Journal of Mathematical Psychology 97 (2020) 102358

Contents lists available at ScienceDirect

Journal of Mathematical Psychology

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

Adding a bias to vector models of association memory provides item memory for free☆

Jeremy B. Caplan *, Kaiyuan Xu, Sucheta Chakravarty, Kelvin E. Jones

Department of Psychology, Neuroscience and Mental Health Institute, University of Alberta, Edmonton, AB, T6G 2E9, Canada

ARTICLE INFO

Article history:
Received 5 December 2019
Received in revised form 25 March 2020
Accepted 29 March 2020
Available online xxxx

ABSTRACT

Anderson (1970) introduced two models that are at the core of artificial neural network models as well as cognitive mathematical models of memory. The first, a simple summation of items, represented as vectors, can support rudimentary item-recognition. The second, a heteroassociative model consisting of a summation of outer products between paired item vectors, can support cued recall of associations. Anderson recommended fixing the element-value mean to zero, for tractability, and with minimal loss of generality. However, in a neural network model, if element values are represented by firing rates, this mean-centering is violated, because firing rates cannot be negative. We show, analytically, that adding a bias to item representations produces interference from other studied list items. Although this worsens cued recall, it also tempts the model to make intrusion responses to other studied items, not unlike human participants. Moreover, an unexpected feature appears: when probed with a constant vector, containing no "information," the model retrieves a weighted sum of studied items, formally equivalent to Anderson's item-memory model. This speaks to Hockley and Cristi's (1996) findings that associative study strategies led to high item-recognition, but not vice versa. We show that such a model can achieve high levels of performance (d′), when the bias is greater than zero but not too large relative to the standard deviation of element values. We demonstrate these effects in a two-layer spiking-neuron network model. Thus, when modelers have striven for realism and relaxed mean-centering, such models may not only still function at adequate levels, but acquire a spin-off functionality that can actually be used, without the need for additional encoding terms specific to item-memory.

1. Introduction

Anderson (1970) introduced two models that remain at the core of artificial neural network models and cognitive mathematical models of memory to the present day (e.g., Howard & Kahana, 1999, 2002; Osth & Dennis, 2015; as well as their mathematical cousins, convolution-based models, e.g., Eliasmith et al., 2012; Franklin & Mewhort, 2015). These models start with the – now standard – assumption that items (such as words) can be represented as vectors, where the elements of the vector are thought of as feature strengths. First, the so-called "matched filter" model is a simple weighted sum of vectors corresponding to list items. This model, which we call the Item Model, can simulate rudimentary item-memory tasks, including item-recognition, which Anderson (1973) developed further. In episodic item recognition, the participant (or model) is asked to respond "old" to items that were on the target list (target probes) and "new" to others (lure or foil probes). The second, so-called "linear associator" model, which we call the Association Model, is a heteroassociative model comprised of a weighted sum of outer products between paired item vectors (later generalized to the tensor model by Humphreys, Bain, & Pike, 1989). This model can support cued recall of associations (having studied the pair AB: given A as a probe, respond with B) and associative recognition (having studied pairs AB and CD, respond "intact" to probes AB and CD; respond "rearranged" or "recombined" to probes AD and CB), earlier known as pair recognition.

Anderson (1970) recommended fixing the mean element value to zero, for analytic simplicity, and with minimal loss of generality, although he pointed out that the optimal signal-to-noise ratio in his models was at a slightly positive mean element value. In symbolic implementations of the matrix model, it is standard

☆ We thank Ulises Rodríguez Domínguez for helpful feedback on the manuscript. Funded in part by the Natural Sciences and Engineering Research Council of Canada. Model code can be obtained from https://osf.io/h8s6p/?view_only=1a88d5c92a504d8099792c4366cc68e7.

* Corresponding author.
E-mail address: jcaplan@ualberta.ca (J.B. Caplan).

https://doi.org/10.1016/j.jmp.2020.102358
0022-2496/

practice to follow this advice. Even if feature values are drawn at random, as is often done, the expectation is zero. In practice, the mean will fluctuate around zero, but those fluctuations are smaller, on average, as the dimensionality, n, of the vectors increases, closely approximating mean-centering.

However, in a realistic neural network model, if element values are represented by firing rates, as is commonly done, mean-centering is violated, because firing rates cannot be negative. Raising this concern, Anderson (1970) suggested that this violation of mean-centering could be solved by incorporating a population of inhibitory neurons, or by implementing a bias, such that exclusively positive element values could be computationally treated as positive above the threshold, and negative below the threshold (but truncated at zero).

Here we consider what happens when one deviates from mean-centered representations. We will show that although a loss of optimality and algebraic simplicity may have disadvantages, relaxing the mean-centering condition comes with some advantages. Moreover, a non-mean-centered item representation is congruent with non-negative firing rates, while still, in some sense, remaining more parsimonious than an opponent-code, which would double the number of neurons required. Finally, other well supported models have been motivated by other considerations to construct item representations from non-negative feature values, such as geometric distributions of feature values in REM (Shiffrin & Steyvers, 1997) and binary vectors (e.g., Cox & Shiffrin, 2017; Tsodyks & Feigel'man, 1988). It may, therefore, be important to understand how a memory model performs if one has independent reasons to favor a non-negative, and thus non-mean-centered, vector representation.

We were inspired to reconsider the effects of leaving vector representations uncentered, due to an incidental finding in a spiking-neuron network implementation of the matrix model. We noticed that although our model, when given cued-recall probes, was retrieving the firing-rate pattern that best resembled the correct target item, the retrieved vector also *moderately* resembled target items from other studied pairs. We understood this geometrically, as depicted in Fig. 1. In two dimensions, consider two vectors that are orthogonal. The angle between them, $\theta = 90°$, makes their dot product zero. Adding a constant value to all vector dimensions is equivalent to shifting the origin. This causes previously orthogonal vectors to lengthen, but more importantly, their angle, θ', becomes smaller than 90°, and they are thus no longer orthogonal. Their dot product becomes non-zero, and this "similarity" between vectors has the effect that probes are less selective for the one best-matching encoded term. In effect, the model with non-zero bias retrieves not only the correct target item, but a weighted sum of all studied target items. In the limit, infinite bias forces all vectors to point in the same direction, becoming identical. As the bias increases, therefore, items become increasingly confusable with one another, suggesting the bias should not be arbitrarily large.

