequence, but there is less of an association between two sequential items across an event boundary in the reconstruction.

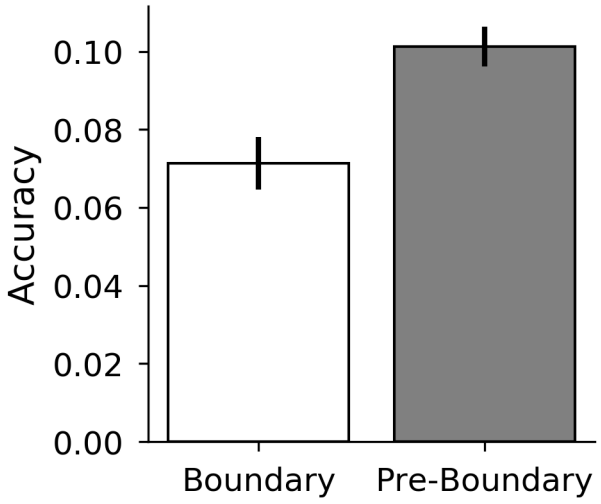

Figure 11. **Serial recall across an event boundary.** The proportion of correct transitions between an item and the following item was worse across an event boundary than immediately prior to the boundary.

Discussion

In this article, we have tackled the problem of learning and understanding events through the lens of probabilistic reasoning. We have made 3 main contributions with our model: first, we demonstrated that SEM can produce human-like segmentation of naturalistic data, suggesting that the computational principles outlined here are scalable to real-world problems; second, we have shown how events can capture generalizable structure that can be used for multiple cognitive functions; and third, we have shown that these principles are sufficient to explain a wide range of empirical phenomena.

Segmentation

In the current work, we have built on previous models of event segmentation and shown that our model is able to produce human-like segmentation of naturalistic datasets. While the low-dimensional and often non-ecological stimuli used to evaluate previous computational models are diagnostic of computational principles, it is important to show these principles are sufficient to account for naturalistic environments. Without this demonstration, it is unclear whether the behavior produced by the model is a consequence of the artificiality of the stimuli. Furthermore, as we are

able to show that human event boundaries mirror the boundaries predicted by the model, we can argue that human event segmentation is sensitive to the same underlying regularities in the stimuli that the model is, further validating the model.

Previous computational models of events have, like SEM, used recurrent neural networks to learn the event dynamics. Even as the models differ in the specifics of the networks used, their similarity leads to qualitatively similar predictions. This is most evident in our simulations of community structure (Fig. 5). SEM recreates the same qualitative pattern of behavior as both the Schapiro et al. (2013) model and human subjects. All three show increase tendency to delineate an event boundary at community transition points. It is not surprising that SEM and the Schapiro model learn similar event boundary points given the similar method of learning event dynamics.

Interestingly, these event boundaries in SEM correspond with increased prediction error, even as the Schapiro task was designed to *not* elicit prediction errors at community transition points. We note that this is a consequence of the distinction between the generative process of the task (in which community transitions are equally probable as other transitions) and the generative process assumed by the agent (i.e., SEM's generative model). SEM embodies a set of assumptions and inductive biases that differ from a direct inversion of the generative process of the Schapiro task. We would expect this to hold true with people as well, as people generally do not have access to the true generative process of an unknown task. As a consequence of these difference in assumptions, SEM predicts an increase in prediction error at community transitions points, even though this isn't a feature of the task. Thus, we don't view the prediction error account of event segmentation (Reynolds et al., 2007; Zacks et al., 2007) as incompatible with the community structure account proposed by Schapiro et al. (2013), since we can demonstrate the both sensitivties within SEM.

Algorithmically, SEM is closely related to the model proposed by Reynolds et al. (2007). Both models employ a hierarchical process where event dynamics are learned with a recurrent neural network at the lower lever, and where a higher level "event" constrains the lower-level model. These models are also similar in that prediction errors

play a key role in determining the identity of the higher-level "event" (Equation 7). Computationally, the type of gating mechanism used in the Reynolds model is similar to non-parametric clustering with a Chinese Restaurant Process prior (Collins & Frank, 2013), and while we make different commitments and assumptions, similar computational principles drive computation in both models. More broadly, using a recurrent neural network to learn temporal dependencies is a powerful computational tool, and has become more common recently in models of cognition. In a notable piece of recent theoretical work, Wang et al. (2018, 2016) argue that an architecture of stacked recurrent neural networks is sufficient to explain a host of empirical findings in human reinforcement learning and serves as a good model for the prefrontal cortex. Unlike SEM, their model requires extensive pre-training but nonetheless generalizes to novel task-variants efficiently. A similar architecture was also employed by Butz, Bilkey, Humaidan, Knott, and Otte (2018) to show that events are useful in goal-directed planning. They trained an agent to learn a forward model of states using recurrent neural networks and show that allowing these networks to be contextualized by events leads to more efficient learning in an artificial domain.

SEM is also related to computational techniques used to model sequential processes by inferring latent states. The hidden Markov model (HMM) is an instructive comparison because, like SEM, it assumes each observation is generated by a latent process associated with a discrete latent variable (Bishop, 2006; Rabiner, 1989). The key difference is that the HMM does not model the internal dynamics of the latent processes across multiple time points. While this limits the HMM as a model of events, it is nonetheless an empirically useful model. Recent work by Baldassano et al. (2017) used HMMs to model event boundaries in human fMRI data, providing strong evidence that people spontaneously generate hierarchical event boundaries with realistic experience. SEM is also similar to the switching linear dynamical system (SLDS; Fox, Sudderth, Jordan, & Willsky, 2010; Ghahramani & Hinton, 1996), a generalization of the HMM that assumes the process associated with each discrete latent state is a linear dynamical system. This assumption facilitates learning the types of dynamical processes that

constitute events, albeit at the cost of strong constraints on the form of these dynamical systems. SEM can be viewed as a form of switching nonlinear dynamical system.

Generalization

A key difference between SEM and other computational models of event cognition is how SEM generalizes previously learned events to novel experiences. SEM accomplishes this via non-parametric Bayesian inference, which allows SEM to learn and reuse events as appropriate. Furthermore, because SEM assumes that events exist in a structured and distributed representational space, this generalization applies not only to the surface features of the event but also its underlying relational structure. As such, SEM can generalize an event dynamic even when many of the features are different, including role/filler bindings (Figure 7). The generalization of the relational structure is advantageous because its dynamics are thought to be smoother and thus easier to generalize than the surface features of a task (Radvansky & Zacks, 2011; Richmond & Zacks, 2017).

This places the burden of encoding structure on the representational space (in our case, an HRR), and allows the dynamics to be learned with a parametrized function. While we do not make a strong commitment to how this representational system is learned by neural systems, we note that the convolutions required to compute an HRR are thought to be plausible in biological networks (Yamins & DiCarlo, 2016). Regardless of how this representational space is learned, smooth functions are sufficient to generalize because similar structures are represented with similar vectors. This allows SEM to leverage relational structure when identifying the event boundaries and consequently, learn more general event dynamics that abstract away some surface features. Hence, the event dynamics SEM learns are consistent with previous theoretical accounts in which events are encoded in terms of abstract, high-level features (Radvansky & Zacks, 2011).

Neural and behavioral measures tend to support the claim that events represent high-level information. A recent study by Baldassano, Hasson, and Norman (2018)

found a neural representation for events that shared a high-level schematic structure but otherwise had very different features. In this passive task, subjects either watched a movie or listened to an audio narrative that followed a shared script, such as ordering food in a restaurant, but varied the actors, genre and timing. Brain activity in the medial prefrontal cortex was highly predictive of the script and could be used to align the timing of events within the story, regardless of modality and other features. This suggests that subjects maintained a form of structural information about the films independent of their low-level features. More broadly, relational information is critical to memory (Cohen, Poldrack, & Eichenbaum, 1997), and thus we would expect to see it in event representations. This neural evidence is consistent with earlier memory studies in which subjects tended to lose lower-level descriptive details of narrative texts (e.g., the exact words used in order) while nonetheless maintaining an accurate high-level description of events (Kintsch et al., 1990; Zwaan, 1994).

