Remembering the "When": Hebbian Memory Models for the Time of Past Events

Johanni Brea, Alireza Modirshanechi, Wulfram Gerstner

November 28, 2022

Abstract

Humans and other species can remember how long ago specific events happened. Based on a systematic exploration of neural coding and association schemes, we develop different hypotheses about how neural networks with Hebbian plasticity could enable an automatic reconstruction of the time of past events. We assess these hypotheses from a computational perspective and in light of the available behavioral and neurophysiological data. Existing models of how the time of past events can be remembered are special cases of our theory. We conclude that many models lead to indistinguishable behavior and future experiments with neural recordings are needed to disambiguate different models.

1 Introduction

Humans and animals track temporal information on multiple time scales, to estimate, for example, the location of a sound source based on milliseconds time differences of sound arrival at the two ears, the interval duration between the perception of lightning and thunder, or the days, months or years that has elapsed since an autobiographical event took place [1, 2, 3]. For autobiographical memories, recall of the "when"-information is often an explicit and conscious reconstruction-based process [4] but there is also evidence for automatic processes, in particular in young infants without a fully developed episodic memory system [5, 6]. Also corvids, rodents, and other species with an episodic-like, what-where-when memory can remember the time of past events on time scales of days to months [6].

On time scales from milliseconds to minutes, multiple mechanisms based on changing neuronal activity patterns are known to support accurate interval timing [1, 2, 3]. Less is known about the neuronal mechanisms that support the recall of the time of past events on much longer times scales, but they rely presumably on synaptic plasticity and possibly on systems consolidation [7]. Different research communities have developed models of episodic memory to explain recall of past events [8]. These models focus on different aspects, like replicating behavioral data in free or serial recall of lists [9, 10, 11, 12], developing attractor neural networks consistent with anatomical and physiological knowledge of the hippocampal formation [13, 14, 8] or explaining systems consolidation [7, 15]. Although some of these models would allow the reconstruction of the elapsed time between storage and recall of events and attempts at categorizing different theories have been made [4, 16], there exists no systematic exploration with computational models that focus on the time of past events.

Here we study from a theoretical perspective different neural mechanisms that enable *automatic* reconstruction of the time of past events. Whereas theoretical considerations about the representation of temporal information are relevant for tracking time on any scale, we focus, in particular, on settings used to investigate episodic-like memory, where a stream of sensory inputs on a time scale

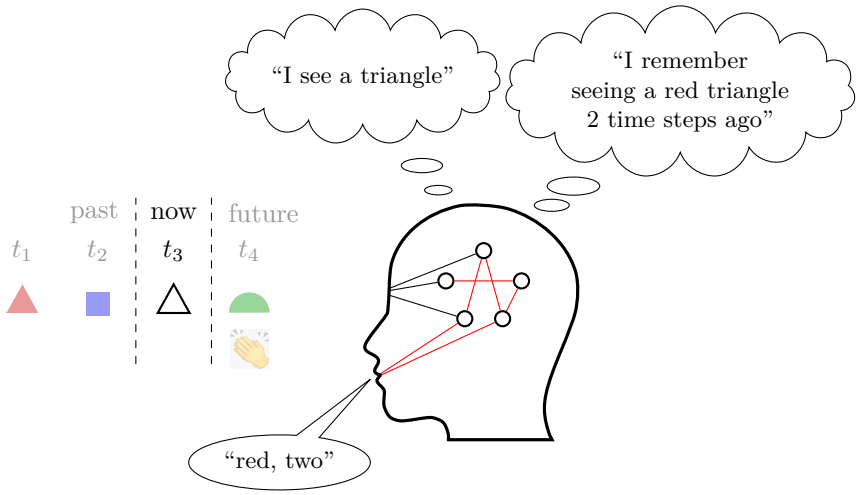

Figure 1: As a simple setting to study episodic-like memory, we consider discrete sensory streams, like perceiving a red triangle at time t_1 and a blue circle at time t_2. The time points $t_1, t_2, \ldots$ are not necessarily equally spaced and my be separated by hours or days. At time t_3, a triangle triggers, first, activity in the brain that corresponds to the perception of the triangle and, second, activity that corresponds to remembering the red triangle, including the information how long ago the red triangle was perceived. An action is performed upon memory retrieval, like saying "red, two". In the next time step, a new stimulus can be given, together with a reinforcement signal (clapping hands). Our goal is to find neural network dynamics and synaptic plasticity rules that change the connections between neurons (red lines) such that the sensory stream can be remembered and action selection rules that depend on the recalled event can be learned.

of days or months is perceived by an organism that can recall past events and respond with some actions (Figure 1). Typical examples are experiments where food caching birds learn to retrieve from caches they made the same day and ignore caches they made a few days ago [17] or ignore caches made on the same day and retrieve from older caches [18]. Our goal is to develop concrete hypotheses of biological neural circuits and Hebbian synaptic plasticity rules that support this behavior.

2 Results

2.1 Representing Information: the Space of Possible Codes

The sensory stream is in $\mathbb{R}^N \times \mathbb{R}$, where N is the dimensionality of the sensory state x_t, which evolves in time $t \in \mathbb{R}$. The "sensory state" here refers quite broadly to anything that could be stored in the episodic memory system, like observations of external events ("I see a red triangle") or internal events ("I feel excited"). For simplicity we consider a discretized version of the sensory stream in $\prod_{i=1}^N \mathcal{C}_i \times \mathcal{T}$, where $\mathcal{C}_i$ denotes the finite set of values sensor i can take and $\mathcal{T}$ denotes the finite set of possible time points.

Elements of products $\prod_{i=1}^N \mathcal{C}_i$ of finite sets $\mathcal{C}_i$ can be represented in multiple ways as neural activities. For a single finite set $\mathcal{C} = \{c_1, \ldots, c_{|\mathcal{C}|}\}$ with elements c_i and cardinality $|\mathcal{C}|$, we define three different elementary coding schemes: (i) a *rate code* is a scalar-valued function $\texttt{rate}(c_i) = r(i) \in \mathbb{R}$, where r is an arbitrary, injective function; (ii) a *one-hot code* is the $|\mathcal{C}|$-tuple-valued function $\texttt{onehot}$ with $\texttt{onehot}_j(c_i) = A_i \delta_{ij}$ for all $i, j \in \{1, \ldots, |\mathcal{C}|\}$, amplitude $A_i > 0$ and Kronecker delta $\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$ otherwise; and (iii) a *distributed code* as an injective M-tuple-valued func-