In what follows, first we analytically derive recognition-memory discriminability (d') for the Item Model and show what happens when a bias is added. We then turn to the Association Model and show, analytically, that adding a bias to item representations produces interference from other studied list items and makes the model worse at cued recall. This also tempts the model to make intrusion responses that are reminiscent of the kinds of errors that human participants make. Then we propose a simple way in which the Association Model, with a bias, can be used to perform the item-recognition task, even without items being explicitly encoded. Specifically: when probed with a constant vector, containing no "information," the model retrieves a weighted sum of studied items, formally equivalent to Anderson's Item Model. This echoes findings that associative study strategies lead to high item-recognition, but not vice versa (Hockley & Cristi, 1996). We show that such a model can achieve high levels of performance (d'), when the bias is greater than zero but not too large relative to the standard deviation of element values. Finally, although analytic derivations are informative and speak to the generality of the results, the story can change when one builds an actual model implementation that can be simulated, and additional practical considerations need to be addressed. Indeed, this was our initial inspiration. We therefore present a simulation that demonstrates the presence of this kind of intrinsic item-memory in a two-layer artificial neural network using realistic spiking neurons.

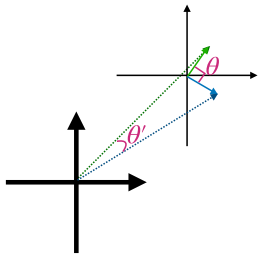

Fig. 1. Schematic depiction of the effect of features values having a non-zero mean. Two vectors, depicted in blue and green solid lines, respectively, start out orthogonal relative the origin (thin axis lines). Thus, the angle between them, $\theta = 90°$. When a constant value is added to all cells of the vectors (in this simple example, only two dimensions each), the geometric interpretation is that the origin is shifted (thick axis lines). The new vectors (darker, dotted lines) now have a much smaller angle, θ', between them. Because cued recall starts with, mathematically, a dot product between the probe item and the memory matrix, the addition of a bias has the effect of cueing, to some degree (depending on the magnitude of the bias itself), all other associations.

2. The item model

Let items be represented by column vectors of length n, denoted in boldface, lowercase letters, such as $\mathbf{f}$. We follow the standard assumption that the cells of each item vector are composed of independent, identically distributed values; thus, $\mathbf{f}(k) \sim N(\mu, \sigma^2)$. Anderson (1970) used this assumption throughout his derivations, but then showed that little is lost if one simplifies by assuming that $\mu = 0$; in other words, that vectors are mean-centered. Interestingly, he showed that the signal-to-noise ratio of the model was optimal at a point where $\mu > 0$, but still quite close to zero. We will, instead, maintain $\mu \neq 0$ throughout, and examine the dependence of the results on the value of μ. Anderson (1970) proposed that memory for a list of items could be stored as the vector sum of the corresponding item vectors, where the weights, α_i, stand in for variable encoding strengths, which we assume are drawn from a Gaussian distribution with mean μ_α and variance σ_α^2, written $\alpha_i \sim N(\mu_\alpha, \sigma_\alpha^2)$:

$$\mathbf{m} = \sum_{i=1}^{L} \alpha_i \mathbf{f}_i, \tag{1}$$

where i indexes items and L is the list length. Item recognition can be conducted by computing the dot product (also called the inner product) between a probe item, $\mathbf{f}_x$, and memory,

$$s = \mathbf{m}^\top \mathbf{f}_x. \tag{2}$$

We set aside encoding variability to simplify the derivations (i.e., setting $\alpha_i = 1, \ \forall i$). The dot product of an item with itself, which Anderson (1970) termed its "Power",

$$\mathbf{f}_i^\top \mathbf{f}_i = \sum_{k=1}^{n} f_i(k)^2 \tag{3}$$

has mean:

$$M_{ii} = \mathrm{E}\left[\mathbf{f}_i^{\mathrm{T}}\mathbf{f}_i\right] = \sum_{k=1}^{n}(\mu + X_k)(\mu + X_k)$$
$$= n\left(\mu^2 + 2\mu\mathrm{E}[X] + \mathrm{E}[X^2]\right) = n\left(\mu^2 + \sigma^2\right), \quad (4)$$

where X_k (and likewise, Y_k and Z_k, used below) denotes a single draw from a mean-centered distribution $\sim N(0, \sigma)$, E[] denotes the expectation and X (and below, Y and Z) denotes a standard normal (Gaussian-distributed) random variable with a mean of zero and variance σ^2. We have used the fact that the expectation of odd powers of X is zero, due to odd symmetry (e.g., Anderson, 1970; Weber, 1988), and solutions from Weber (1988) including: $\mathrm{E}[X^2] = \sigma^2$, $\mathrm{E}[X^4] = 3\sigma^4$ (also used in later derivations below). Substituting the common assumption, $\sigma^2 = 1/n$, which produces vectors that are approximately normalized (unit length), this simplifies to:

$$M_{ii} = n\mu^2 + 1. \quad (5)$$

Note that if $\mu = 0$, the mean simplifies to 1. Its variance is:

$$V_{ii} = \mathrm{Var}\left[\mathbf{f}_i^{\mathrm{T}}\mathbf{f}_i\right] = \mathrm{E}\left[\left(\mathbf{f}_i^{\mathrm{T}}\mathbf{f}_i\right)^2\right] - \mathrm{E}\left[\mathbf{f}_i^{\mathrm{T}}\mathbf{f}_i\right]^2$$
$$= \mathrm{E}\left[\left(\sum_{k=1}^{n}\mathbf{f}_i(k)^2\right)\left(\sum_{l=1}^{n}\mathbf{f}_i(l)^2\right)\right] - M_{ii}^2$$
$$= n\mathrm{E}\left[(X+\mu)^4\right] + (n^2-n)\mathrm{E}\left[(X+\mu)^2(Y+\mu)^2\right] - M_{ii}^2$$
$$= n\mathrm{E}\left[X^4 + 6\mu^2X^2 + \mu^4\right] + (n^2-n)$$
$$\times\,\mathrm{E}\left[X^2Y^2 + \mu^2X^2 + \mu^2Y^2 + \mu^4\right] - M_{ii}^2$$
$$= n\left(3\sigma^4 + 6\mu^2\sigma^2 + \mu^4\right) + (n^2-n)\left(\sigma^4 + 2\mu^2\sigma^2 + \mu^4\right)$$
$$-n^2\left(\mu^4 + 2\mu^2\sigma^2 + \sigma^4\right)$$
$$= 2n\sigma^4 + 4n\sigma^2\mu^2,$$

where $\mathrm{Var}[] = \mathrm{E}\left[()^2\right] - \mathrm{E}\left[()\right]^2$ denotes variance. Substituting $\sigma^2 = 1/n$, this simplifies to:

$$V_{ii} = 2/n + 4\mu^2, \quad (6)$$

reducing to $2/n$ if $\mu = 0$, decreasing with n toward an asymptote of 0. This variance term also increases with increasing μ, but independent of the effect of n. The dot product between different items ($j \neq i$) has mean:

$$M_{ij} = \mathrm{E}\left[\mathbf{f}_i^{\mathrm{T}}\mathbf{f}_j\right] = \sum_{k=1}^{n}\mathbf{f}_i(k)\mathbf{f}_j(k) = \sum_{k=1}^{n}(\mu + X_k)(\mu + Y_k)$$
$$= n\left(\mu^2 + \mu\mathrm{E}[X] + \mu\mathrm{E}[Y] + \mathrm{E}[X]\mathrm{E}[Y]\right) = n\mu^2, \quad (7)$$