A limitation of our model is that the use of structure is limited to within an event model, and elements of structure cannot be combined across events. This contrasts with human cognition, which is believed to be *systematic*, meaning that component pieces of knowledge can be combined in a rule-like manner to give rise to novel thoughts (Fodor & Pylyshyn, 1988; James, 1890). For example, an event representing a lunch meeting could in theory be composed of events separately representing lunches and meetings. This requires a different form of structure than the form outlined here. Specifically, this form of productive inference requires compositional event representations. Computationally, compositionality can introduce an adaptive statistical bias (Franklin & Frank, 2018) and simplify representational demands by a combinatorial factor, dramatically accelerating learning. As such, compositionality is likely to play an important role in learning event dynamics.

How event dynamics can be composed and how their component knowledge may be learned is an open question for future research. Notwithstanding, we can outline a few possible mechanisms for a compositional system. Perhaps the simplest form of compositionality would be to learn the dynamics of subsets of features independently

and combine the independent predictions. As an intuitive example, the trajectory of a bird flying is generally independent of the trajectory of cars stopping at an intersection. If we observe both in the same scene, we might combine their previously learned patterns independently to generate a single prediction for the next scene. In the embedding space we've described, this is equivalent to linearly combining the predictions of two different systems via vector addition. A second intriguing possibility coming from machine learning and function learning is that of composable kernel functions. Complex functions can often be decomposed into a combination of multiple, simpler functions (e.g. linear or periodic functions) with kernel methods (e.g. Gaussian processes or support vector machines; Duvenaud, Lloyd, Grosse, Tenenbaum, & Ghahramani, 2013; Schulz, Tenenbaum, Duvenaud, Speekenbrink, & Gershman, 2017). Conveniently, these compositions rely solely on matrix algebra. To the degree that we expect event dynamics to resemble combinations of multiple simpler dynamics (e.g., the movement of a ship tacking in the wind resembled a combination of a linear and periodic function) then these compositions may provide a good model of how these components are learned. It is interesting to note that the false memory paradigm (Figure 8) does show aspects of a productive (or constructive) system, creating a novel trace composed from distinct pieces. While there is no explicit mechanism for generating these false memory traces, their probability under the model nonetheless increases as an emergent property of reconstruction.

A second form of compositionality important for generalization is temporal hierarchy, in which longer events are formed from a sequence of shorter events (Radvansky & Zacks, 2011). When asked to judge coarse or fine events, subjects' event boundaries tend to line up such that the boundaries of longer (coarse) events closely align in time with that of shorter (fine) events (Kurby & Zacks, 2011). Evidence for event segmentation at multiple timescales is also present in neuroimaging data (Baldassano et al., 2017). To provide an intuitive example, if washing dishes reliably follows dinner, a model of event cognition should learn this relationship. We have not directly addressed hierarchical events in the current work, but SEM can potentially

solve this problem simply by learning multiple event representations at different time scales (for example, learning a dinner event, a dish-washing event and a dinner+dish-washing event). This is somewhat unsatisfying because a primary computational motivation for learning events is to be able to compose multiple, sequential events together as a forward prediction. This form of compositionality is important for generalization but falls outside the capabilities of SEM as currently instantiated. One potential solution is to learn the relationship between events over time, but how this can be instantiated is a question for future research.

Memory

The memory component of our model builds on several previous probabilistic models of memory. Most directly, it is an extension of the *Dirichlet process-Kalman filter* model (DP-KF) proposed by Gershman et al. (2014), a model of memory using a non-parametric form of the switching Kalman filter. In the DP-KF, observations were assumed to be generated by a switching linear-Gaussian diffusion process, and memory was modeled in terms of inference over this process. Our model is similar to the DP-KF in some ways, and can be seen as an extension with learnable (vs random) dynamics defining each event. This is critical for generalization (e.g., figure 7) and the reconstruction of order memory (e.g., figure 11).

The role that dynamics play in event memory is critical for the simulations presented here. This link has been well established empirically. Humans subjects learn the sequential dependencies between stimuli when grouped together in a coherent event (Avrahami & Kareev, 1994; DuBrow & Davachi, 2013, 2016; Heusser et al., 2018). Furthermore, when asked to recall lists of items without constraints, recall can be temporally organized even when participants are not constrained to recall items in order (Kahana, 1996).

In our model, this sequential information corresponds to schema knowledge. People tend to remember schema-typical information better than atypical information (D. A. Smith & Graesser, 1981) and grammatical sequences better than agrammatical

sequences (Botvinick & Bylsma, 2005), suggesting that sequential dependencies and overall event knowledge play an important role in memory processing.

Event boundaries play an important role in our model by preventing information from spreading between multiple events, which has previously been argued to be a type of fan effect (Radvansky & Zacks, 2017). Because smoothing of event dynamics is broken up by the event boundaries, reconstruction interference is prevented from occurring between items in different events. A similar principle is present in the DP-KF, which predicts that slowly drifting sensory stimuli will be pooled together, whereas a punctate change-point will result in multiple segments (and thus, less memory regularization). This account is supported by human behavioral data showing less averaging of memory traces separated by a punctate change point (Gershman et al., 2014). In our account, events facilitate a better partitioning of the sensory experience by further relying on the dynamics of the process (which is inclusive of slowly drifting processes). This partitioning leads to better reconstructive memory.

Our memory model shares several features with other memory models. SEM is conceptually similar to the 'schema-copy plus tag' model, in which a memory trace is composed of a corrupted copy of a schema and a set of schema a-typical features (Graesser & Nakamura, 1982) The perspective of memory retrieval as inference over a noisy memory trace is shared with the REM model (Shiffrin & Steyvers, 1997) and the regularizing effect of the event dynamics is qualitatively similar to the category induced bias proposed by Huttenlocher et al. (1991). The model is also similar to the temporal context model (TCM; Howard, Fotedar, Datey, & Hasselmo, 2005; Howard & Kahana, 2002) and the related context maintenance and retrieval model (CMR; Polyn et al., 2009) in that the posterior over events can be interpreted as a context that changes over time (see also Socher et al., 2009). In TCM, a context vector that encodes an average of recently seen items is bound to each new memory item, a strategy similar to grouping the items of an event together in a set of learned dynamics. The CMR model adds to this the property that task changes induce large context shifts, which partitions traces from different contexts, another feature present in our model.

Nevertheless, SEM offers a novel and complementary approach to reconstructive memory that is meaningfully different from prior memory models. Broadly, reconstructive memory can be thought of as the process by which we reconstruct the past from fragmentary recollections. Neisser (1967) made a famous analogy about how reconstructive memory is like a paleontologist making a dinosaur from dug-up bones. Existing models (e.g., Norman & O'Reilly, 2003; Raaijmakers & Shiffrin, 1980; Shiffrin & Steyvers, 1997), are concerned with which bones are found (how a memory fragment is retrieved), whereas with SEM we take the perspective of the paleontologist and ask how do we put the pieces together. These are two complementary problems in memory and as such SEM complements other models of memory.

This difference between the reconstruction process of SEM and the generation of memory traces of prior models is similar, but not equivalent, to the distinction between semantic knowledge and episodic memory. The noisy memory trace SEM uses in the reconstruction process can roughly be thought of as a form of episodic memory whereas the event dynamics are more akin to semantic knowledge used to organize the during reconstruction. This comparison breaks down when we consider that novel events have episodic-like qualities; event dynamics for a newly learned event capture much more detail about the specific event than would be expected for an event that has been experienced multiple times. As an event is experienced multiple times, the learned dynamics come to reflect the more generalize dynamics and average away the specific features of an individual event. This leads to a transition of an episodic-like event dynamic to a semantic-like event over time.

Neural Correlates

A further open question is how our model of events is implemented in the brain. A growing body of evidence has linked the hippocampus and the posterior medial network to event segmentation and memory (Ranganath & Ritchey, 2012). The hippocampus plays a crucial role binding features and contextual information (Davachi, 2006; Diana, Yonelinas, & Ranganath, 2007; Eacott & Gaffan, 2005; Eichenbaum,

Yonelinas, & Ranganath, 2007; Knierim, Lee, & Hargreaves, 2006; Ranganath, 2010), and could therefore play a role in binding features of an event. Consistent with this hypothesis, recognizing objects across an event boundary is associated with an increase of hippocampal activity (Swallow et al., 2011), and event boundaries themselves are linked to an increase in hippocampal signal (Baldassano et al., 2017).