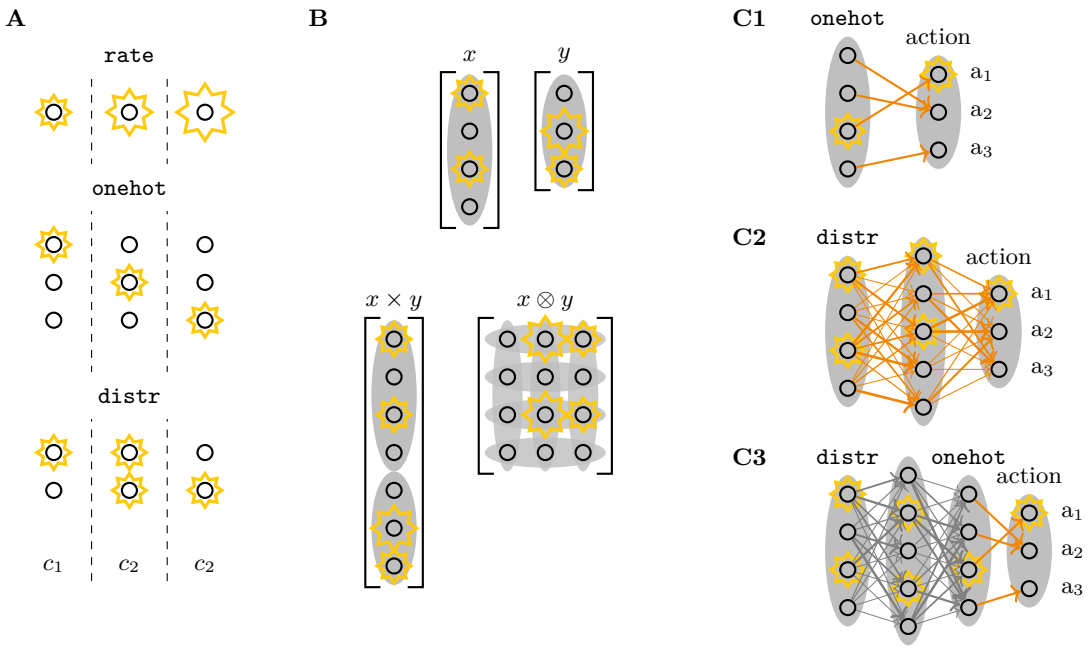

Figure 2: **Neuronal Encoding Schemes. A** The elements c_1, c_2, c_3 of a set C can be encoded in a rate code (top), one-hot code (middle) or a distributed code (bottom). **B** Code words x and y, encoding e.g. the content and the time of an event, can be **concatenated** ($x \times y$) or the **product** can be formed ($x \otimes y$). The two-dimensional arrangement of the twelve neurons in the product code is for visualization purposes; they could also be arranged in a vector with 12 elements. **C** The difficulty of learning flexible rules depends on the code. **C1 (direct readout):** For one-hot coding, any rule is learnable with direct connections to action neurons. **C2 (complex readout):** Learning arbitrary rules based on distributed or rate coding can be achieved with plastic connections in multilayer perceptrons or **C3 (complex readout with fixed preprocessing):** with hard-wired transformations into, for example, one-hot codes that allow flexible learning.

tion or random vector $\texttt{distr}$ with $M > 1$ and $0 < \texttt{distr}_j(c_i) \leq \texttt{distr}_{j'}(c_i)$ for at least one pair $j \neq j'$ and at least one i (Figure 2A). Certain distributed codes can be reduced to rate or one-hot codes by summing the activity of subsets of neurons. For example, the deterministic code $\texttt{distr}_j(c_i) = 1$ if $j \leq i$ and $\texttt{distr}_j(c_i) = 0$ otherwise or a stochastic code with probabilities $\Pr\big(\sum_j \texttt{distr}_j(c_i) = i\big) = 1$ can be transformed to the rate code $\texttt{rate}(c_i) = i = \sum_{j=1}^{|\mathcal{C}|} \texttt{distr}_j(c_i)$; and the redundant code $\texttt{distr}_j(c_i) = 1$ if $i - 1 < j/K \leq i$, for some group size K, and $\texttt{distr}_j(c_i) = 0$ otherwise, can be transformed to a one-hot code by summing over each group, i.e. $\texttt{onehot}_k(c_i) = K\delta_{ki} = \sum_{j=(k-1)K+1}^{kK} \texttt{distr}_j(c_i)$. Distributed codes that can be reduced by summing will be called *population rate codes* ($\texttt{poprate}$) or *population-hot codes* ($\texttt{pophot}$). In the following we use the term "distributed code" whenever such a reduction is not possible, as is for example the case for the binary representation of integers ($0000 \equiv 0, 0001 \equiv 1, 0010 \equiv 2, 0011 \equiv 3, \ldots$). We will write $x(\mathcal{C})$ for a set in x-coding, and $\ell(x(\mathcal{C}))$ or $\ell(x)$ for the length of an element of $\mathcal{C}$ in x-coding, e.g. $\ell(\texttt{rate}(\mathcal{C})) = 1, \ell(\texttt{onehot}(\mathcal{C})) = |\mathcal{C}|$.

For simplicity we restrict ourselves to the discussion of discretized versions of the sensory stream. However, for continuous versions one can easily find analogous results with, e.g. rate coding $\texttt{rate}(x) = x$ for some continuous variable x, an analog of one-hot coding with e.g. radial basis functions $\texttt{rbf}_j(x) = e^{-(j-x)^2}$ for a population of neurons with indices $j = 1, \ldots, M$ or an analog of some distributed code with e.g. mixtures of radial basis functions.

2.1.1 Static and Dynamic Representations of Time

We distinguish two ways of representing time: in *static* representations, time is measured relative to a fixed moment in time (e.g. the birth of an individual), and in *dynamic* representations, time is measured relative to changing moments in time (e.g. the current moment in time). In static representations of time, the neural activity during recall of a given event is always the same, no matter when recall happens. In dynamic representations of time, the neural activity during recall of a given event depends on how much time has elapsed between storage and recall.

2.2 Associating Information: How to Combine What, Where and When

The *concatenation of codes* is defined as $x \times y = (x_1, \ldots, x_{\ell(x)}, y_1, \ldots, y_{\ell(y)})$ for codes x and y (Figure 2B). The concatenation of codes has the property $\ell(x \times y) = \ell(x) + \ell(y)$. A generalization of concatenations of codes can be found in *mixed codes* $x \times_M y = M(x \times y)$, where M is a linear map. In mixed codes it may be impossible to identify any neuron as a pure x or a pure y neuron. Such a mixed representation is consistent with ideas of mixed selectivity [19] or criticism of a naive coding interpretations of neural data [20]. Association with concatenation codes can be implemented with synaptic plasticity in auto-associative [21] or hetero-associative [22] neural networks.

The *product of codes* is defined as $x \otimes y = x \cdot y_1 \times x \cdot y_2 \times \cdots \times x \cdot y_{\ell(x)}$, where the product $x \cdot y_i$ between a tuple x and a scalar y_i has the standard meaning $x \cdot y_i = (x_1 y_i, \ldots, x_{\ell(x)} y_i)$ (Figure 2B). The product of codes has the property $\ell(x \otimes y) = \ell(x) \cdot \ell(y)$. Concatenations of one-hot or rate codes lead to distributed codes, products of one-hot codes remain one-hot codes. The products of codes considered here are special cases of tensor product variable binding [23]. We also define *generalized product codes* $x \otimes_f y = f(x, y_1) \times f(x, y_2) \times \cdots \times f(x, y_{\ell(y)})$ for some function f.

Alternatively to associations with concatenation and product codes, (circular) convolutions can be used to associate x and y [24].

All association operations (concatenations, products or convolutions) can be combined to obtain complex codes like $((\texttt{onehot}(C_1) \times \texttt{onehot}(C_2) \times \texttt{rate}(C_3)) \otimes \texttt{onehot}(C_4)) \times \texttt{rate}(C_5)$.

Different aspects of the "what" of an event can be encoded and associated in different ways. Similarly, the "where" and the "when" can be associated with the "what" in multiple ways. In the following, we assume the "what" and the "where" are given as content x_t in some code and we focus exclusively on how the content x_t is associated with the "when" information.

2.3 Retrieving Information: How to Learn Rules

The code used for representing age and content of a recalled event has implications on the learnability of flexible rules. For a strict one-hot code, for example $\texttt{onehot}(\mathcal{C}) \otimes \texttt{onehot}(\mathcal{T})$, any rule that maps content and age of a recalled event to a given action can be learned with *direct readout*, i.e. plastic connections between the layer of recalled activity to action neurons (Figure 2C1). For distributed or rate coding, direct readout allows learning of some rules, but *complex readout* is needed to learn any rule. For example, the notorious XOR rule, where a certain action is taken if and only if two input neurons are jointly active or jointly inactive, cannot be learned with direct readout, but it can be learned with a multilayer perceptron (Figure 2C2). Learning all connections in a multilayer perceptron can be achieved with the backpropagation algorithm or biologically plausible variants thereof [25], but it is rather slow if learning happens in an online fashion, where each example is used just once. An alternative is to rely on fixed weights in most layers to transform the input into a useful feature representation and quickly learn flexible rules with plasticity in the last layer (Figure 2C3).