Note that for $\mu = 0$, $M_{ij} = 0$. Its variance is:

$$V_{ij} = \mathrm{Var}\left[\mathbf{f}_i^{\mathrm{T}}\mathbf{f}_j\right] = \mathrm{E}\left[\left(\mathbf{f}_i^{\mathrm{T}}\mathbf{f}_j\right)^2\right] - \mathrm{E}\left[\mathbf{f}_i^{\mathrm{T}}\mathbf{f}_j\right]^2$$
$$= \mathrm{E}\left[\left(\sum_{k=1}^{n}\mathbf{f}_i(k)\mathbf{f}_j(k)\right)\left(\sum_{l=1}^{n}\mathbf{f}_i(l)\mathbf{f}_j(l)\right)\right] - M_{ij}^2$$
$$= n\mathrm{E}\left[(X+\mu)^2(Y+\mu)^2\right] + (n^2-n)$$
$$\times\,\mathrm{E}\left[(W+\mu)(X+\mu)(Y+\mu)(Z+\mu)\right] - M_{ij}^2$$
$$\times\, n\sigma^4 + 2n\mu^2\sigma^2 + n\mu^4 + (n^2-n)\mu^4 - n^2\mu^4$$
$$= n\sigma^4 + 2n\mu^2\sigma^2.$$

Substituting $\sigma^2 = 1/n$, this simplifies to:

$$V_{ij} = 1/n + 2\mu^2, \quad (8)$$

reducing to $1/n$ when $\mu = 0$. For $\mu > 0$, this variance increases as the square of μ, again independent of n.

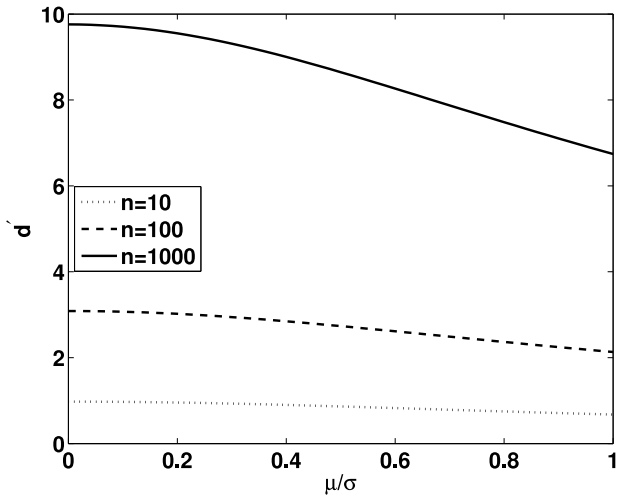

Fig. 2. Analytic solution for the Item Memory model: d' for list length 10, as a function μ (relative to σ), for three example values of vector dimensionality (n).

Departing from Anderson (1970), who derived the signal-to-noise ratio, we derive d' because it is the current standard measure of recognition-memory performance. We explicitly assume that lure items are generated in the same way as target items, but that lure items were simply not presented during the study phase. Thus:

$$d' = \frac{\mu_{\text{target}} - \mu_{\text{lure}}}{\frac{1}{\sqrt{2}}\sqrt{\sigma_{\text{target}}^2 + \sigma_{\text{lure}}^2}}. \quad (9)$$

For a target probe, there will be one term where $i = x$ and $L - 1$ terms with $i \neq x$. For a lure probe, there will be L terms with $i \neq j$. Because means and variances add:

$$\mu_{\text{target}} = M_{ii} + (L-1)M_{ij}$$
$$\mu_{\text{lure}} = LM_{ij}$$
$$\sigma_{\text{target}}^2 = V_{ii} + (L-1)V_{ij}$$
$$\sigma_{\text{lure}}^2 = LV_{ij}$$
$$\therefore d' = \frac{M_{ii} - M_{ij}}{(1/\sqrt{2})\sqrt{V_{ii} + (2L-1)V_{ij}}}. \quad (10)$$

Substituting Eqs. (5)–(8),

$$d' = \sqrt{\frac{2}{(2L+3)\mu^2 + (2L+1)/n}}. \quad (11)$$

Plotted for $L = 10$, at a few values of n (Fig. 2), one can see that if $\mu = 0$, this reduces to $\sqrt{\frac{2n}{(2L+1)}}$, which increases with increasing n (as $\sqrt{n}$) and decreases with L (i.e., a list-length effect, approximately $1/\sqrt{L}$ for large L). The effect of $\mu > 0$ is to increase the denominator independently of n, but proportionally to μ itself, and to the square root of L. Thus, introducing a bias reduces d', but in addition, introduces a further cost for large list lengths.