The posterior medial network, a component of the default network, has anatomical connections to the hippocampus and encompasses the parahippocampal, posterior cingulate, retrosplenial, ventromedial prefrontal cortex, precuneus and the angular gyrus (Ranganath & Ritchey, 2012). Hippocampal and PM regions consistently co-activate during recall of experienced events, recollection of context, retrieval of temporal sequences, imagination of future events, and spatial navigation (Ranganath & Ritchey, 2012; Rugg & Vilberg, 2013; Spreng & Grady, 2010). A linking thread through these tasks is the need for a constructed event representation. Consistent with this idea, fMRI and electrocorticography studies have shown that regions of the PM network integrates over long time scales (hundreds of seconds) (Hasson, Chen, & Honey, 2015), and PM network activity is modulated by event boundaries (Kurby & Zacks, 2008). Furthermore, event boundaries are identifiable in the angular gyrus and posterior medial cortex in unsupervised fMRI analysis (Baldassano et al., 2017). In terms of memory, medial prefrontal cortex brain activity has been associated with schema-consistent memory (van Kesteren et al., 2013), further implicating the posterior medial network in event processing. Integrating over these findings, we propose that event dynamics could be learned and represented by the PM network (Figure 12). We further propose that the ventromedial prefrontal cortex (vmPFC) encodes the posterior probability distribution over event models, as the vmPFC has been previously implicated in latent cause inference (Chan, Niv, & Norman, 2016) and the state representation (Schuck, Cai, Wilson, & Niv, 2016; Wilson, Takahashi, Schoenbaum, & Niv, 2014), two similar computational problems. Finally, we hypothesize that the PM-hippocampal interactions are responsible for the reconstructive memory process, as this process involved the reinstantiation of scene dynamics with memory traces.

Hence, we would expect to see these regions modulated by events in different ways as a function of the experimental paradigm. For example, we would expect to see stronger modulation of the vmPFC in tasks where the strength of evidence for each individual event model varies dramatically. Likewise, we might expect to see modulation in the hippocampus and PM network in tasks that manipulate event reconstruction but less modulation of vmPFC in such a task. We also might expect the relationship between event segmentation and subsequent memory measures to increase following sleep, reflecting prior research linking sleep and the increased semanticization of memory (Dudai, Karni, & Born, 2015), and we would expect this to be reflected by hippocampal replay. Although all of these predictions are strongly motivated by the existing literature, our neural predictions are an open empirical question for future research.

A related issues is how the computational-level model relates to a circuit-level implementation. We note that many key aspects of our theoretical account are neurally plausible. Of primary interest is how event dynamics are learned by the PM system. One component of this problem is learning an effective representational space, which can dramatically simplify the problem of learning event dynamics. While the exact representational format of scenes is an open question, Richmond and Zacks (2017) argued that we should expect these representations to be smooth over time, as event dynamics are consequently easier to learn and generalize. This mirrors techniques in statistics and machine learning, where transformation from a complex representational space to a simpler, intermediate representation is a commonly used tool (Bishop, 2006).

To learn the dynamics over these representations, we have used recurrent neural networks as a function approximator. Gating mechanisms, the ability to selectively store the internal state of the network, are a critical element of these networks and make them well-suited to learning sequential dependencies (Hochreiter & Schmidhuber, 1997; LeCun et al., 2015). Gating mechanisms are common in biologically inspired models of human rule learning and reinforcement learning (Collins & Frank, 2013; Kriete, Noelle, Cohen, & O'Reilly, 2013; O'Reilly & Frank, 2006; Rougier, Noelle, Braver, Cohen, & O'Reilly, 2005; Wang et al., 2018) and hypothesized to be supported

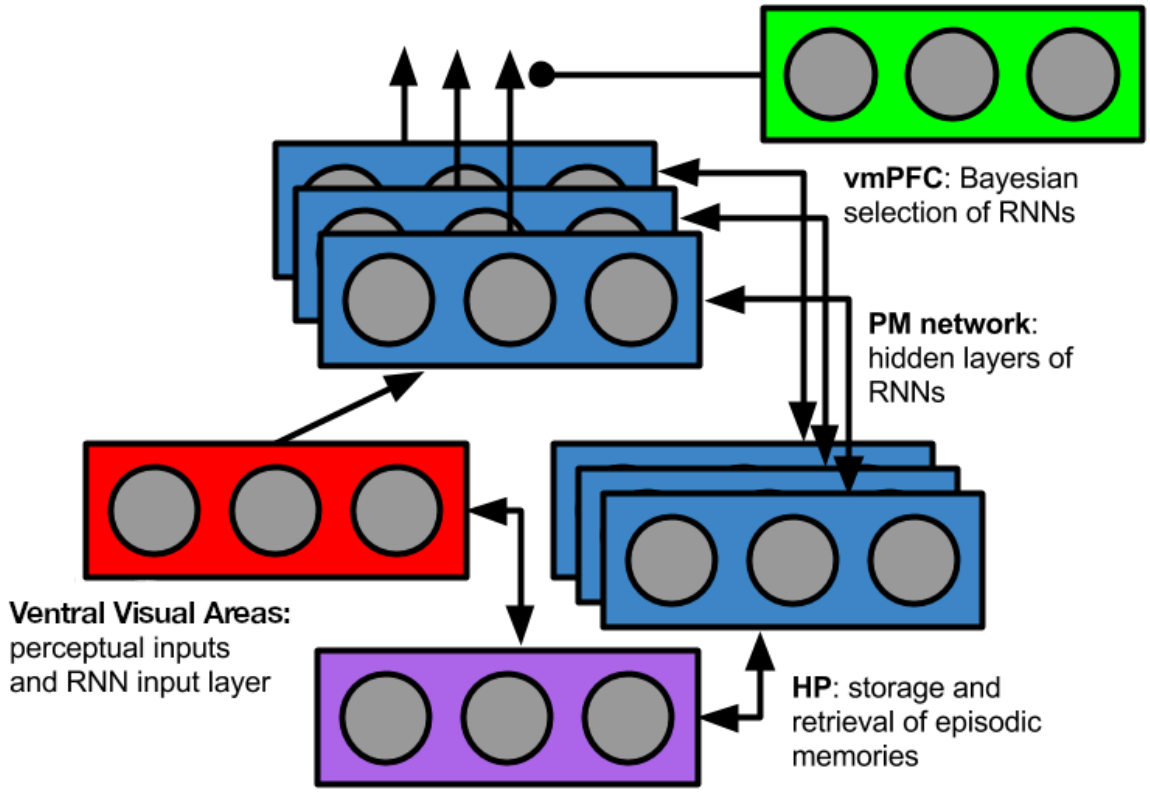

Figure 12. **Neural Correlates of SEM** Diagram of SEM architecture with corresponding brain regions. Information about entities from the ventral visual areas is fed into event models, which are instantiated in the PM network (blue) as recurrent neural networks (RNNs). vmPFC (green) is hypothesized to select the currently relevant event schema/RNN. The hippocampus is hypothesized to support storage and retrieval of event-specific information (i.e., activity patterns in the currently-selected RNN).

by the midbrain dopamine system (Collins & Frank, 2013; Frank & Badre, 2011; O'Reilly & Frank, 2006). In line with these models, one possibility is that the midbrain dopamine system (including the ventral tegmental area and substantia nigra) provides an error signal for learning event dynamics through a gating mechanism similar to the ones found in artificial neural networks. This fits with the generalized view of dopamine prediction errors recently espoused by Gardner, Schoenbaum, and Gershman (2018). As an alternative possibility, the PM network may learn the event dynamics directly through local connectivity and its own prediction errors. How this may occur is an open question, but prior computational modeling work has suggested that gating mechanisms

are consistent with cortical microcircuits (Costa, Assael, Shillingford, de Freitas, & Vogels, 2017) and that neural oscillations may be sufficient to generate a training signal (O'Reilly, Wyatte, & Rohrlich, 2014).

Language and other outstanding questions

In the current work, we have not considered tasks that rely on comprehension of natural language texts. In the event cognition literature, many of the effects that have been observed use narrative texts in their experimental design and measure reading speeds (for review, see Radvansky & Zacks, 2014). For example, subjects show slower reading speeds following a change in goals (Suh & Trabasso, 1993) or causes (Zwaan, Langston, & Graesser, 1995) in narrative texts, an effect that has previously been interpreted as reflecting an event boundary (Radvansky & Zacks, 2014). Memory for specific sentences read in a narrative declines across time even as the memory of the events described remains stable (Fletcher & Chrysler, 1990; Kintsch et al., 1990; Schmalhofer & Glavanov, 1986). In principle, SEM could be applied to natural language texts as long as these texts could be encoded into the logical scene description language that we embed into vector space using HRRs.