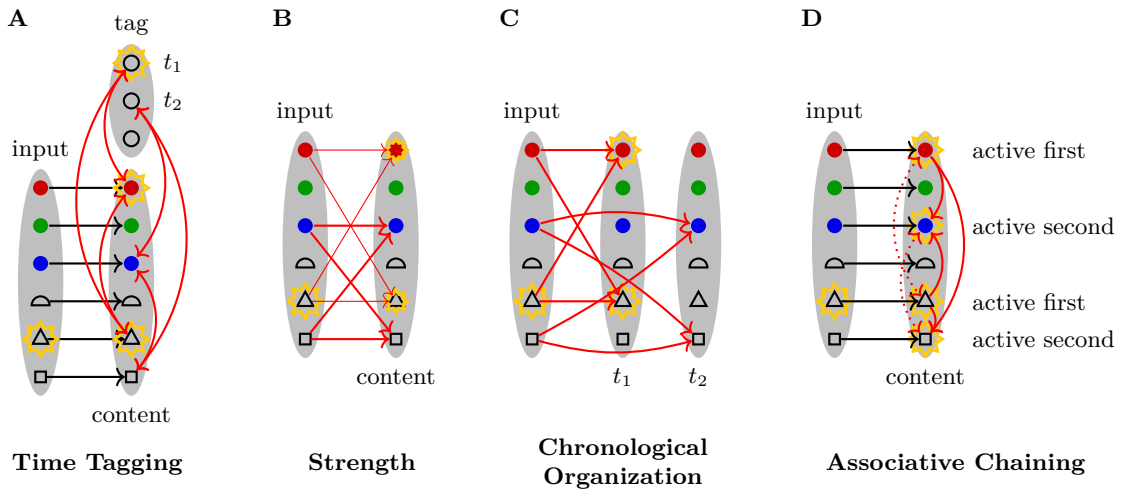

Figure 3: **Four examples of Episodic-Like Memory Systems.** Memory systems store a sequence of events (e.g. "red triangle", "blue square") in their synaptic connections and allow retrieval of the time of past events given an input(e.g. "triangle"). Arrows indicate synaptic connections; black: fixed, red: grown during memory encoding. Yellow stars indicate active neurons. **A (Time Tagging)**: The time (t_1) of the past event "red triangle" is encoded in the recalled activity pattern, i.e. active neuron with tag t_1. The connectivity pattern in this example enables auto-associative recall through the red recurrent weights. An alternative would be hetero-associative recall (see strength representation). **B (Strength)**: The time of past event is encoded in the strength of activity of the active neurons during recall. Recall is hetero-associative in this example, i.e. there are no recurrent connections, but only connections from the input layer to the target neurons. **C (Chronological Organization)**: The time of a past event is encoded in the identity of active neurons during recall; the active neurons are in assembly t_1. **D (Associative Chaining)**: The time of a past event is encoded in its temporal position in a sequence of activity patterns; dotted arrows provide pattern completion – we assume completion is almost instantaneous, such that the "triangle" neuron and the "red" neuron are active at the same time; solid red arrows allow activity to advance to the next element in the sequence.

2.4 Examples of Episodic-Like Memory Systems

In this section we sketch on an abstract level how the different coding and associations schemes lead to different models of episodic-like memory and we discuss their suitability for learning rules based on recalled events.

2.4.1 Time Tagging

Tagging models are characterized by concatenation of content and time information, i.e. activity patterns during recall of the form $x(\mathcal{C}) \times y(\mathcal{T})$, where $x(\mathcal{C})$ encodes the content (what-where) of the memory and $y(\mathcal{T})$ encodes the time information (when).

The time information can be given in two ways: in static tagging, the tag encodes the time point of storage (Figure 4A1), whereas in dynamic tagging, the tag is updated with the passage of time, such that the value of the tag encodes the time elapsed since storage (Figure 4A2, c.f. subsubsection 2.1.1). Dynamic tagging with rate coding is particularly simple to implement: decaying synaptic weights to the tag neuron are sufficient (Figure 5A). One-hot or distributed coding of elapsed time would require a more elaborate plasticity mechanism, e.g. the red neuron in Figure 4A2) targets another tag neuron in the second time step than in the first one.

Learning to take actions based on the age of memories for dynamic one-hot-coded tags can be achieved with direct readout, where plastic weights connect the tag neurons directly to some action or decision neurons (Figure 4B2). Dynamic rate or distributed tagging requires in general a complex readout mechanism, for example, using a multilayer perceptron (c.f. Figure 2C). Static tagging requires a comparison of the time at retrieval with the retrieved time of the past event. Learning flexible rules based on such a comparison requires even for one-hot-coded tags a complex readout mechanism (Figure 4B1). To learn general rules that depend both on the age of the retrieved memory and its content, both, dynamic and static tagging models need a complex readout mechanism, because the concatenation of tag and content necessarily leads to a distributed representation that may be subject to XOR-like problems.

Tags can be generated by intrinsic sources like oscillators that are independent of past experiences, extrinsic sources like the context in which an event occurs. We call the first case "intrinsically generated tags" and the second one "extrinsically generated tags". An example of intrinsically generated tags are synfire chains [26, 27], sequences of groups of neurons linked by feedforward connections, but synfire chains that run on a scale of milliseconds are costly to bridge events on the scale of minutes, hours or days. Combining oscillators of different frequencies is another way to create tags, with the potential to represent long durations (see e.g. the OSCAR model [28]), and it is possible to generate periodic orbits of duration that increases exponentially with the number of neurons in a recurrent neural networks [29]. In a static tagging approach, such a dedicated recurrent neural network could run like a clock or temporal backbone and provide the input to the "tag" neurons during storage and to the "now" neurons during retrieval.

Alternatively, synaptic dynamics could be used for intrinsically generated dynamic tagging. For example, connections to tag neurons formed during storage may decay with a constant rate at random moments in time; thus the number of active connections would allow to estimate roughly the elapsed time since storage (Figure 5A). Another example are static random context tags that are created by low-pass filtering a noise source (c.f. section 2.2.1 Random Context Models in [9]).

Extrinsically generated tags can be extracted from a weighted sum of past experiences. For example, in temporal context models (TCM) [9] and context maintenance and retrieval models (CMR) [10], static tags are obtained by low-pass filtering linearly transformed versions of the input stream, potentially using multiple low-pass filters with different time constants [30, 12] to approximate a power law [31].

In random or temporal context models the time information may be stored in a lossy manner, where the age of a memory can be retrieved only approximately, by comparing the similarity of the "now" tag with the retrieved tag (Figure 4B1).

2.4.2 Strength Models

In strength models (Figure 3B), the activity pattern of recalled memories has the form $x(\mathcal{C}) \otimes y(\mathcal{T})$ for $x \in \{\texttt{distr}, \texttt{onehot}\}$ and y a simple `rate` code or a (stochastic) population rate code. Time and content information could not be separated, if the content would also be stored in a rate code. The strength could be static or dynamic (c.f. subsubsection 2.1.1). In both cases, learning of flexible rules requires complex readout (c.f. Figure 2C); and in the case of static strengths a comparison with the rate code for the current moment is needed (c.f. Figure 4B1). Similarly to the synaptic implementation of intrinsically generated tags, we can construct a stochastic, dynamic, population rate model: events are stored by growing connections to multiple groups of neurons coding for the same content and connections decay at random moments in time (Figure 5B).