3. The association model

During the study phase of an association-memory task, associations between pairs of items are stored in a memory matrix, M, as a weighted sum of outer products between pairs of item vectors (Anderson, 1970), where the weights, $\alpha_i \sim N(\mu_\alpha, \sigma_\alpha^2)$,

stand in for variable encoding strengths, similar to the vector model of item recognition:

$$M = \sum_{i=1}^{L} \alpha_i \mathbf{g}_i \mathbf{f}_i^{\mathsf{T}} \tag{12}$$

where L is the list length (in the Association Model, this is the number of pairs per list), $\mathbf{f}_i$ is the left-hand item of pair i, and $\mathbf{g}_i$ is the corresponding right-hand item, constructed in exactly the same way as the $\mathbf{f}_i$ vectors. Probing with a given left-hand item from the list, $\mathbf{f}_x$ by multiplying from the right retrieves the vector $\mathbf{g}_r$:

$$\mathbf{g}_r = M\mathbf{f}_x = \sum_{i=1}^{L} \alpha_i \mathbf{g}_i (\mathbf{f}_i^{\mathsf{T}} \mathbf{f}_x) \tag{13}$$

If all $\mathbf{f}_i$ and $\mathbf{g}_i$ are mutually orthogonal, as is approximately the case for large n and small L (5–20 are typical values of L) and when $\mathbf{f}_i$ and $\mathbf{g}_i$ are mean-centered, retrieval is perfect, because $\mathrm{E}[\mathbf{f}_i^{\mathsf{T}} \mathbf{f}_x] = 1$ if $i = x$ and zero if $i \neq x$, thus yielding, on average, $\alpha_i \mathbf{g}_i$; in other words, the correct target item multiplied by a scalar encoding strength. When the orthogonality assumption is relaxed, as is the case here, when element values are drawn at random, $\mathbf{g}_r$ will still tend to include a $\mathbf{g}_x$ as its dominant term, but this will be contaminated by other items, $\alpha_i \mathbf{g}_i$, in proportion to $(\mathbf{f}_j^{\mathsf{T}} \mathbf{f}_x)$, the degree of similarity of the probe item to each corresponding left-hand item.

Now, to incorporate the bias by writing the corresponding vectors with bias included as $\tilde{\mathbf{f}}_i = \mathbf{f}_i + \mu\mathbf{I}$, where $\mathbf{I}$ is a vector of length n, containing the value 1 as every element (following Anderson, 1970), retrieval in the Association Model based on $\tilde{\mathbf{f}}_i$, Eq. (2), becomes:

$$\begin{aligned}\mathbf{g}_r &= \sum_{i=1}^{L} \alpha_i (\mathbf{g}_i + \mu\mathbf{I})(\mathbf{f}_i + \mu\mathbf{I})^{\mathsf{T}} (\mathbf{f}_x + \mu\mathbf{I}) \\ &= \sum_{i=1}^{L} \alpha_i \left(\mathbf{g}_i (\mathbf{f}_i^{\mathsf{T}} \mathbf{f}_x + \mu^2 n) + \mathbf{I} \left(\mu \mathbf{f}_i^{\mathsf{T}} \mathbf{f}_x + \mu^3 n \right) \right. \\ &\quad \left. + \mathbf{g}_i \left(\mu \mathbf{f}_i^{\mathsf{T}} \mathbf{I} + \mu \mathbf{I}^{\mathsf{T}} \mathbf{f}_x \right) + \mu \mathbf{I} \left(\mu \mathbf{f}_i^{\mathsf{T}} \mathbf{I} + \mu \mathbf{I}^{\mathsf{T}} \mathbf{f}_x \right) \right),\end{aligned} \tag{14}$$

The right-hand terms, colored gray, all have $\mathrm{E}[] = 0$ because $\mathrm{E}[\mathbf{f}_i \cdot \mathbf{I}] = 0$. Cued recall is accurate if the model selects $\mathbf{g}_x$ as the response and incorrect otherwise. For models lacking an explicit model of redintegration ("cleaning up" the retrieved vector to identify an acceptable response item from the lexicon consisting of the set of known items), the standard assumption is that the relative similarity (dot product) of the retrieved vector to response candidates determines accuracy. Thus, for the target, on average, and assuming large n:

$$\mathrm{E}[\mathbf{g}_r^{\mathsf{T}} (\mathbf{g}_x + \mu\mathbf{I})] = \alpha_x (1 + \mu^2 n) + \alpha_x n \mu^2 + L \mu_\alpha n^2 \mu^4, \tag{15}$$

where we have used $\mathrm{E}[\mathbf{f}_i^{\mathsf{T}} \mathbf{f}_x] = 1$ if $i = x$ and 0 otherwise. Thus, the match to the correct target item vector is simply equal to the tested pair's encoding strength when $\mu = 0$, but for $\mu > 0$, increases with n and L. Although the additional terms introduced by μ increase the matching strength to the target item, an additional typical assumption is that recall is inversely related to the matching strengths to non-targets. Thus, this is offset by increases in the match to incorrect target items (i.e., the other $\mathbf{g}_y$ vectors, $y \neq x$, of which there are $L - 1$):

$$\mathrm{E}[\mathbf{g}_r^{\mathsf{T}} (\mathbf{g}_y + \mu\mathbf{I})] = \alpha_y \mu^2 n + \alpha_x n \mu^2 + L \mu_\alpha n^2 \mu^4. \tag{16}$$

Again, if $\mu = 0$, this reduces to zero. For $\mu > 0$, the match to each of the $(L - 1)$ other studied targets increases with n and L. Remarkably, the incorrect target item strengths also increase partly in proportion to the strength of the *correct target* (middle term, proportional to α_x). Thus, one can clearly see how increasing μ swiftly reduces the discriminability of the correct target item from other studied target items. To compare, the match to other *probe* items (assuming an item cannot be both a left- and a right-hand item) is lower; all that survives is the term "probed" by the bias itself, $\mu\mathbf{I}$:

$$\mathrm{E}[\mathbf{g}_r^{\mathsf{T}} (\mathbf{f}_y + \mu\mathbf{I})] = \alpha_x n \mu^2 + L \mu_\alpha n^2 \mu^4, \tag{17}$$

and reduces to zero when $\mu = 0$. As the index y does not enter into the result, this is true for the match to the probe itself, $\mathbf{f}_x$, as well as all items that were not included in the study list whatsoever.

To estimate the probability of a correct response would require additional assumptions, such as whether retrieval is winner-take-all or probabilistic, and what constitutes the set of candidate response items, which is beyond the scope of the current article. However, the analytic solutions reveal some important insights. Comparing Eqs. (15)–(17), visualized for $L = 10$ and $n = 1000$ in Fig. 3b, one can see that on average, $\mathbf{g}_r$ will match best to the correct target, less well to target items from other studied pairs, and least to the probe items and any other non-studied items. This immediately leads to the prediction that intrusions in cued recall will be predominantly to incorrect target items, and only rarely to probe items from other studied pairs.