More broadly, while our model of events included the ability to encode structured representations, it is limited in its scope of semantic knowledge. We made this choice so that we may better isolate the effects of events on perception and memory from that of semantic knowledge, but ultimately we hope to outfit SEM with a richer semantic database. For example, in our simulations on the video data from Zacks et al. (2006), each of the videos that we examined concerned a single person completing an everyday action. Our purpose was to demonstrate that SEM could generate human-like segmentation from pixel-level data via unsupervised training alone and without access to hand-tuned representations of objects. Nonetheless, the effectiveness of SEM on this task was likely influenced by the simplicity of the videos. A video in which several actors cooperate on a task, for example, or a video with background motion (cars in the distance, birds moving around) might be much more difficult for the computational

model. The model as implemented has no way of telling what the subject of a video is and that it should ignore other irrelevant changes in the features. This limitation becomes clear if we were to try to model natural language tasks, as the model would require the semantic knowledge to properly represent the narratives. A full cognitive model of event cognition would contain access to this structured semantic knowledge, as well as a library of previously encountered events.

References

Altmann, G. T., & Kamide, Y. (1999). Incremental interpretation at verbs: Restricting the domain of subsequent reference. *Cognition*, *73*(3), 247–264.

Altmann, G. T., & Mirković, J. (2009). Incrementality and prediction in human sentence processing. *Cognitive science*, *33*(4), 583–609.

Anderson, J. R. (1974). Retrieval of propositional information from long-term memory. *Cognitive psychology*, *6*(4), 451–474.

Anderson, J. R. (1991). The adaptive nature of human categorization. *Psychological review*, *98*(3), 409-429.

Aslin, R. N., Saffran, J. R., & Newport, E. L. (1998). Computation of conditional probability statistics by 8-month-old infants. *Psychological science*, *9*(4), 321–324.

Avrahami, J., & Kareev, Y. (1994). The emergence of events. *Cognition*, *53*(3), 239–261.

Axmacher, N., Cohen, M. X., Fell, J., Haupt, S., Dümpelmann, M., Elger, C. E., ... Ranganath, C. (2010). Intracranial eeg correlates of expectancy and memory formation in the human hippocampus and nucleus accumbens. *Neuron*, *65*(4), 541–549.

Baldassano, C., Chen, J., Zadbood, A., Pillow, J. W., Hasson, U., & Norman, K. A. (2017). Discovering event structure in continuous narrative perception and memory. *Neuron*, *95*(3), 709–721.

Baldassano, C., Hasson, U., & Norman, K. A. (2018). Representation of real-world event schemas during narrative perception. *bioRxiv*, 252718.

Baldwin, D., Andersson, A., Saffran, J., & Meyer, M. (2008). Segmenting dynamic human action via statistical structure. *Cognition*, *106*(3), 1382–1407.

Bartlett, F. C. (1932). *Remembering: A study in experimental and social psychology* (Vol. 14). Cambridge University Press.

Bellman, R. (1961). *Adaptative control processes.* Princeton University Press.

Bishop, C. M. (2006). *Pattern recognition and machine learning.* Springer.

Botvinick, M., & Bylsma, L. M. (2005). Regularization in short-term memory for serial order. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *31*(2), 351.

Bower, G. H., Black, J. B., & Turner, T. J. (1979). Scripts in memory for text. *Cognitive psychology*, *11*(2), 177–220.

Brady, T. F., Konkle, T., & Alvarez, G. A. (2009). Compression in visual working memory: Using statistical regularities to form more efficient memory representations. *Journal of Experimental Psychology: General*, *138*(4), 487-502.

Bransford, J. D., Barclay, J. R., & Franks, J. J. (1972). Sentence memory: A constructive versus interpretive approach. *Cognitive psychology*, *3*(2), 193–209.

Butz, M. V., Bilkey, D., Humaidan, D., Knott, A., & Otte, S. (2018). Learning, planning, and control in a monolithic neural event inference architecture. *arXiv preprint arXiv:1809.07412*.

Chan, S. C., Niv, Y., & Norman, K. A. (2016). A probability distribution over latent causes, in the orbitofrontal cortex. *Journal of Neuroscience*, *36*(30), 7817–7828.

Cho, K., Van Merriënboer, B., Bahdanau, D., & Bengio, Y. (2014). On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*.

Chollet, F., et al. (2015). *Keras.* `https://keras.io`.

Cohen, N. J., Poldrack, R. A., & Eichenbaum, H. (1997). Memory for items and memory for relations in the procedural/declarative memory framework. *Memory*, *5*(1-2), 131–178.

Cohn, N., Jackendoff, R., Holcomb, P. J., & Kuperberg, G. R. (2014). The grammar of visual narrative: Neural evidence for constituent structure in sequential image comprehension. *Neuropsychologia*, *64*, 63–70.

Collins, A. G., & Frank, M. J. (2013). Cognitive control over learning: Creating, clustering, and generalizing task-set structure. *Psychological review*, *120*(1), 190-229.

Costa, R., Assael, I. A., Shillingford, B., de Freitas, N., & Vogels, T. (2017). Cortical

microcircuits as gated-recurrent neural networks. In *Advances in neural information processing systems* (pp. 272–283).

Davachi, L. (2006). Item, context and relational episodic encoding in humans. *Current opinion in neurobiology*, *16*(6), 693–700.

Diana, R. A., Yonelinas, A. P., & Ranganath, C. (2007). Imaging recollection and familiarity in the medial temporal lobe: a three-component model. *Trends in cognitive sciences*, *11*(9), 379–386.

Doersch, C. (2016). Tutorial on variational autoencoders. *arXiv preprint arXiv:1606.05908*.

Doumas, L. A., & Hummel, J. E. (2005). A symbolic-connectionist model of relation discovery. In *Proceedings of the annual meeting of the cognitive science society* (Vol. 27).

DuBrow, S., & Davachi, L. (2013). The influence of context boundaries on memory for the sequential order of events. *Journal of Experimental Psychology: General*, *142*(4), 1277–1286. doi: 10.1037/a0034024

DuBrow, S., & Davachi, L. (2016). Temporal binding within and across events. *Neurobiology of Learning and Memory*, *134*, 107–114. doi: 10.1016/j.nlm.2016.07.011

Dudai, Y., Karni, A., & Born, J. (2015). The consolidation and transformation of memory. *Neuron*, *88*(1), 20–32.

Duvenaud, D., Lloyd, J. R., Grosse, R., Tenenbaum, J. B., & Ghahramani, Z. (2013). Structure discovery in nonparametric regression through compositional kernel search. *arXiv preprint arXiv:1302.4922*.

Eacott, M., & Gaffan, E. (2005). The roles of perirhinal cortex, postrhinal cortex, and the fornix in memory for objects, contexts, and events in the rat. *The Quarterly Journal of Experimental Psychology Section B*, *58*(3-4), 202–217.

Ehrlich, S. F., & Rayner, K. (1981). Contextual effects on word perception and eye movements during reading. *Journal of verbal learning and verbal behavior*, *20*(6), 641–655.

Eichenbaum, H., Yonelinas, A. P., & Ranganath, C. (2007). The medial temporal lobe and recognition memory. *Annu. Rev. Neurosci.*, *30*, 123–152.

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

Elman, J. L., & McRae, K. (2019). A model of event knowledge. *Psychological Review*.

Ezzyat, Y., & Davachi, L. (2011). What constitutes an episode in episodic memory? *Psychological Science*, *22*(2), 243–252.

Fiser, J., & Aslin, R. N. (2001). Unsupervised statistical learning of higher-order spatial structures from visual scenes. *Psychological science*, *12*(6), 499–504.

Fiser, J., & Aslin, R. N. (2002). Statistical learning of higher-order temporal structure from visual shape sequences. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *28*(3), 458.

Fletcher, C. R., & Chrysler, S. T. (1990). Surface forms, textbases, and situation models: Recognition memory for three types of textual information. *Discourse Processes*, *13*(2), 175–190.

Fodor, J. A., & Pylyshyn, Z. W. (1988). Connectionism and cognitive architecture: A critical analysis. *Cognition*, *28*(1-2), 3–71.