The representations of the content in different groups of neurons of a strength model (e.g. input group, intermediate group, memory group) do not need to be identical. Instead, each group could represent the content with a different transformation of the original code, for example $\tilde{z}^{(i)} = m^{(i)} z$

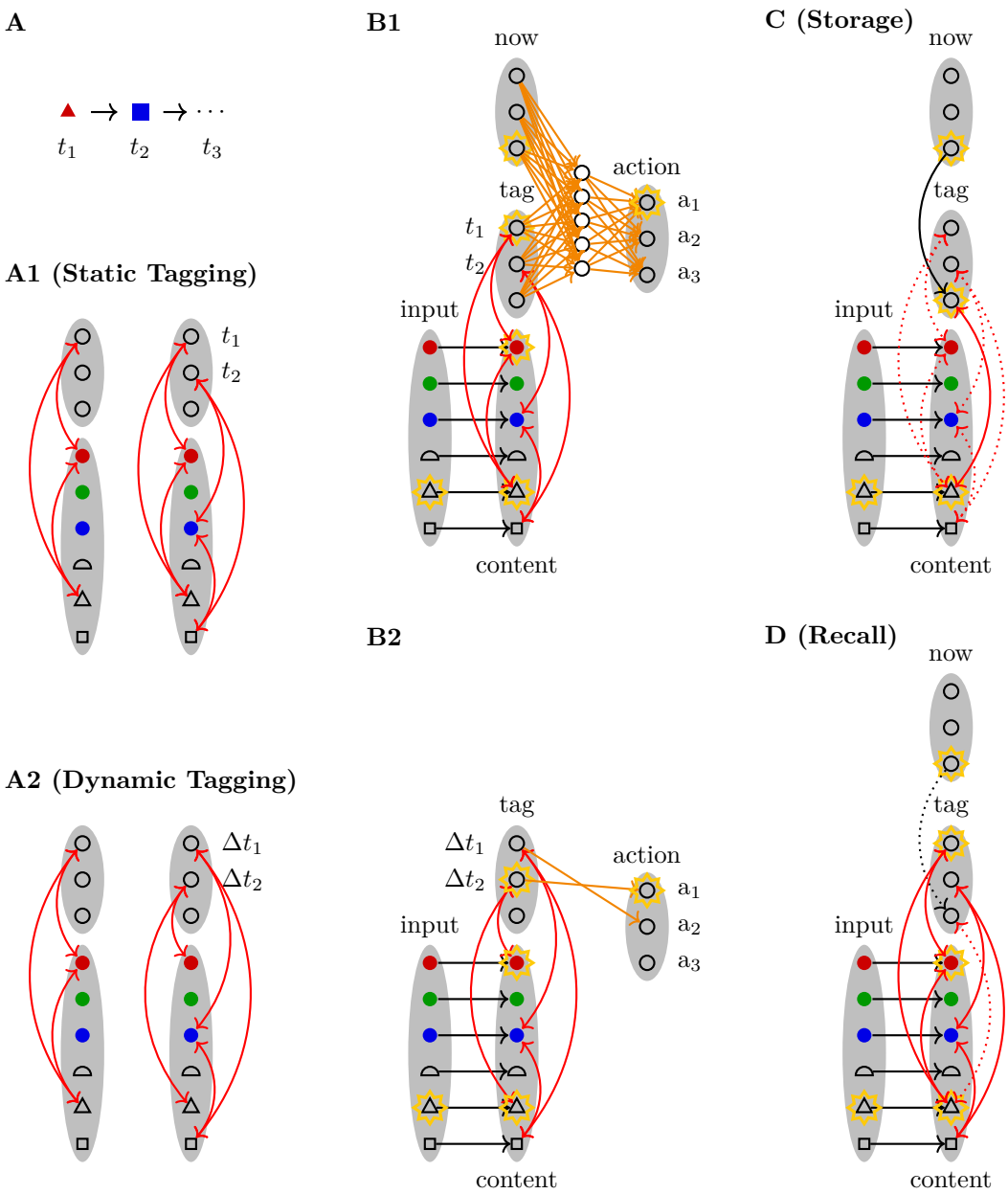

Figure 4: **Tagging Strategies.** **A** At time t_1 the event "red triangle" is observed and, then, at time t_2 the event "blue square" occurs. **A1** In static tagging, the association (indicated with red connections between "when" and "what" neurons) of "when" and "what" neurons does not change over time. **A2** In dynamic tagging, the associations between "when" and "what" neurons changes with the passage of time: initially the "red" and the "triangle" neurons are connected to the Δt_1 neuron, indicating "this just happened"; later, when the event "blue square" is added, the neurons encoding the first event are connected to the Δt_2 neuron. **B** Learning to respond with action a_1 when the retrieved memory has age Δt_2 and to respond with action a_2 when the retrieved memory as age Δt_1. **B1** In static tagging, there needs to be some comparison of the tag with a neural representation of the current moment in time (now). Learning flexible rules based on this comparison can be achieved with multi-layer perceptrons. **B2** In dynamic tagging, the age Δt of a memory is readily available in the activity of the tag neurons and flexible rules can simply be learned by changing synaptic weights between tag and action neurons. **C** In the storage phase, the tag and the content neurons could be driven by feedforward input. **D** In the recall phase, recurrent connections could retrieve the tag of the recalled event. Dotted lines indicate connections that are inactive in the given phase.

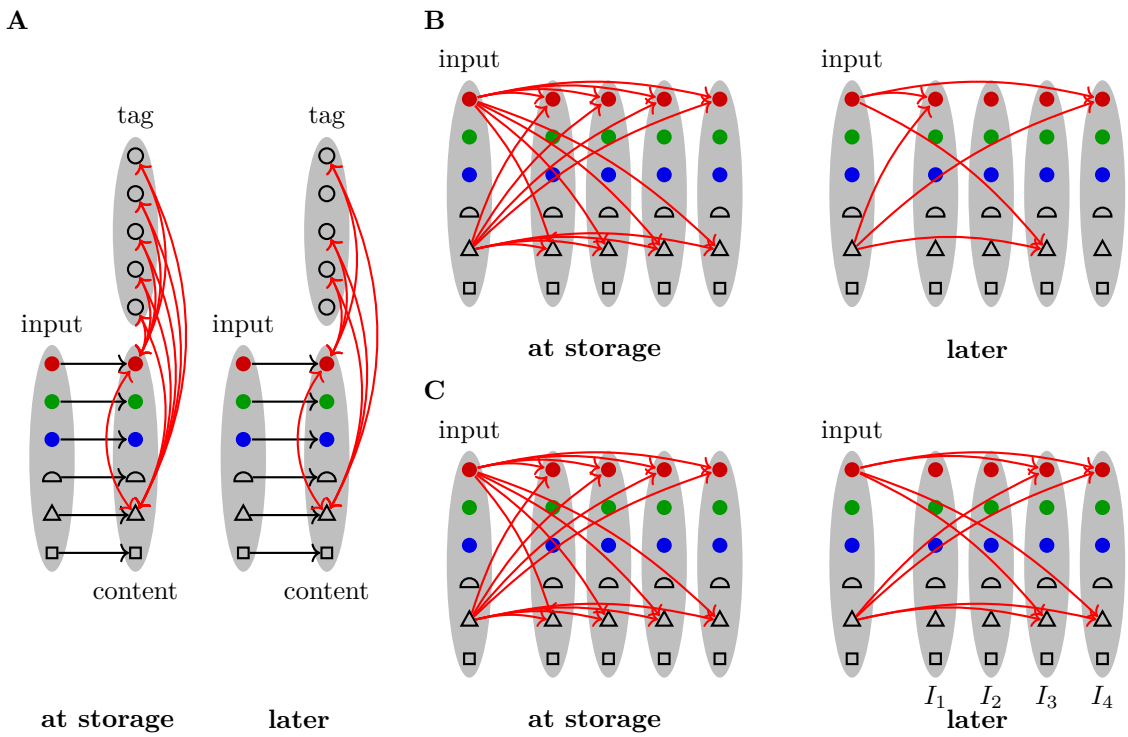

Figure 5: **Population Rate Codes with Synaptic Decay for Dynamic Tagging.** **A** At storage, connections to all tag neurons are formed. They decay at random moments in time. The number of active "when" neurons during recall encodes the elapsed time since storage: if many "when" neurons are active during recall, the recalled event happened recently and if few "when" neurons are active, it happened long ago. **B** In a strength model with stochastic population rate code, connections are formed to redundant content neurons during storage. They decay at random moments in time. As in A, the number of active "when" neurons during recall encodes the elapsed time since storage. **C** In a chronological organization model the connections decay with different rates in a spatially organized manner. Here, the identity of the active neurons during recall is informative about the elapsed time: neurons in group I_i are active, if the recalled event happened in the time interval $I_i = [0, \Delta t_i]$.