In the limit of large n, the strength to the correct target (Eq. (15)) and strength to each of the incorrect studied target items (Eq. (16)) converge, suggesting that the model would be correct on no more than $1/L$ cued recall tests. This could be counteracted, of course, by a large encoding strength for a particular pair, α_x, but this would not help performance over all

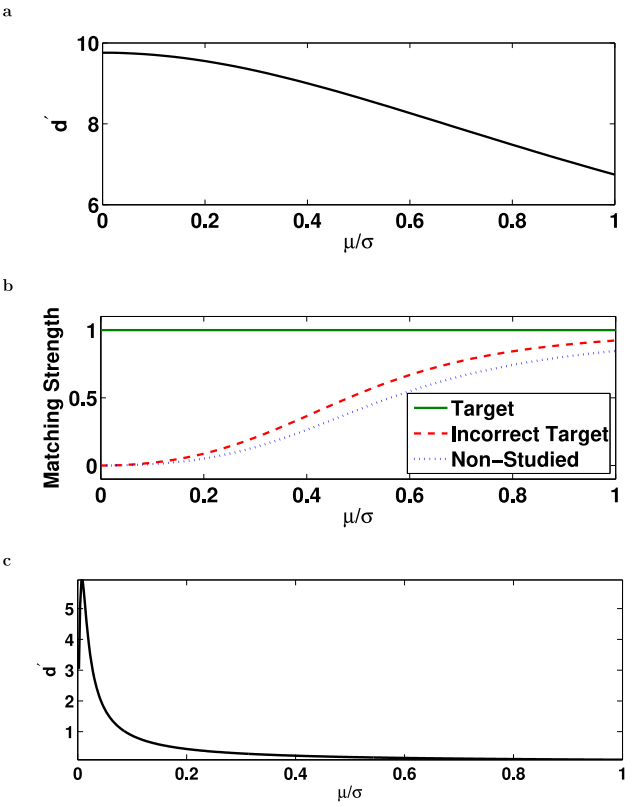

Fig. 3. Analytic solution for the Item Model, Eq. (11), with $L = 10$, $n = 1000$ (a); cued recall of the Association Model (b), where matching strength (dot product of retrieved vector, $\mathbf{g}_r$, with candidate response item vectors) is plotted relative to strength of the correct target (Eqs. (15)–(17)), and item memory retrieved from the Association Model (Eqs. (30)–(31)) (c). The horizontal axes (μ/σ) are aligned for comparison across panels.

pairs. As n becomes even larger, the rightmost term dominates and all strengths approach equality; thus, for non-zero μ, vector-dimensionality acts against probability of recall. As μ gets large, again the strength of the target approaches the strengths of studied incorrect target items. For very large μ, again, the right-most term dominates, introducing even more of a challenge from non-studied items. As L increases as well, the three strength equations quickly converge, as long as $\mu > 0$.

Thus, although we will show that a small bias may offer some benefit in the form of the availability of item-memory as a by-product of encoding of associations, the bias must not be very high, or else cued-recall performance will suffer quite drastically.

3.1. Item recognition by probing the association model with a constant vector

The Item Model (vector-sum), itself, can be retrieved by simply probing this model with a pure bias vector, $\mu\mathbf{I}$. This would correspond to exciting the input-layer neurons of a 2-layer artificial neural network by injecting the same amount of current to all input neurons (note that this will be an upper limit of performance, because a realistic model would have to assume additional noise on top of the $\mu\mathbf{I}$ vector, which would propagate through to retrieval):

$$\mathbf{m}_r = M(\mu\mathbf{I}) = \sum_{i=1}^{L} \alpha_i(\mathbf{g}_i + \mu\mathbf{I})(\mathbf{f}_i + \mu\mathbf{I})^{\intercal}\mu\mathbf{I} \tag{18}$$

$$= \sum_{i=1}^{L} \alpha_i \left((\mathbf{g}_i + \mu\mathbf{I})(\mu\mathbf{f}_i^{\intercal}\mathbf{I} + \mu^2\mathbf{I}^{\intercal}\mathbf{I})\right). \tag{19}$$

Recognition judgments can be made using this retrieved vector in the same manner as before, on the basis of the dot product of a probe vector, $\tilde{\mathbf{g}}_x$, with the memory, $\mathbf{m}_r$:

$$\tilde{\mathbf{g}}_x^{\intercal}\mathbf{m}_r = \sum_{i=1}^{L} \alpha_i \left((\mathbf{g}_x + \mu\mathbf{I})^{\intercal}(\mathbf{g}_i + \mu\mathbf{I})(\mu\mathbf{f}_i^{\intercal}\mathbf{I} + \mu^2\mathbf{I}^{\intercal}\mathbf{I})\right). \tag{20}$$

If we maintain the assumption that $\mathbf{I}$ is a perfect, noiseless constant vector, then we can substitute the scalar $\mathbf{I}^{\intercal}\mathbf{I} = n$. We can then rewrite Eq. (20):

$$\tilde{\mathbf{g}}_x^{\intercal}\mathbf{m}_r = T_1 + T_2$$

where

$$T_1 = n\mu^2 \sum_{i=1}^{L} \alpha_i(\mathbf{g}_x + \mu\mathbf{I})^{\intercal}(\mathbf{g}_i + \mu\mathbf{I})$$

$$T_2 = \sum_{i=1}^{L} \alpha_i \left((\mathbf{g}_x + \mu\mathbf{I})^{\intercal}(\mathbf{g}_i + \mu\mathbf{I})\mu\mathbf{f}_i^{\intercal}\mathbf{I}\right). \tag{21}$$

T_1 is the original item-recognition vector model multiplied by a constant factor, $n\mu^2$. T_2 is the same model multiplied by the dot product of $\mathbf{f}_i$ with $\mathbf{I}$, equivalent to the sum over the elements of $\mathbf{f}_i$. New expressions for μ'_{target} and μ'_{lure} are thus straight-forward to calculate, since $\mathrm{E}[T_2] = 0$. Setting all $\alpha_i = 1$ for simplicity:

$$\mu'_{\text{target}} = n\mu^2 + Ln^2\mu^4 \tag{22}$$

$$\mu'_{\text{lure}} = Ln^2\mu^4. \tag{23}$$

Next, the target-strength variance:

$$\sigma'^2_{\text{target}} = \mathrm{E}[T_1^2 + 2T_1T_2 + T_2^2] - \mu'^2_{\text{target}} \text{ where } x \text{ matches one } i$$

Solving these terms:

$$\mathrm{E}[T_1^2] = \left(L^2\, n^4\right)\, \mu^8 + \left(2\, L\, n^3 + n^2\, \left(L^2 + 3\, L\right)\right)\, \mu^6 + \left(n^2 + (L+1)\, n\right)\, \mu^4 \tag{24}$$

$$2\mathrm{E}[T_1T_2] = 0 \tag{25}$$

$$\mathrm{E}[T_2^2] = \left(L\, n^2\right)\, \mu^6 + (2\, n + 2\, L + 2)\, \mu^4 + \left(\frac{L+n+1}{n}\right)\, \mu^2 \tag{26}$$