Fox, E., Sudderth, E., Jordan, M., & Willsky, A. (2010). Bayesian nonparametric methods for learning markov switching processes.

Frank, M. J., & Badre, D. (2011). Mechanisms of hierarchical reinforcement learning in corticostriatal circuits 1: computational analysis. *Cerebral cortex*, *22*(3), 509–526.

Franklin, N. T., & Frank, M. J. (2018). Compositional clustering in task structure learning. *PLoS computational biology*, *14*(4), e1006116.

Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition. In (p. 193-202). Springer.

Gardner, M. P., Schoenbaum, G., & Gershman, S. J. (2018). Rethinking dopamine as generalized prediction error. *Proceedings of the Royal Society B*, *285*(1891), 20181645.

Gelman, A., Stern, H. S., Carlin, J. B., Dunson, D. B., Vehtari, A., & Rubin, D. B. (2013). *Bayesian data analysis*. Chapman and Hall/CRC.

Geman, S., Bienenstock, E., & Doursat, R. (1992). Neural networks and the bias/variance dilemma. *Neural computation*, *4*(1), 1–58.

Gershman, S. J., & Blei, D. M. (2012). A tutorial on bayesian nonparametric models. *Journal of Mathematical Psychology*, *56*(1), 1–12.

Gershman, S. J., Radulescu, A., Norman, K. A., & Niv, Y. (2014). Statistical computations underlying the dynamics of memory updating. *PLoS computational biology*, *10*(11), e1003939.

Ghahramani, Z., & Hinton, G. E. (1996). *Switching state-space models* (Tech. Rep.). Citeseer.

Godden, D. R., & Baddeley, A. D. (1975). Context-dependent memory in two natural environments: On land and underwater. *British Journal of psychology*, *66*(3), 325–331.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press. (`http://www.deeplearningbook.org`)

Goodman, N. D., Ullman, T. D., & Tenenbaum, J. B. (2011). Learning a theory of causality. *Psychological review*, *118*(1), 110.

Graesser, A. C., & Nakamura, G. V. (1982). The impact of a schema on comprehension and memory. In *Psychology of learning and motivation* (Vol. 16, pp. 59–109). Elsevier.

Graesser, A. C., Singer, M., & Trabasso, T. (1994). Constructing inferences during narrative text comprehension. *Psychological review*, *101*(3), 371-395.

Halford, G. S., Wilson, W. H., & Phillips, S. (1998). Processing capacity defined by relational complexity: Implications for comparative, developmental, and cognitive psychology. *Behavioral and Brain Sciences*, *21*(6), 803–831.

Hanson, C., & Hirst, W. (1989). On the representation of events: A study of orientation, recall, and recognition. *Journal of Experimental Psychology: General*, *118*(2), 136-147.

Hard, B. M., Tversky, B., & Lang, D. S. (2006). Making sense of abstract events: Building event schemas. *Memory & cognition*, *34*(6), 1221–1235.

Hare, M., Elman, J. L., Tabaczynski, T., & McRae, K. (2009). The wind chilled the spectators, but the wine just chilled: Sense, structure, and sentence comprehension. *Cognitive Science*, *33*(4), 610–628.

Hasson, U., Chen, J., & Honey, C. J. (2015). Hierarchical process memory: memory as an integral component of information processing. *Trends in cognitive sciences*, *19*(6), 304–313.

Hemmer, P., & Steyvers, M. (2009). A bayesian account of reconstructive memory. *Topics in Cognitive Science*, *1*(1), 189–202.

Heusser, A. C., Ezzyat, Y., Shiff, I., & Davachi, L. (2018). Perceptual boundaries cause mnemonic trade-offs between local boundary processing and across-trial associative binding. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 1075–1090.

Hinton, G. E., McClelland, J. L., & Rumelhart, D. E. (1986). *Distributed representations* (D. E. Rumelhart & M. J. L, Eds.). Carnegie-Mellon University Pittsburgh, PA.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735–1780.

Hou, X., Shen, L., Sun, K., & Qiu, G. (2017). Deep feature consistent variational autoencoder. In *2017 ieee winter conference on applications of computer vision* (pp. 1133–1141).

Howard, M. W., Fotedar, M. S., Datey, A. V., & Hasselmo, M. E. (2005). The temporal context model in spatial navigation and relational learning: toward a common explanation of medial temporal lobe function across domains. *Psychological review*, *112*(1), 75-116.

Howard, M. W., & Kahana, M. J. (2002). A distributed representation of temporal context. *Journal of Mathematical Psychology*, *46*(3), 269–299.

Hummel, J. E., & Holyoak, K. J. (2003). A symbolic-connectionist theory of relational inference and generalization. *Psychological review*, *110*(2), 220-264.

Huttenlocher, J., Hedges, L. V., & Duncan, S. (1991). Categories and particulars:

prototype effects in estimating spatial location. *Psychological review*, *98*(3), 352–376.

James, W. (1890). The principles of. *Psychology*, *2*, 94.

Kahana, M. J. (1996). Associative retrieval processes in free recall. *Memory & cognition*, *24*(1), 103–109.

Kemp, C., & Tenenbaum, J. B. (2008). The discovery of structural form. *Proceedings of the National Academy of Sciences*, 10687–10692.

Kemp, C., Tenenbaum, J. B., Niyogi, S., & Griffiths, T. L. (2010). A probabilistic model of theory formation. *Cognition*, *114*(2), 165–196.

Kietzmann, T. C., McClure, P., & Kriegeskorte, N. (2018). Deep neural networks in computational neuroscience. *bioRxiv*, 133504.

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kingma, D. P., & Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Kintsch, W., Welsch, D., Schmalhofer, F., & Zimny, S. (1990). Sentence memory: A theoretical analysis. *Journal of Memory and language*, *29*(2), 133–159.

Knierim, J. J., Lee, I., & Hargreaves, E. L. (2006). Hippocampal place cells: parallel input streams, subregional processing, and implications for episodic memory. *Hippocampus*, *16*(9), 755–764.

Kriete, T., Noelle, D. C., Cohen, J. D., & O'Reilly, R. C. (2013). Indirection and symbol-like processing in the prefrontal cortex and basal ganglia. *Proceedings of the National Academy of Sciences*, 201303547.

Kurby, C. A., & Zacks, J. M. (2008). Segmentation in the perception and memory of events. *Trends in cognitive sciences*, *12*(2), 72–79.

Kurby, C. A., & Zacks, J. M. (2011). Age differences in the perception of hierarchical structure in events. *Memory & cognition*, *39*(1), 75–91.

Kutas, M., & Hillyard, S. A. (1984). Brain potentials during reading reflect word expectancy and semantic association. *Nature*, *307*(5947), 161—163.

Lashley, K. S. (1951). *The problem of serial order in behavior* (Vol. 21). Bobbs-Merrill.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, *521*(7553), 436.

Lisman, J. E., & Grace, A. A. (2005). The hippocampal-vta loop: controlling the entry of information into long-term memory. *Neuron*, *46*(5), 703–713.

Maas, A. L., Hannun, A. Y., & Ng, A. Y. (2013). Rectifier nonlinearities improve neural network acoustic models. In *Proceedings of the 30th international conference on machine learning.*

MacKay, D. J. (2003). *Information theory, inference and learning algorithms.* Cambridge university press.

Matsuki, K., Chow, T., Hare, M., Elman, J. L., Scheepers, C., & McRae, K. (2011). Event-based plausibility immediately influences on-line language comprehension. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *37*(4), 913-934.

McClelland, J. L., McNaughton, B. L., & O'reilly, R. C. (1995). Why there are complementary learning systems in the hippocampus and neocortex: insights from the successes and failures of connectionist models of learning and memory. *Psychological review*, *102*(3), 419-457.

McKoon, G., & Ratcliff, R. (1992). Inference during reading. *Psychological review*, *99*(3), 440-466.

McRae, K., Hare, M., Elman, J. L., & Ferretti, T. (2005). A basis for generating expectancies for verbs from nouns. *Memory & Cognition*, *33*(7), 1174–1184.

McRae, K., & Matsuki, K. (2009). People use their knowledge of common events to understand language, and do so as quickly as possible. *Language and linguistics compass*, *3*(6), 1417–1429.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems* (pp. 3111–3119).

Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., ... Kavukcuoglu, K. (2016). Asynchronous methods for deep reinforcement learning.