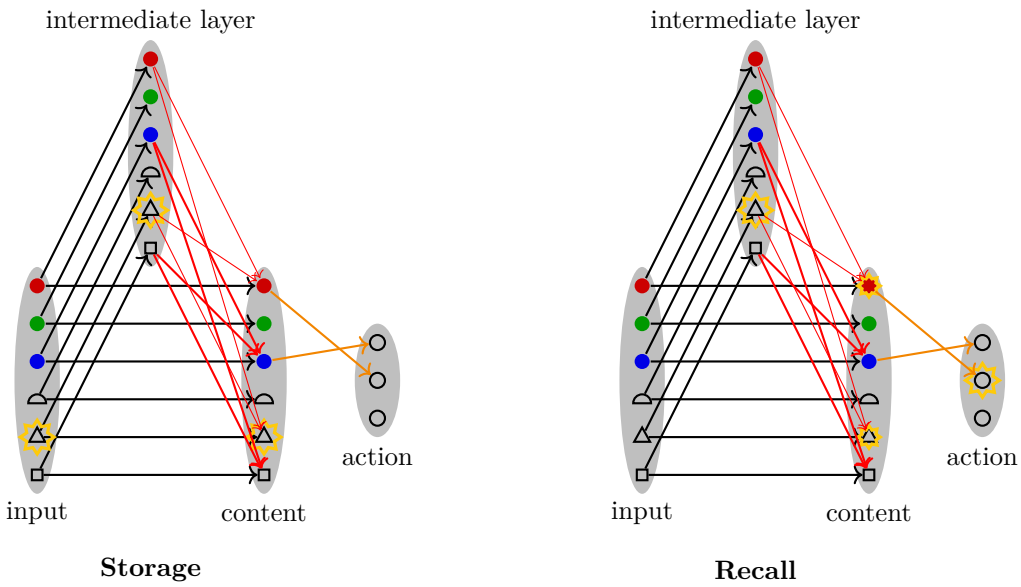

Figure 6: **Example of a Storage and Retrieval Mechanism for Strength and Chronological Organization Models**. During storage the input drives the activity in the intermediate and the content layer. Plastic connections between the intermediate and the content layer (red arrows) can grow with a Hebbian plasticity rule. Because of synaptic delays, the intermediate layer is shortly thereafter the main input of the content layer and already grown connections enable recall of previous events. In the recall phase, the content layer drives the action through weights that implement some learned rule (orange arrows).

for each group i and matrices $m^{(i)}$.

2.4.3 Chronological Organization

Chronological organization models are characterized by activity patterns where the identity of content-encoding neurons depends on time, e.g. $x(\mathcal{C}) \otimes y(\mathcal{T})$, where $x(\mathcal{C})$ encodes the content (what-where) of the memory and $y(\mathcal{T})$ encodes the time information (when), for y a `onehot`, a `distr`, a deterministic population rate or a population-hot code.

In statically organized memories, recall of a given event activates always the same neurons, independently of when recall happens (Figure 3C). An analogy for statically organized memories are the individual pictures on a movie tape, where recall of a given scene corresponds at all times to retrieval of an image at a given position on the movie tape. For storage and recall, statically organized memories need a mechanism to identify the groups of neurons that code for the current moment. For example, synfire chains or coupled oscillators could modulate the memory network during storage, such that only those neurons that code for the current moment in time can be activated by the event to be stored.

The active neurons during recall of a given event in dynamically organized memories depends on the time of recall, similarly to how suitcases on a conveyor belt change their position relative to an unmoved observer. A simple example with population rate codes is given in Figure 5C. At storage, synapses are grown to multiple groups of neurons that code for the same memory content. The connections targeting the different groups decay at different rates, such that the groups with active neurons during recall indicate how much time has elapsed since storage. Although this model is similar to the strength model with stochastic population rate code in Figure 5B, the

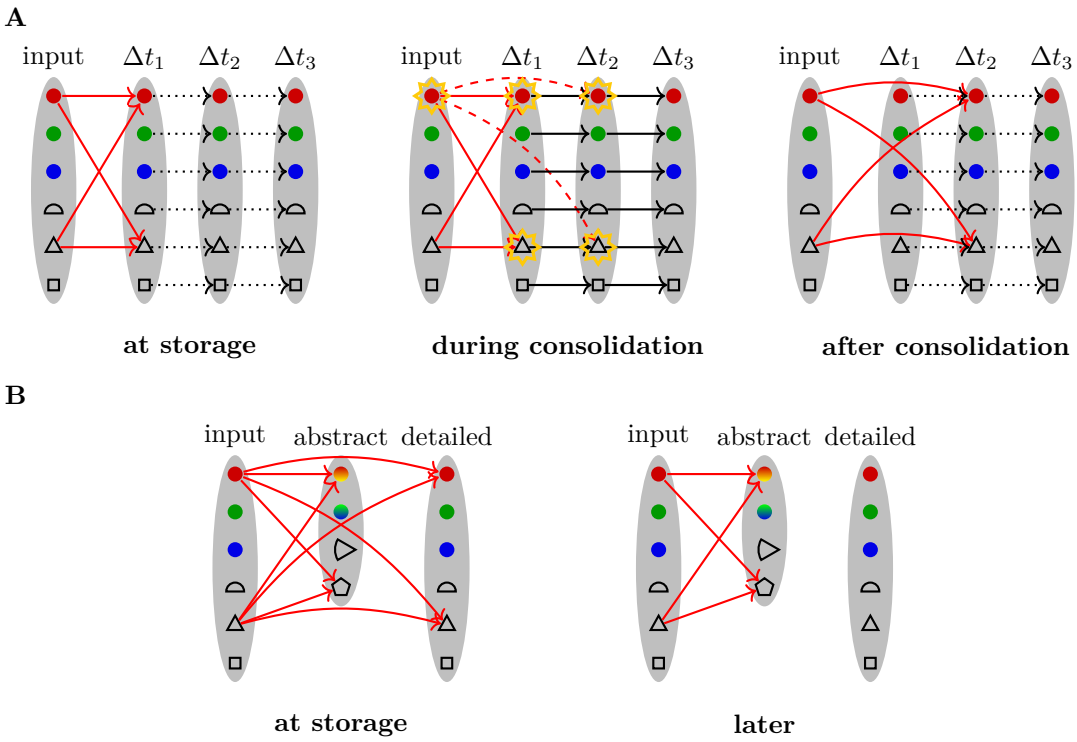

Figure 7: **Examples of Chronological Organization Models**. **A** In a dynamically organized model, a systems consolidation mechanism could move the location where a certain memory is stored. During storage, the feedforward weights (black dotted arrows) are inactive. During consolidation, input neurons are randomly active and activity is forward propagated (black arrows; indirect pathway from input to Δt_2), such that new, direct-pathway connections (dashed red) can grow between input and Δt_2 neurons. Simultaneously, the original weights between input and Δt_1 neurons decay, such that after consolidation only the new grown weights remain. **B** A lossy, dynamically organized model that stores initially detailed representations together with compressed, abstract representations (cold or hot color, polygons or disk segment) of the original event. The connections of the abstract representation decay slower than the connections of the detailed representation. Thus the age of a memory can be estimated by how detailed the recall is.

spatial organization of the decay rates gives it an advantage in learning flexible rules based on the age of memory: if the time since storage is encoded by linearly independent activity patterns $\texttt{distr}(\Delta t_1), \texttt{distr}(\Delta t_2), \ldots, \texttt{distr}(\Delta t_T)$, any rule that depends on $\Delta t_1, \ldots, \Delta t_T$ can be learned with a direct readout (Figure 2C1), as follows from the Kronecker-Capelli theorem and the fact that a matrix formed by T linearly independent vectors has full rank.