For lures, many terms reduce to zero:

$$\sigma'^2_{\text{lure}} = n^2\mu^4\mathrm{E}[T_1^2 + 2T_1T_2 + T_2^2] - \mu'^2_{\text{lure}} \text{ where } x \text{ matches no } i \text{ or } j$$

Solving these terms :

$$\mathrm{E}[T_1^2] = \left(L^2\, n^4\right)\, \mu^8 + \left(n^2\, \left(L^2 + L\right)\right)\, \mu^6 + (L\, n)\, \mu^4 \tag{27}$$

$$2\mathrm{E}[T_1T_2] = 0 \tag{28}$$

$$\mathrm{E}[T_2^2] = \left(L\, n^2\right)\, \mu^6 + (2\, L)\, \mu^4 + \left(\frac{L}{n}\right)\, \mu^2. \tag{29}$$

Finally, solving for d' (Eq. (9), using Eqs. (22)–(29)), and illustrated in Fig. 3c,

$$d' = \mu\sqrt{2n/C}, \tag{30}$$

$$\text{where } C = \left(L^2n^3 + 3Ln^3\right)\mu^4 + \left((L+3/2)n^2 + (2L+1)n\right)\mu^2 + (L+n+1)/2. \tag{31}$$

This can be seen by inspecting distributions of matching strengths for targets and lures, produced by a numerical simulation used to verify the analytic derivations. Fig. 4 shows that when μ is too small (panel a) or too large (panel c), strength distributions for targets and lures overlap considerably, resulting in fairly low values of d'. For certain values of μ (panel b), the distribution can separate far more, resulting in quite high values of d'.

In the limit where $\mu = 0$, the model retrieves nothing. In other words, by probing with the bias vector, $\mu\mathbf{I}$, alone, the Association Model retrieves the vector sum of all target items, weighted by their corresponding pair-encoding strengths.

Note that the same α_i multiplies the matching strength for recognition of $\mathbf{g}_i$ and retrieval strength for cued-recall of corresponding pair, i. This leads to the further prediction that association-memory (tested with cued recall or associative recognition) should be highly correlated to recognition of the constituent items, where the correlation is computed across i, within each list.

This may also be part of the reason Hockley and Cristi (Hockley & Cristi, 1996) found that when participants studied a list in anticipation of associative recognition tests, they retained memory for the constituent items about as well as when studying for item-recognition, but not vice versa (see the General Discussion for more on this).

In sum, the greater the bias, μ, the less accurate both item recognition and cued recall become. At the same time, the greater μ, the more item-memory (the vector sum of all studied target items) can be retrieved by probing with a constant vector ($\mu\mathbf{I}$). This item memory "for free" inherits one property from the original item-memory with bias; namely, the more the bias, the worse item-recognition becomes. These tradeoffs suggest one could optimize μ to trade off cued recall accuracy for accuracy of memory for the accompanying target item.

4. Realistic spiking model

In this final section, we evaluate the effect of a bias in a spiking-neuron network model. We are particularly interested in whether incorrect target items are retrieved more than non-studied items in cued recall, and whether the model can perform at non-trivial levels at item-recognition when excited with constant input. Although the analytic models, derived above, are

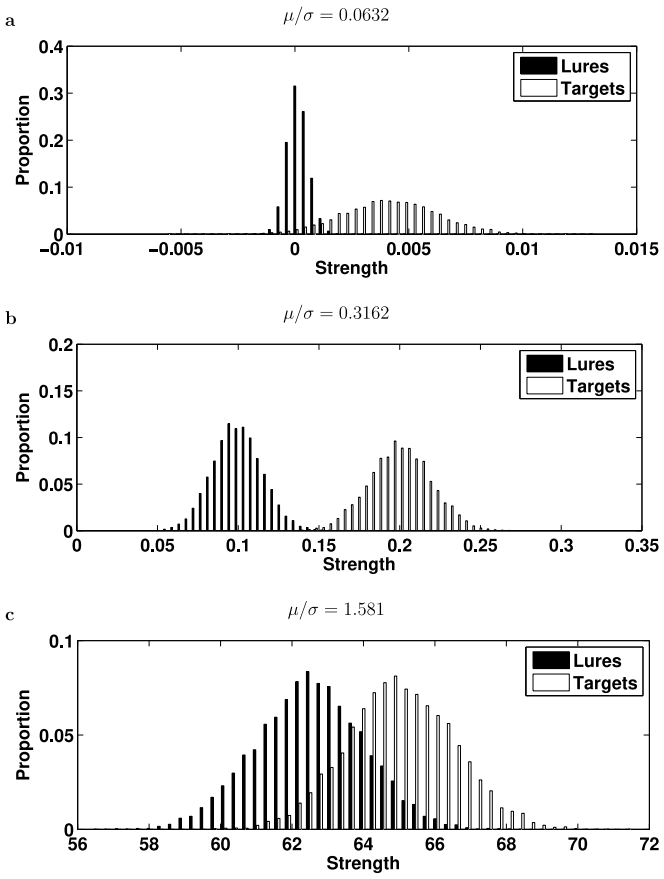

Fig. 4. Probability distributions of matching strengths for targets and lures, from the numerical simulation with $n = 1000$, $L = 10$, averaged across 1000 simulated lists. (a) $\mu = 0.002$, $d' = 2.66$. (b) $\mu = 0.01$, $d' = 5.59$, (c) $\mu = 0.05$, $d' = 1.65$.

effective in showing the generality of effects, one would like to evaluate the magnitudes of those effects when one implements a more complete and realistic simulation, which requires consideration of the plausibility of model parameters, and in the case of a spiking network, forces one to use strictly non-negative feature values encoded in nearly discrete events (action potentials). For tractability, the analytic models above allowed negative element values, in the negative tails of Gaussian distributions. In particular, we sought to test whether the (a) the match to incorrect targets would be substantially, not just nominally, greater than for non-studied items, and (b) d' for item-recognition based upon cueing with a constant input vector would be substantially, not just nominally, above chance. Depicted schematically in Fig. 5, we implemented the matrix model by setting matrix cell values as synaptic strengths, in an artificial neural network, using the Izhikevich (2003) model of simple spiking neurons.