In *International conference on machine learning* (pp. 1928–1937).

Nassar, M. R., & Frank, M. J. (2016). Taming the beast: extracting generalizable knowledge from computational models of cognition. *Current opinion in behavioral sciences*, *11*, 49–54.

Neisser, U. (1967). *Cognitive psychology: Classic edition*. Psychology Press.

Nelson, A. B., & Shiffrin, R. M. (2013). The co-evolution of knowledge and event memory. *Psychological Review*, *120*(2), 356.

Newtson, D. (1973). Attribution and the unit of perception of ongoing behavior. *Journal of Personality and Social Psychology*, *28*(1), 28-38.

Newtson, D., & Engquist, G. (1976). The perceptual organization of ongoing behavior. *Journal of Experimental Social Psychology*, *12*(5), 436–450.

Nissen, M. J., & Bullemer, P. (1987). Attentional requirements of learning: Evidence from performance measures. *Cognitive psychology*, *19*(1), 1–32.

Norman, K. A., & O'Reilly, R. C. (2003). Modeling hippocampal and neocortical contributions to recognition memory: a complementary-learning-systems approach. *Psychological review*, *110*(4), 611–646.

O'Reilly, R. C., & Frank, M. J. (2006). Making working memory work: a computational model of learning in the prefrontal cortex and basal ganglia. *Neural computation*, *18*(2), 283–328.

O'Reilly, R. C., Wyatte, D., & Rohrlich, J. (2014). Learning through time in the thalamocortical loops. *arXiv preprint arXiv:1407.3432*.

Pettijohn, K. A., & Radvansky, G. A. (2016). Walking through doorways causes forgetting: Event structure or updating disruption? *The Quarterly Journal of Experimental Psychology*, *69*(11), 2119–2129.

Pettijohn, K. A., Thompson, A. N., Tamplin, A. K., Krawietz, S. A., & Radvansky, G. A. (2016). Event boundaries and memory improvement. *Cognition*, *148*, 136–144. doi: 10.1016/j.cognition.2015.12.013

Plate, T. A. (1995). Holographic reduced representations. *IEEE Transactions on Neural networks*, *6*(3), 623–641.

Polyn, S. M., Norman, K. A., & Kahana, M. J. (2009). A context maintenance and retrieval model of organizational processes in free recall. *Psychological review*, *116*(1), 129.

Raaijmakers, J. G., & Shiffrin, R. M. (1980). Sam: A theory of probabilistic search of associative memory. In *Psychology of learning and motivation* (Vol. 14, pp. 207–262). Elsevier.

Rabiner, L. R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, *77*(2), 257–286.

Radford, A., Metz, L., & Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*.

Radvansky, G. A. (2012). Across the event horizon. *Current Directions in Psychological Science*, *21*(4), 269–272.

Radvansky, G. A., & Copeland, D. E. (2006). Walking through doorways causes forgetting: Situation models and experienced space. *Memory & cognition*, *34*(5), 1150–1156.

Radvansky, G. A., Krawietz, S. A., & Tamplin, A. K. (2011). Walking through doorways causes forgetting: Further explorations. *Quarterly journal of experimental psychology*, *64*(8), 1632–1645.

Radvansky, G. A., Tamplin, A. K., & Krawietz, S. A. (2010). Walking through doorways causes forgetting: Environmental integration. *Psychonomic bulletin & review*, *17*(6), 900–904.

Radvansky, G. A., & Zacks, J. M. (2011). Event perception. *Wiley Interdisciplinary Reviews: Cognitive Science*, *2*(6), 608–620.

Radvansky, G. A., & Zacks, J. M. (2014). *Event cognition*. Oxford University Press.

Radvansky, G. A., & Zacks, J. M. (2017). Event boundaries in memory and cognition. *Current opinion in behavioral sciences*, *17*, 133–140.

Ranganath, C. (2010). Binding items and contexts: The cognitive neuroscience of episodic memory. *Current Directions in Psychological Science*, *19*(3), 131–137.

Ranganath, C., & Rainer, G. (2003). Cognitive neuroscience: Neural mechanisms for detecting and remembering novel events. *Nature Reviews Neuroscience*, *4*(3), 193.

Ranganath, C., & Ritchey, M. (2012). Two cortical systems for memory-guided behaviour. *Nature Reviews Neuroscience*, *13*(10), 713-726.

Reber, A. S. (1989). Implicit learning and tacit knowledge. *Journal of experimental psychology: General*, *118*(3), 219-235.

Reynolds, J. R., Zacks, J. M., & Braver, T. S. (2007). A computational model of event segmentation from perceptual prediction. *Cognitive Science*, *31*(4), 613–643.

Richmond, L. L., & Zacks, J. M. (2017). Constructing experience: Event models from perception to action. *Trends in cognitive sciences*, *21*(12), 962–980.

Robertson, E. M. (2007). The serial reaction time task: implicit motor skill learning? *Journal of Neuroscience*, *27*(38), 10073–10075.

Roediger, H. L., & McDermott, K. B. (1995). Creating false memories: Remembering words not presented in lists. *Journal of experimental psychology: Learning, Memory, and Cognition*, *21*(4), 803-814.

Rougier, N. P., Noelle, D. C., Braver, T. S., Cohen, J. D., & O'Reilly, R. C. (2005). Prefrontal cortex and flexible cognitive control: Rules without symbols. *Proceedings of the National Academy of Sciences*, *102*(20), 7338–7343.

Rugg, M. D., & Vilberg, K. L. (2013). Brain networks underlying episodic memory retrieval. *Current opinion in neurobiology*, *23*(2), 255–260.

Saffran, J. R., Aslin, R. N., & Newport, E. L. (1996). Statistical learning by 8-month-old infants. *Science*, *274*(5294), 1926–1928.

Saffran, J. R., Johnson, E. K., Aslin, R. N., & Newport, E. L. (1999). Statistical learning of tone sequences by human infants and adults. *Cognition*, *70*(1), 27–52.

Sargent, J. Q., Zacks, J. M., Hambrick, D. Z., Zacks, R. T., Kurby, C. A., Bailey, H. R., ... Beck, T. M. (2013). Event segmentation ability uniquely predicts event memory. *Cognition*, *129*(2), 241–255.

Schank, R. C., & Abelson, R. P. (1977). *Scripts, plans, goals, and understanding: An inquiry into human knowledge structures*. Psychology Press.

Schapiro, A. C., Rogers, T. T., Cordova, N. I., Turk-Browne, N. B., & Botvinick, M. M. (2013). Neural representations of events arise from temporal community structure. *Nature neuroscience*, *16*(4), 486–492.

Schiffer, A.-M., & Schubotz, R. I. (2011). Caudate nucleus signals for breaches of expectation in a movement observation paradigm. *Frontiers in Human Neuroscience*, *5*, 38.

Schmalhofer, F., & Glavanov, D. (1986). Three components of understanding a programmer's manual: Verbatim, propositional and situational representations. *Journal of Memory and Language*, *25*, 279–294.

Schuck, N. W., Cai, M. B., Wilson, R. C., & Niv, Y. (2016). Human orbitofrontal cortex represents a cognitive map of state space. *Neuron*, *91*(6), 1402–1412.

Schulz, E., Tenenbaum, J. B., Duvenaud, D., Speekenbrink, M., & Gershman, S. J. (2017). Compositional inductive biases in function learning. *Cognitive psychology*, *99*, 44–79.

Schwanenflugel, P. J., & Shoben, E. J. (1985). The influence of sentence constraint on the scope of facilitation for upcoming words. *Journal of Memory and Language*, *24*(2), 232-252.

Shiffrin, R. M., & Steyvers, M. (1997). A model for recognition memory: Rem—retrieving effectively from memory. *Psychonomic bulletin & review*, *4*(2), 145–166.

Siegelmann, H. T., & Sontag, E. D. (1991). Turing computability with neural nets. *Applied Mathematics Letters*, *4*(6), 77–80.

Smith, D. A., & Graesser, A. C. (1981). Memory for actions in scripted activities as a function of typicality, retention interval, and retrieval task. *Memory & Cognition*, *9*(6), 550–559.

Smith, S. M. (1982). Enhancement of recall using multiple environmental contexts during learning. *Memory & Cognition*, *10*(5), 405–412.

Smith, S. M. (1984). A comparison of two techniques for reducing context-dependent forgetting. *Memory & Cognition*, *12*(5), 477–482.