An alternative dynamic chronological organization depends on a systems consolidation process, where new synapses are grown to groups of neurons that code for older memories. For example, in consolidation phases during sleep, randomly activated input neurons could trigger recall of past events in neurons connected to the input by an indirect pathway, thereby allowing to learn direct-pathway connections (Figure 7A, c.f. parallel pathway theory [15]).

Independently of whether the passage of time leads to the growth of new synaptic connections or only a loss thereof, chronological organization models can be generalized to include models where the different groups of content neurons are not just copies of each other, but the representations of the memory content in each group differ from each other. For example, a lossy, dynamically organized model could store an abstract representation of an event together with a detailed representation and forget the detailed representation faster than the abstract representation (Figure 5C; c.f. trace transformation theory [7]).

2.4.4 Associative Chaining

Chaining models differ from tagging, strength and chronological organization models in that time information is not inferable from a single recall step. Instead, recall triggers a replay of sequences of past experiences. For example, a simple chaining model could grow *auto-completion* connections between the neurons activated by the current stimulus (dotted in Figure 3D) and *forward* connections between neurons activated by the previous stimulus and the current one [32]. In the recall step, the current stimulus would activate the pattern of the recalled event using the auto-completion connections and then replay subsequent experiences using the forward connections, until the replay sequences ends. The time since storage of the recalled event could be estimated by the length of the replay sequence. Although a neural circuit with ramping activity could perform such an estimation, this only works if the replay sequence actually ends and does not enter a limit cycle, which would be caused by experiencing the same event at different moments in time, for example.

Instead of chaining indifferently all events together, a selective chaining mechanism could chain related events only. For example, if a certain stimulus B triggers the associative recall of a past event A, synaptic connections could grow from the neurons activated during recall of A to the neurons encoding stimulus B. This associative chain of A and B establishes an *order code* that stores the information "event A happened before B". If in the future another stimulus C triggers the recall of stimulus A, stimulus B would be retrieved and the time since storage of A could be roughly estimated by the length of the associative chain of related events.

2.5 Neural Implementations

We model brains that observe sensory states $x_t, x_{t+1}, \ldots$ and take actions (or decisions) $a_t, a_{t+1}, \ldots$ on a slow timescale. These brains have internal neural states $z_{t,\tau}$ and synaptic connection parameters $w_{t,\tau}$ that evolve on a faster timescale, indicated by the second time index τ. Our goal is to find functions f and g that govern the fast dynamics $(z_{t,\tau}, w_{t,\tau}) = f(z_{t,\tau-1}, w_{t,\tau-1}, x_t)$ and slow decision making dynamics $a_t = g(z_{t,T_t}, w_{t,T_t})$, where T_t is the number of fast neural processing steps at time t. The fast dynamics governs also the transition from t to $t+1$, i.e. $(z_{t+1,1}, w_{t+1,1}) = f(z_{t,T_t}, w_{t,T_t}, x_t)$. These dynamics should allow to solve tasks like learning to take actions that depend on the age of memories.

2.5.1 Storage and Retrieval of Information with Hebbian Plasticity Rules and Modulatory Factors

Most models require separate storage and retrieval phases. Suppose "red triangle" has already been stored at time step t_1, but the input"triangle" should be stored as happening at time t_3 in Figure 3. Then, the tag neuron t_1 and the content neurons "triangle" and "red" should be active in the retrieval phase of the remembered event "red triangle" (Figure 4D), and the tag neuron t_3 and the content neuron "triangle" should be active in the storage phase, if storage is implemented with some sort of Hebbian synaptic plasticity (Figure 4C). Separate storage and retrieval phases could be implemented by a periodic process in which external input drives the memory network during the storage phase and recurrent connectivity in the memory network dominates during the retrieval phase. The theta rhythm in the hippocampus could drive such a periodic process [33]. Alternatively, the separation into storage and retrieval phases could be implemented with synaptic delays (Figure 6) or some modulatory factors, like neurotransmitters or neuropeptides. During the storage phase, Hebbian plasticity, potentially modulated by a third factor [34, 35, 36, 37, 38], leads to the formation of new memories.

2.5.2 Synaptic Dynamics

Implementations of the examples discussed in the previous section require synaptic changes at different moments in time. In all examples, Hebbian synaptic plasticity during the storage phase would be sufficient to generate a trace of the sensory experiences in the memory system. Behavioural rules that depend on the content and the age of recalled memories could be learned with neoHebbian synaptic plasticity [34, 35, 36, 37, 38], where jointly active neurons generate an eligibility trace that is modulated by a subsequent signal. This modulating signal could communicate the reward received after a successful action.

For dynamic representations of time, synaptic changes occur also to reflect the passage of time. In models with dynamic rate coding of time, the strength or the number of certain synapses changes with time (c.f. Figure 5). In models with dynamic one-hot or distributed coding of time, parts of the memory networks are rewired by losing some synapses and growing new ones with the passage of time (c.f. Figure 4A2 and subsubsection 2.4.3).

For static representations of time, the synaptic changes for memorization and learning are sufficient. Although this is an appealing advantage of static representations of time, it comes at the cost of an increased complexity to compute the age of recalled memories, because a representation of the current moment in time needs to be compared with the time of storage of the recalled event (c.f. Figure 4B1).

3 Discussion

We showed that different coding and association schemes of content and time information leads to a family of models where the time of past events can be retrieved through an automatic process. The models in this family can be classified into time tagging models, where the time information is concatenated with the content information, strength models, where the time information is encoded in the activity level of recalled events, or chronological organization models, where time information can be retrieved from the identity of the content neurons active during recall. For each class of models there exist static variants, where the activity during recall of a given event is independent of when recall happens, and dynamic variants, where the representation of memory contents changes with the passage of time. We also discussed associative chaining models, where the time information can be retrieved from the temporal pattern of recalled activity.

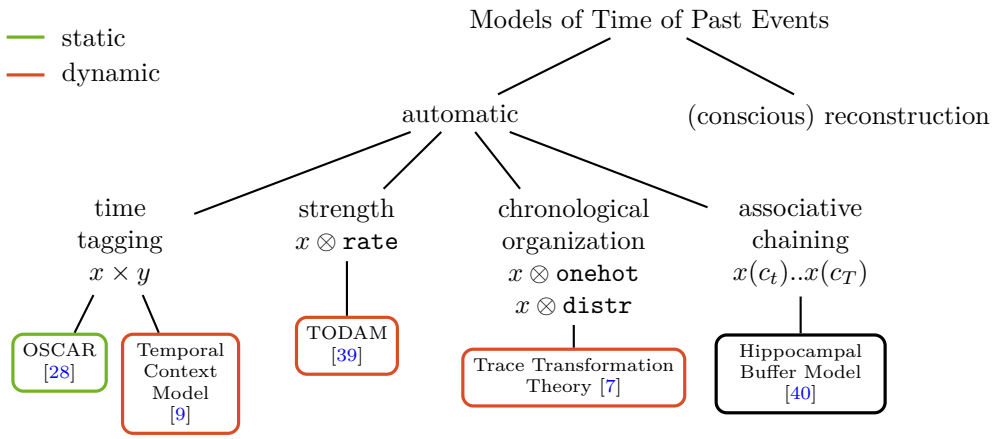

Figure 8: **A Categorization of Models of Time of Past Events with Examples of Published Models**.

Memory storage and learning of new tasks in the proposed models relies on Hebbian and neoHebbian synaptic plasticity, for which there is ample experimental evidence [34, 35, 36, 37, 38]. An implementation of dynamic rate representations of time, could rely on synapses that decay at different rates on a time scale of days to weeks [41, 42]. A rewiring of networks in dynamic one-hot or distributed encoding of time is consistent with the observed phenomena of rewiring of connections [43], representational drift [44] and systems memory consolidation [45, 7, 15]. Although there is experimental evidence for all the synaptic processes needed to implement the models here, further experiments should be done to determine which process is actually to remember the time of past events.