4.1. Methods

4.1.1. Single-neuron model

The Izhikevich (2003) model can reproduce many distinct firing patterns of neurons while maintaining reasonable computing overhead. All simulated neurons are regular spiking neurons, with simulation parameters $a = 0.02$, $b = 0.2$, $c = -65$, $d = 8$, hand-chosen from parameter space maps in Izhikevich (2003).

4.1.2. Model synapse

Synapses are modeled using the alpha function approach following Dayan and Abbott (2001). Consider two neurons, where

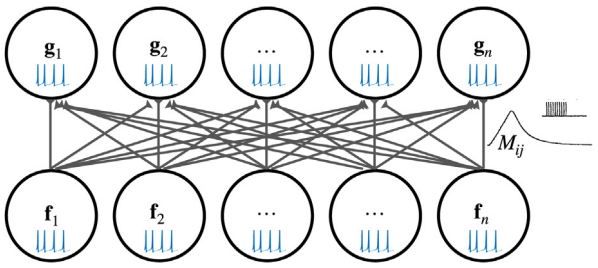

Fig. 5. Schematic illustration of the realistic spiking model. See text for details. The $\mathbf{f}_i$ units comprise the input layer and the $\mathbf{g}_i$ units comprise the output layer. Synapses are hard-coded to be equal to the cells of the matrix (Association Model).

one neuron, the input neuron, synapses onto the other neuron, the output neuron. When an input neuron fires, the output neuron's membrane potential changes. For synaptic input J, the change in post-synaptic membrane potential, P_s, as a function of time since firing, t, is

$$P_s(t) = J\frac{P_{\max}t}{\tau}e^{1-t/\tau}, \tag{32}$$

where $P_{max} = 5\ \mu V$ is the peak post-synaptic potential, $\tau = 3$ ms is time to reach peak post-synaptic potential.

4.1.3. Item representations and design of the network model

Because we wish to represent item vectors with firing rates, and firing rates must be positive-valued, there will always necessarily be a bias. Although one could approach the realistic spiking network in several other ways, the following approach made the most sense to us. We conceptualized the "lexicon" as consisting of signed vectors representing items that were, in some sense, known to the network. Thus, the lexicon includes items that could be identified as candidate responses, although we do not model the response process, itself. In the current simulation, we compute matches of the retrieved firing-rate pattern (vector) with vector-representations of items in the lexicon and use these matching strengths to estimate probability of producing a given item as a response, presumably through redintegration, which we also do not model explicitly. Thus, the idealized item representations include positive and negative values, which allows for the possibility of negative signs influencing synaptic weights negatively (not just positively), but whenever items are to be represented with firing rates (input and output layers), they are strictly non-negative. To our view, this was the least complicated, and thus, least objectionable, way to generate representations with a bias, albeit not the least complicated to describe to the reader.

To be more specific, we chose to draw firing-rate values at random from a Gaussian distribution with fixed standard deviation and mean approximately equivalent to the bias. We say "approximately" because Gaussian functions are, in principle, infinite in domain. Because negative firing rates are not possible, we maintained the negative values in the "lexicon" (set of items "known" to the model). Similarly, for cued recall, we injected negative current when the corresponding element of the cue item was negative. Negative values were also maintained when computed synaptic weights; thus, synaptic weights could be net-inhibitory as well as excitatory. This implementation choice means that the match between the retrieved vector and the vector in the lexicon can never be perfect, but we judged this to add realism to the model, and lend additional credibility to its performance level. Thus, $2L$ traces were randomly generated, where each trace was a list of n real numbers. The first L traces were defined as input

traces (i.e., left-hand items of the associations), $\mathbf{f}_i$, $i = 1..L$, and the remaining L traces were output traces (i.e., right-hand items of the associations), $\mathbf{g}_i$, $i = 1..L$. Then, we initialized $2n$ Izhikevich neurons, where the first n neurons ($n = 1000$ in the simulations) comprised the input layer neurons and the last n neurons comprised the output layer.

The synaptic weights were computed by summing the outer products of each output trace and its corresponding input trace. The matrix was then scaled to have a maximum element value of 1:

$$M = \frac{\sum_{i=1}^{L} \mathbf{g}_i \mathbf{f}_i^{\mathsf{T}}}{\max(\sum_{i=1}^{L} \mathbf{g}_i \mathbf{f}_i^{\mathsf{T}})}. \tag{33}$$

In other words, we did not model incremental learning, but simply assumed that the network has already learned and stored the summed matrix outer products in its synaptic strengths.

In each simulation, the neurons were initially left at rest (i.e., no input) for 1 s so they could reach their resting potential –70 mV.

4.1.4. Cued recall

The input current was computed by multiplying the values of the cue item by an arbitrary scalar value, 10 mA, that converted element values into current. This input current was applied as a square wave to the input-layer neurons for 100 ms. The number of action potentials fired by each output-layer neuron during a 100 ms estimation window was recorded. We calculated the similarity (dot product) of this retrieved firing-rate vector to the item representations, separately considering the target item, the other studied target items and one non-target items, a newly, randomly generated vector.

4.1.5. Recovering item memory with constant input

For this simulation, the input current was a vector for which all elements had the same, arbitrarily chosen current values, 10 mA, also applied to the input layer neurons for 100 ms. As before, the number of action potentials fired by each neuron was recorded, and this firing-rate vector was compared, via dot product, to the "target" vectors (right-hand items within studied pairs), newly, randomly generated item vectors, as well as "probe" vectors (left-hand items within studied pairs).

4.2. Results

4.2.1. Cued recall

The spiking-neuron network model was simulated with $n = 1000$, varying μ/σ over the range 0–1. As illustrated in Fig. 6a,b, and consistent with the symbolic model, the average match to the correct target item was greatest, and this was the case across the entire range of bias values. As the bias rises from zero, the other studied (but incorrect) target items match less than the correct target, but more than non-studied items. As bias increases toward infinity, the three item-types converge, consistent with the intuition that the larger the bias, the more similar all item-vectors will be.

4.2.2. Recovering item memory with constant input

When excited with constant current, the retrieved firing-rate pattern on the output layer was compared (via dot product) with the studied target items, as well as non-studied items (Fig. 6c). Fig. 6c also shows the matching strengths to "probe" (left-hand) items within the studied pairs. Due to the non-commutativity of the outer product, these match no better (nor worse) than non-studied items. The d' values are comfortably above chance, and reach levels in the range 2–3, that are typical of human recognition-memory performance on typical experimental tasks.