Smith, S. M., & Rothkopf, E. Z. (1984). Contextual enrichment and distribution of practice in the classroom. *Cognition and Instruction*, *1*(3), 341–358.

Smolensky, P. (1990). Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial intelligence*, *46*(1-2), 159–216.

Socher, R., Gershman, S., Sederberg, P., Norman, K., Perotte, A. J., & Blei, D. M. (2009). A Bayesian analysis of dynamics in free recall. In *Advances in neural information processing systems* (pp. 1714–1722).

Speer, N. K., Zacks, J. M., & Reynolds, J. R. (2007). Human brain activity time-locked to narrative event boundaries. *Psychological Science*, *18*(5), 449–455.

Spreng, R. N., & Grady, C. L. (2010). Patterns of brain activity supporting autobiographical memory, prospection, and theory of mind, and their relationship to the default mode network. *Journal of cognitive neuroscience*, *22*(6), 1112–1123.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, *15*(1), 1929–1958.

Suh, S., & Trabasso, T. (1993). Inferences during reading: Converging evidence from discourse analysis, talk-aloud protocols, and recognition priming. *Journal of memory and language*, *32*(3), 279-300.

Swallow, K. M., Barch, D. M., Head, D., Maley, C. J., Holder, D., & Zacks, J. M. (2011). Changes in Events Alter How People Remember Recent Information. *Journal of cognitive neuroscience*, *23*(5), 1052–1064. doi: 10.1162/jocn.2010.21524

Trabasso, T., & Van Den Broek, P. (1985). Causal thinking and the representation of narrative events. *Journal of memory and language*, *24*(5), 612–630.

Tversky, B., Zacks, J. M., & Hard, B. M. (2008). The structure of experience. *Understanding events*, 436–464.

van Kesteren, M. T., Beul, S. F., Takashima, A., Henson, R. N., Ruiter, D. J., & Fernández, G. (2013). Differential roles for medial prefrontal and medial temporal

cortices in schema-dependent encoding: from congruent to incongruent. *Neuropsychologia*, *51*(12), 2352–2359.

Wang, J. X., Kurth-Nelson, Z., Kumaran, D., Tirumala, D., Soyer, H., Leibo, J. Z., ... Botvinick, M. (2018). Prefrontal cortex as a meta-reinforcement learning system. *Nature neuroscience*, *21*(6), 860.

Wang, J. X., Kurth-Nelson, Z., Tirumala, D., Soyer, H., Leibo, J. Z., Munos, R., ... Botvinick, M. (2016). Learning to reinforcement learn. *arXiv preprint arXiv:1611.05763*.

Wilson, R. C., Takahashi, Y. K., Schoenbaum, G., & Niv, Y. (2014). Orbitofrontal cortex as a cognitive map of task space. *Neuron*, *81*(2), 267–279.

Yamins, D. L., & DiCarlo, J. J. (2016). Using goal-driven deep learning models to understand sensory cortex. *Nature neuroscience*, *19*(3), 356.

Zacks, J. M., Kurby, C. A., Eisenberg, M. L., & Haroutunian, N. (2011). Prediction error associated with the perceptual segmentation of naturalistic events. *Journal of Cognitive Neuroscience*, *23*(12), 4057–4066.

Zacks, J. M., Speer, N. K., Swallow, K. M., Braver, T. S., & Reynolds, J. R. (2007). Event perception: a mind-brain perspective. *Psychological bulletin*, *133*(2), 273–293.

Zacks, J. M., Speer, N. K., Vettel, J. M., & Jacoby, L. L. (2006). Event understanding and memory in healthy aging and dementia of the alzheimer type. *Psychology and aging*, *21*(3), 466–482.

Zacks, J. M., Tversky, B., & Iyer, G. (2001). Perceiving, remembering, and communicating structure in events. *Journal of Experimental Psychology: General*, *130*(1), 29–58. doi: 10.1037/0096-3445.130.1.29

Zhao, S., Song, J., & Ermon, S. (2017). Infovae: Information maximizing variational autoencoders. *arXiv preprint arXiv:1706.02262*.

Zwaan, R. A. (1994). Effect of genre expectations on text comprehension. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *20*(4), 920–933.

Zwaan, R. A., Langston, M. C., & Graesser, A. C. (1995). The construction of situation

models in narrative comprehension: An event-indexing model. *Psychological science*, *6*(5), 292–297.

Appendix A

Variational Autoencoder

A variational auto-encoder (Doersch, 2016; Kingma & Welling, 2013) is a dimensionality reduction technique that employs two networks, an "encoder" and a "decoder" network, to project the data into a lower-dimensional embedding space and re-project it into the original space. A Gaussian prior is defined over the embedding space and the network is trained to minimize the difference between the original image and its reconstruction, subject to the embedding prior. We used a variant of the Deep Convolutional Generative Adversarial Networks (DCGAN) architecture (Radford et al., 2015) for our encoder and decoder networks and use a maximum mean discrepancy kernel as a prior over our embedding space (Zhao, Song, & Ermon, 2017). Our encoder network consisted of a five layers: three convolutional layers (with 64×3, 128×3 and 256×3 channels, respectively) and two fully connected layers. All of the layers used a leaky, rectified linear activation function except the last layer, which was linear and 100-dimensional. Our decoder network reversed this process with a symmetrical network. Prior to training, each frame was downsampled to a resolution of 64x64 using linear interpolation. Randomly selected batches of frames were used with the ADAM optimization algorithm (Kingma & Ba, 2014) to train the autoencoder. A PyTorch implementation of our neural network is available at https://github.com/ProjectSEM/VAE-video.

Appendix B

Holographic reduced representation

Holographic reduced representations (HRR) leaves intact the similarity structure of the composed vectors. HRRs consist of two operations, vector addition and circular convolution. Vector addition preserves similarity, such that if $\mathbf{a}$, $\mathbf{b}$ and $\mathbf{c}$ are vectors, then then $(\mathbf{a}+\mathbf{c})$ and $(\mathbf{b}+\mathbf{c})$ are typically more similar to each other than $\mathbf{a}$ and $\mathbf{b}$ are to each other. This is always true for zero-mean orthogonal vectors and is true in the expectation for zero-mean random vectors in high dimensional space (e.g. $\mathbf{x} \sim \mathcal{N}(0, I)$) To demonstrate this, compare the orthogonal vectors $\mathbf{a}$, $\mathbf{b}$ and $\mathbf{c}$ with dot product as our measure of vector similarity, then $\mathbf{a}^{\mathrm{T}}\mathbf{b} = 0$ while $(\mathbf{a}+\mathbf{c})^{\mathrm{T}}(\mathbf{b}+\mathbf{c}) = \mathbf{c}^{\mathrm{T}}\mathbf{c}$. Likewise, if we assume the two vectors $\mathbf{x}$ and $\mathbf{y}$ are similar to each other such that $\mathbf{x}^{\mathrm{T}}\mathbf{y} \neq 0$, then adding orthogonal features to them does not change their similarity. If $\mathbf{x}$ and $\mathbf{y}$ are also orthogonal to $\mathbf{a}$ and $\mathbf{b}$, then $(\mathbf{x}+\mathbf{a})^{\mathrm{T}}(\mathbf{y}+\mathbf{b}) = \mathbf{x}^{\mathrm{T}}\mathbf{y}$. Thus adding random feature vectors does not change the similarity between two vectors.

Using circular convolution as a binding operation also preserves the similarity structure. That is, if two vectors are similar to each other in vector space, then their convolution with a third vector will retain that similarity. We can show this by approximating a circular convolution with a tensor product (Doumas & Hummel, 2005; Plate, 1995), and noting that tensor operations are distributive, such that $(\mathbf{x}+\mathbf{y}) \otimes \mathbf{z} = \mathbf{x} \otimes \mathbf{z} + \mathbf{y} \otimes \mathbf{z}$. If two vectors $\mathbf{a}$ and $\mathbf{b}$ share a common factor $\mathbf{c}$ such that $\mathbf{a} = \mathbf{a}_0 + \mathbf{c}$, and $\mathbf{b} = \mathbf{b}_0 + \mathbf{c}$, where $\mathbf{a}_0$, $\mathbf{b}_0$ and $\mathbf{c}$ are orthogonal vectors, we can decompose their tensor product with $\mathbf{d}$ into the sum of two separate tensors

$$\mathbf{a} \otimes \mathbf{d} = \mathbf{a}_0 \otimes \mathbf{d} + \mathbf{c} \otimes \mathbf{d}$$

and

$$\mathbf{b} \otimes \mathbf{d} = \mathbf{b}_0 \otimes \mathbf{d} + \mathbf{c} \otimes \mathbf{d}$$

provided that $\mathbf{d}$ is orthogonal to $\mathbf{a}$ and $\mathbf{b}$. Because both tensors products share a common tensor as a linear factor, we can use the arguments above to show that they are similar to each other. Thus, taking the tensor product of two similar vectors and a

third random vector will result in two similar tensor product. Since circular convolution is a compressed tensor product operation (Doumas & Hummel, 2005; Plate, 1995), this argument will hold generally for HRRs as well.