Many well-known theories of episodic memory treat time information with one of the automatic processes discussed here (Figure 8). The wealth of experiments on list learning following Ebbinghaus' seminal work [46], led to the development of many computational models that attempt to replicate human behavior in these experiments, with different kinds of strength models [47, 48, 49], (note that e.g. TODAM2 [49] uses convolutions to associate contents, but a simple strength decay model to associate content with explicit time information) or time tagging models with oscillators [28] or contextual tags [9, 10, 12]. Theories of systems memory consolidation focus on different forms of amnesia and the associated role of the hippocampus [45, 7, 15]. In one form or another, all systems consolidation theories posit that memories for specific events are represented in hippocampus and neocortex and that this representation can change over time; thus they are chronological organization theories. There exists also models of episodic memory were time information is not stored and the time since storage could not be retrieved [13, 14, 50].

Alternative taxonomies of the processing of time information have been proposed, for example, by [4]. Friedman [4] described on a conceptual level eight theories: strength, chronological organization, time tagging, contextual overlap, encoding perturbation, associative chaining, reconstruction and order codes. We agree with this categorization, but see time tagging, contextual overlap and encoding perturbation as specific versions of time tagging models. Additionally, our work provides concrete hypotheses for neural implementations of all these models and discusses their implications on readout of temporal and content information.

In our systematic investigation of models of time of past events we discussed exclusively automatic processes (Figure 8), but (consious) reconstruction-based processes seem to by highly relevant for adult humans [4]. Such reconstructions may depend on order codes that are more complex than the ones discussed in the chaining section of this article [4, 51, 5].

Theory-ladenness of observations together with the richness of biological phenomena allows to find

support in experimental data for different theories, like chronological organization as proposed by theories of systems consolidation [45, 7], time tagging models with context tags that decay on different time scales [52, 53], time tagging models based on intrinsic oscillators [54], time tagging models with hippocampal CA1 time cells [55] and hippocampus-dependent reconstruction-based theories [56], and it is reasonable to assume that different mechanisms are at play in different species and even within the same individual for tasks on different time scales.

The different versions of time tagging, strength, chronological organization and chaining models discussed here can serve as concrete hypotheses for future experimental investigations of the neuronal mechanisms that allow recall of the time of past events and learning rules based on the age of memories. Importantly, behavioral experiments will not be sufficient to discriminate all the models presented in this work. For example, behavioral experiments investigating the what-where-when memory of animals could be reproduced with all the models presented here and only the generalization of learned rules based on the age of memories might give some hints about the underlying mechanism, because some rules require complex readout and are more difficult to learn with some coding strategies than with others.

References

[1] Buhusi, C. V. & Meck, W. H. What makes us tick? functional and neural mechanisms of interval timing. *Nature Reviews Neuroscience* **6**, 755–765 (2005). URL http://dx.doi.org/10.1038/nrn1764.

[2] Issa, J. B., Tocker, G., Hasselmo, M. E., Heys, J. G. & Dombeck, D. A. Navigating through time: A spatial navigation perspective on how the brain may encode time. *Annual Review of Neuroscience* **43**, 73–93 (2020). URL http://dx.doi.org/10.1146/annurev-neuro-101419-011117.

[3] Paton, J. J. & Buonomano, D. V. The neural basis of timing: Distributed mechanisms for diverse functions. *Neuron* **98**, 687–705 (2018). URL http://dx.doi.org/10.1016/j.neuron.2018.03.045.

[4] Friedman, W. J. Memory for the time of past events. *Psychological Bulletin* **113**, 44–66 (1993). URL http://dx.doi.org/10.1037/0033-2909.113.1.44.

[5] Friedman, W. J. The development of memory for the times of past events. *The Wiley Handbook on the Development of Children's Memory* 394–407 (2013). URL http://dx.doi.org/10.1002/9781118597705.ch17.

[6] Jelbert, S. A. & Clayton, N. S. Comparing the non-linguistic hallmarks of episodic memory systems in corvids and children. *Current Opinion in Behavioral Sciences* **17**, 99–106 (2017). URL http://dx.doi.org/10.1016/j.cobeha.2017.07.011.

[7] Moscovitch, M. & Gilboa, A. Systems consolidation, transformation and reorganization: Multiple trace theory, trace transformation theory and their competitors. *PsyArXiv Preprints* (2021). URL http://dx.doi.org/10.31234/osf.io/yxbrs.

[8] Norman, K. A., Detre, G. & Polyn, S. M. Computational models of episodic memory. *The Cambridge Handbook of Computational Psychology* 189–225 (2008). URL http://dx.doi.org/10.1017/cbo9780511816772.011.

[9] Howard, M. W. & Kahana, M. J. A distributed representation of temporal context. *Journal of Mathematical Psychology* **46**, 269–299 (2002). URL http://dx.doi.org/10.1006/jmps.2001.1388.

[10] Polyn, S. M., Norman, K. A. & Kahana, M. J. A context maintenance and retrieval model of organizational processes in free recall. *Psychological Review* **116**, 129–156 (2009). URL http://dx.doi.org/10.1037/a0014420.

[11] Kahana, M. J. *Foundations of Human Memory* (Oxford University Press, 2012).

[12] Howard, M. W. Formal models of memory based on temporally-varying representations. *arXiv e-prints* arXiv:2201.01796 (2022). 2201.01796.

[13] Kesner, R. P. & Rolls, E. T. A computational theory of hippocampal function, and tests of the theory: New developments. *Neuroscience & Biobehavioral Reviews* **48**, 92–147 (2015). URL http://dx.doi.org/10.1016/j.neubiorev.2014.11.009.

[14] Norman, K. A. & O'Reilly, R. C. Modeling hippocampal and neocortical contributions to recognition memory: A complementary-learning-systems approach. *Psychological Review* **110**, 611–646 (2003). URL http://dx.doi.org/10.1037/0033-295x.110.4.611.

[15] Remme, M. W. H. *et al.* Hebbian plasticity in parallel synaptic pathways: A circuit mechanism for systems memory consolidation. *PLOS Computational Biology* **17**, e1009681 (2021). URL http://dx.doi.org/10.1371/journal.pcbi.1009681.

[16] Henson, R. N. Short-term memory for serial order: The start-end model. *Cognitive Psychology* **36**, 73–137 (1998). URL http://dx.doi.org/10.1006/cogp.1998.0685.

[17] Clayton, N. S. & Dickinson, A. Episodic-like memory during cache recovery by scrub jays. *Nature* **395**, 272–274 (1998). URL http://dx.doi.org/10.1038/26216.

[18] de Kort, S. R., Dickinson, A. & Clayton, N. S. Retrospective cognition by food-caching western scrub-jays. *Learning and Motivation* **36**, 159–176 (2005). URL http://dx.doi.org/10.1016/j.lmot.2005.02.008.

[19] Rigotti, M. *et al.* The importance of mixed selectivity in complex cognitive tasks. *Nature* **497**, 585–590 (2013). URL http://dx.doi.org/10.1038/nature12160.

[20] Brette, R. Is coding a relevant metaphor for the brain? *Behavioral and Brain Sciences* **42** (2018). URL http://dx.doi.org/10.1017/S0140525X19000049.

[21] Hopfield, J. J. Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci. USA* **79**, 2554–2558 (1982).

[22] Willshaw, D. J., Buneman, O. P. & Longuet-Higgins, H. C. Non-holographic associative memory. *Nature* **222**, 960–962 (1969). URL http://dx.doi.org/10.1038/222960a0.

[23] Smolensky, P. Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial Intelligence* **46**, 159–216 (1990). URL http://dx.doi.org/10.1016/0004-3702(90)90007-m.

[24] Kelly, M. A., Blostein, D. & Mewhort, D. J. K. Encoding structure in holographic reduced representations. *Canadian Journal of Experimental Psychology / Revue canadienne de psychologie expérimentale* **67**, 79–93 (2013). URL http://dx.doi.org/10.1037/a0030301.