Appendix C

Sampling reconstruction memory model

In order to sample the reconstructive memory model, we define a Gibbs sampling algorithm. The algorithm has three interlocking pieces: (1) sampling reconstructed estimates of the scene features $\hat{\mathbf{x}}$ conditioned on samples of $\hat{y}$ and $\hat{e}$, (2) sampling estimates of $\hat{y}$ from conditioned on samples of $\hat{\mathbf{x}}$ and $\hat{e}$ and (3) sampling estimates of $\hat{e}$ conditioned on samples of $\hat{\mathbf{x}}$ and $\hat{y}$. As each component is conditionally independent of the other, they can be iteratively sampled until convergence.

To initialize, at each time n we either draw a sample of $\hat{y}_n = (\hat{\mathbf{x}}, \hat{e}, n)$ from either $\tilde{y}$ without replacement or assign it to be y_0. The features of the scenes $\hat{\mathbf{x}}$ are initialized from a normal distribution and the sequence of events e is initialized from the sCRP prior.

Sampling $\Pr(\mathbf{x}|\tilde{y}, e)$. In the first step of the sampling algorithm, we draw samples of the scene features conditioned on the corrupted memory trace $\tilde{y}$ and the event label e. The probability of the feature vector $\mathbf{x}'$ occurring at time n can be recursively defined under the generative model as:

$$\Pr(\mathbf{x}_n|\mathbf{x}_{1:n-1}, \tilde{y}_n, e_n, \theta) = \mathcal{N}(\bar{\mathbf{x}}, \lambda\mathbf{I}) \tag{19}$$

where the mean $\bar{\mathbf{x}}_n$ is a precision-weighted linear combination of the scene features of the memory trace $\tilde{\mathbf{x}}_n$ from the corrupted memory trace $\tilde{y}_n$ and the predicted location of the embedded scene under the event model:

$$\bar{\mathbf{x}}_n = uf(\mathbf{x}_{1:n-1}, e_t, \theta) + (1-u)\tilde{\mathbf{x}}_n \tag{20}$$

where $u = \beta^{-1}/(\beta^{-1} + \tau^{-1})$ and $\lambda = 1/(\beta^{-1} + \tau^{-1})$ are the mixture weights of the mean and uncertainty of the distribution, respectively.

Sampling $\Pr(\tilde{y}|\mathbf{x}, e)$. The samples of the features vectors are then used to sample $\tilde{y}$ and e. The corrupted memory traces $\tilde{y}$ are sampled to determine both the time the occur as well as whether they are recalled at all. Samples of $\tilde{y}$ are drawn from the conditional distribution of the memory trace given the reconstructed features and event

labels, which is defined by inverting the corruption processes (Equations 12, 13 and 14):

$$\Pr(\tilde{y}_i|\mathbf{x}_n, e_n) = \begin{cases} 0 & \text{if } e_i \notin \{e_n, e_0\} \\ \epsilon/Z & \text{if } \tilde{y}_i = y_0 \\ \mathbb{1}_{|\tilde{n}-n|<b}\mathcal{N}(\tilde{\mathbf{x}}'; \mathbf{x}_n, \tau\mathrm{I})/Z & \text{otherwise} \end{cases} \tag{21}$$

where $\tilde{y}_i = (\tilde{\mathbf{x}}', \tilde{e}', \tilde{n})$, and where $\mathbb{1}_{|\tilde{n}-n|<b}$ is an indicator function that returns 1 if $|\tilde{n} - n| < b$ and 0 otherwise. Thus, the probability the corrupted memory item occurs at time n is zero if the corrupted event label is mismatched, or if the corrupted time index $\tilde{n}$ is more than b steps away from n.

It is important to note that the probability of sampling a null event under this process is both a function of ϵ and the estimate of $\mathbf{x}_n$. This is because the normalizing constant,

$$Z = \epsilon + \sum_{\tilde{y}_i} \mathbb{1}_{|\tilde{n}-n|<b}\mathcal{N}(\tilde{\mathbf{x}}'; \mathbf{x}_n, \tau\mathrm{I}) \tag{22}$$

which will typically be larger as a function of $||\hat{\mathbf{x}}_n - \mathbf{x}_n||$, thus lowering the probability of a 'null' event. Consequently, the quality of the reconstructed features will influence the degree to which a 'null' token is included. It is also worth noting that the use of a uniform prior over a restricted range greatly simplifies the sampling problem when compared to, for example, a discretized normal distribution. Because we assume the each memory token $\tilde{y}_i$ is sampled without replacement, there are less than $2N^{2b+1}$ valid assignments of $\hat{y}$, where as an unbounded time corruption process leads to $2N!$ valid permutations.

Sampling $\Pr(e|\mathbf{x}, \tilde{y})$**.** To complete the inference process, the samples of $\hat{\mathbf{x}}$ and $\hat{y}$ are used to update the estimate of the sequence of events. Conditioned on $\mathbf{x}$, f and θ, the probability of events $e_{1:t}$ under the generative process is recursively defined as:

$$\Pr\left(e|\mathbf{x}, e_{1:n-1}\right) \propto \Pr(\mathbf{x}_t|\mathbf{x}_{1:n-1}, e) \Pr(e = k|e_{1:n-1}) \tag{23}$$

As previously noted, we store a corrupted memory of the event label $\tilde{e}_i$ in the memory trace $\tilde{y}_i$. Conditioning on this memory trace, the reconstruction process is thus:

$$\Pr(e|\tilde{y}_i, \mathbf{x}, e_{n-1}) = \begin{cases} \Pr\left(e|\mathbf{x}_{1:n-1}, e_{n-1},\right) & \text{if } \tilde{e}' = e_0 \text{ or } \tilde{y}_i = y_0 \\ 1 & \text{otherwise} \end{cases} \tag{24}$$

If the associated event memory $\tilde{e}'$ is equal to the 'null' token e_0, or if $\tilde{y}_i$ is the 'null' memory item y_0, then the generative process (Equation 23) is sampled. Otherwise, the memory label is taken from the estimated memory trace $\hat{y}_n$ In the full Gibbs sampler, we draw alternatively draw samples of $\hat{\mathbf{x}}$, $\hat{y}$ and $\hat{e}$ until convergence. To this sample, we can apply different scoring rules to simulate different memory measures. For example, we model recall by whether a trace $\tilde{y}$ is in the final sample. These measures are specific to each simulation and we present each them with the relevant simulation.

Appendix D

Parameter values

Simulation	Embedding Dim.	ν_0	s_0^2	λ	α	β	τ	ϵ
Zacks et al. (2006)	100	10	0.3	10^5	0.1	-	-	-
Schapiro et al. (2013)	25	1	1.0	10^5	1.0	-	-	-
Structured Event Boundaries	266	1.0	0.444	1.0	1.0	2	0.1	0.25
Bower et al. (1979)	185	10.	0.24	1.0	10.0	2	0.15	0.75
Radvansky & Copeland (2006)	100	25	0.2	1.0	1.0	2	0.1	0.25/0.75
Pettijohn & Radvansky (2016)	25	100	0.2	1.0	1.0	2	0.1	0.25
DuBrow & Davachi (2013, 2016)	100	25	0.2	1.0	1.0	2	0.1	0.25

Table D1

Model Parameter values *The parameter values for each simulation is listed separately in a row.*

	learning rate	β_1	β_2	ϵ	decay	n_batches
Parameter	0.01	0.9	0.999	1e-07	0.0	200

Table D2

Adam Parameters values *used to train Gated-Recurrent units to estimate event dynamics. Each simulation used the optimization algorithm to estimate their event dynamics.*