[25] Illing, B., Gerstner, W. & Brea, J. Biologically plausible deep learning — but how far can we go with shallow networks? *Neural Networks* **118**, 90–101 (2019). URL http://dx.doi.org/10.1016/j.neunet.2019.06.001.

[26] Abeles, M. *Local Cortical Circuits: An Electrophysiological Study.* Studies of Brain Function (Springer, Berlin, Germany, 1982).

[27] Trengove, C., van Leeuwen, C. & Diesmann, M. High-capacity embedding of synfire chains in a cortical network model. *Journal of Computational Neuroscience* **34**, 185–209 (2012). URL http://dx.doi.org/10.1007/s10827-012-0413-9.

[28] Brown, G. D. A., Preece, T. & Hulme, C. Oscillator-based memory for serial order. *Psychological Review* **107**, 127–181 (2000). URL http://dx.doi.org/10.1037/0033-295x.107.1.127.

[29] Muscinelli, S. P., Gerstner, W. & Brea, J. Exponentially long orbits in hopfield neural networks. *Neural Computation* **29**, 458–484 (2017). URL http://dx.doi.org/10.1162/NECO_a_00919.

[30] Shankar, K. H. & Howard, M. W. A scale-invariant internal representation of time. *Neural Computation* **24**, 134–193 (2012). URL http://dx.doi.org/10.1162/NECO_a_00212.

[31] Bochud, T. & Challet, D. Optimal approximations of power-laws with exponentials. *arXiv e-prints* (2006). URL https://arxiv.org/abs/physics/0605149.

[32] Sompolinsky, H. & Kanter, I. Temporal association in asymmetric neural networks. *Physical Review Letters* **57**, 2861–2864 (1986). URL http://dx.doi.org/10.1103/physrevlett.57.2861.

[33] Hasselmo, M. E., Bodelón, C. & Wyble, B. P. A proposed function for hippocampal theta rhythm: Separate phases of encoding and retrieval enhance reversal of prior learning. *Neural Computation* **14**, 793–817 (2002). URL http://dx.doi.org/10.1162/089976602317318965.

[34] Lisman, J., Grace, A. A. & Duzel, E. A neohebbian framework for episodic memory; role of dopamine-dependent late ltp. *Trends in Neurosciences* **34**, 536–547 (2011). URL http://dx.doi.org/10.1016/j.tins.2011.07.006.

[35] Gerstner, W., Lehmann, M., Liakoni, V., Corneil, D. & Brea, J. Eligibility traces and plasticity on behavioral time scales: Experimental support of neohebbian three-factor learning rules. *Frontiers in Neural Circuits* **12**, 53 (2018). URL http://dx.doi.org/10.3389/fncir.2018.00053.

[36] Roelfsema, P. R. & Holtmaat, A. Control of synaptic plasticity in deep cortical networks. *Nature Reviews Neuroscience* **19**, 166–180 (2018). URL http://dx.doi.org/10.1038/nrn.2018.6.

[37] Kuśmierz, L., Isomura, T. & Toyoizumi, T. Learning with three factors: modulating hebbian plasticity with errors. *Current Opinion in Neurobiology* **46**, 170–177 (2017). URL http://dx.doi.org/10.1016/j.conb.2017.08.020.

[38] Magee, J. C. & Grienberger, C. Synaptic plasticity forms and functions. *Annual Review of Neuroscience* **43**, 95–117 (2020). URL http://dx.doi.org/10.1146/annurev-neuro-090919-022842.

[39] Murdock, B. B. Context and mediators in a theory of distributed associative memory (todam2). *Psychological Review* **104**, 839–862 (1997). URL http://dx.doi.org/10.1037/0033-295X.104.4.839.

[40] Jensen, O. & Lisman, J. E. Hippocampal sequence-encoding driven by a cortical multi-item working memory buffer. *Trends in Neurosciences* **28**, 67–72 (2005). URL http://dx.doi.org/10.1016/j.tins.2004.12.001.

[41] Abraham, W. C. How long will long-term potentiation last? *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences* **358**, 735–744 (2003). URL http://dx.doi.org/10.1098/rstb.2002.1222.

[42] Statman, A., Kaufman, M., Minerbi, A., Ziv, N. E. & Brenner, N. Synaptic size dynamics as an effectively stochastic process. *PLoS Computational Biology* **10**, e1003846 (2014). URL http://dx.doi.org/10.1371/journal.pcbi.1003846.

[43] Bennett, S. H., Kirby, A. J. & Finnerty, G. T. Rewiring the connectome: Evidence and effects. *Neuroscience & Biobehavioral Reviews* **88**, 51–62 (2018). URL http://dx.doi.org/10.1016/j.neubiorev.2018.03.001.

[44] Rule, M. E., O'Leary, T. & Harvey, C. D. Causes and consequences of representational drift. *Current Opinion in Neurobiology* **58**, 141–147 (2019). URL http://dx.doi.org/10.1016/j.conb.2019.08.005.

[45] Squire, L. R., Genzel, L., Wixted, J. T. & Morris, R. G. Memory consolidation. *Cold Spring Harbor Perspectives in Biology* **7**, a021766 (2015). URL http://dx.doi.org/10.1101/cshperspect.a021766.

[46] Ebbinghaus, H. *Über das Gedächtnis: Untersuchungen zur experimentellen Psychologie* (Duncker & Humblot, 1885). URL https://www.deutschestextarchiv.de/book/show/ebbinghaus_gedaechtnis_1885.

[47] Wickelgren, W. A. & Norman, D. A. Strength models and serial position in short-term recognition memory. *Journal of Mathematical Psychology* **3**, 316–347 (1966). URL http://dx.doi.org/10.1016/0022-2496(66)90018-6.

[48] Lewandowsky, S. & Murdock, B. B. Memory for serial order. *Psychological Review* **96**, 25–57 (1989). URL http://dx.doi.org/10.1037/0033-295X.96.1.25.

[49] Murdock, B., Smith, D. & Bai, J. Judgments of frequency and recency in a distributed memory model. *Journal of Mathematical Psychology* **45**, 564–602 (2001). URL http://dx.doi.org/10.1006/jmps.2000.1339.

[50] Romani, S., Pinkoviezky, I., Rubin, A. & Tsodyks, M. Scaling laws of associative memory retrieval. *Neural Computation* **25**, 2523–2544 (2013). URL http://dx.doi.org/10.1162/NECO_a_00499.

[51] Friedman, W. J. & Janssen, S. M. J. Do people remember the temporal proximity of unrelated events? *Memory & Cognition* **38**, 1122–1136 (2010). URL http://dx.doi.org/10.3758/MC.38.8.1122.

[52] Tsao, A. *et al.* Integrating time from experience in the lateral entorhinal cortex. *Nature* **561**, 57–62 (2018). URL http://dx.doi.org/10.1038/s41586-018-0459-6.

[53] Bright, I. M. *et al.* A temporal record of the past with a spectrum of time constants in the monkey entorhinal cortex. *Proceedings of the National Academy of Sciences* **117**, 20274–20283 (2020). URL http://dx.doi.org/10.1073/pnas.1917197117.

[54] Rolls, E. T. & Mills, P. The generation of time in the hippocampal memory system. *Cell Reports* **28**, 1649–1658.e6 (2019). URL http://dx.doi.org/10.1016/j.celrep.2019.07.042.

[55] Eichenbaum, H. On the integration of space, time, and memory. *Neuron* **95**, 1007–1018 (2017). URL http://dx.doi.org/10.1016/j.neuron.2017.06.036.

[56] Bellmund, J. L. S., Deuker, L., Montijn, N. D. & Doeller, C. F. Mnemonic construction and representation of temporal structure in the hippocampal formation. *Nature Communications* **13** (2022). URL http://dx.doi.org/10.1038/s41467-022-30984-3.