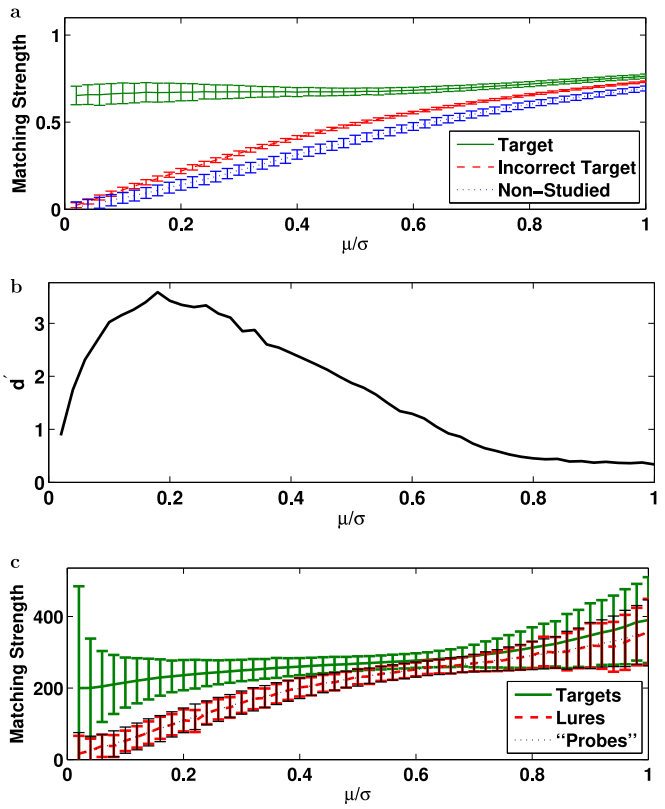

Fig. 6. Simulation of the realistic spiking model. (a) Tested with cued recall, plotting matching strength (normalized dot product) relative to strength of the correct target. (b,c) Item memory retrieved from the spiking-neuron Association Model with constant-current input. Means and standard deviations of strengths for targets (right-hand items within studied pairs), lures and "probes" (left-hand items within studied pairs) are depicted in (c). Horizontal axes (μ/σ) are aligned for comparison across panels. Note that for the simulation, μ/σ refers to the theoretical, not actual values observable in firing rates, which are truncated at zero.

As with the symbolic model, performance reduces to chance ($d' = 0$) when there is no bias ($\mu/\sigma = 0$), but rises rapidly with the introduction of only a very small bias. When the bias becomes larger, d' lessens, due to the increase in similarity between all pairs of item vectors. Also as with the symbolic model, there is a clear optimum bias level. Note that the horizontal scales are not directly comparable; for the network simulation, μ/σ is the theoretical value, based on the original item representations, whereas the functional bias is larger, due to the absence of negative firing rates.

4.3. Discussion

In sum, the spiking-neuron network simulations confirm that the phenomena expected based on the symbolic model (both analytic derivations and numerical simulations) may also occur within realistic parameter ranges, in models that are more constrained by neurophysiology (e.g., discrete action potentials and an absence of negative firing rates).

5. General discussion

In both the symbolic model and the more realistic, spiking-neuron network model, the introduction of an element-value bias in item representations was found to reduce performance in cued recall, as has been long understood. However, two new

phenomena were identified. First, cued recall produced intermediate matches to items that were studied as potential target items within the same study set. This may drive so-called within-list intrusion responses. Second, when probed with a vector containing zero item information (essentially, only the bias value itself), the retrieved vector was not meaningless (i.e., noise), but resembled the classic Item Model (Anderson, 1970). Derivations and simulations both demonstrated that for a range of bias values, this vector-sum could be used to perform at moderately high levels in old/new recognition judgments of the studied target items.

As Anderson (1970) also noted, the greater the bias, the worse both the Item and Association models perform. One might consider that this could be rectified by correcting for the expected additive constant (subtracting it) once the model is probed. This could correct the offset that is common to all retrieved vectors. However, the bias that is present during encoding introduces additional variance that cannot, subsequently, be corrected for.

The Association Model is composed of (studied) item vectors. It is interesting to consider that, when mean-centered, there is no way to directly retrieve those items; one must cue an item specifically with its corresponding paired item. This has led modelers to maintain separate memory variables for item versus associative information (Anderson, 1970), or use tensor space differently (Humphreys et al., 1989). Even in composite representations, modelers have included separate item and association terms (Metcalf Eich, 1982; Murdock, 1982). Simply dropping mean-centering and including a constant bias adds an affordance. As we have shown, the Association Model can now be probed with a constant input vector, containing zero information about studied items or associations, to retrieve a weighted sum of studied items. Equivalent to the Item Model, this is sufficient, for certain parameter values, to support item-recognition judgments.

We wondered why, to our knowledge, the tendency for non-target studied items to partially activate during cued recall has not previously been reported. We speculate that modelers may have seen traces of the "ghosts" of other studied items in their network activity, but that this seemed tangential, did not undermine their main goals with their models, and drew no further investigation. Indeed, examples of this phenomenon may be present in published modeling work. In addition, if item representations are generated at random, or do not have patterns that are readily recognized by eye, in plots of model activity, the "spillover" activation may not be obvious upon visual inspection. In cases for which the representations are chosen to be visually recognizable, the effects we derived may be evident. One example of this is Barkai, Bergman, Horwitz, and Hasselmo (1994), where the "ghosts" of non-target studied items can be seen in their Figures 5 and 7.

To our knowledge, there is not precedent for the spinoff function, the ability to perform item recognition by activating the associative network with constant current. Although performance reached reasonably high levels, even for the spiking-neuron network simulation, this approach to item-recognition has clear limitations. First, the perfectly constant current in our analytic derivations and network simulations is unrealistic; in reality, some level of noise would be expected. Current model performance could be viewed as outlining upper limits of performance, but not necessarily; in the spiking network model sometimes noise actually enhances encoding, as in stochastic resonance (Stein, Gossen, & Jones, 2005). However, added noise would presumably affect the overall performance of the model, and not result in qualitatively different dynamics. We expect the character of our results would be somewhat robust to the addition of noise.

Second, Anderson's Item Model is simple, yet can support item-recognition performance to a rather high level. It is also equivalent to the item-memory term of TODAM (Murdock, 1982), another model that has succeeded in accounting for a broad range of empirical memory findings. However, this vector-summation model has well known limitations. Most importantly, if $\oplus$ denotes concatenation, then if the Item Model stores two items, $\mathbf{m} = (\mathbf{a} \oplus \mathbf{b}) + (\mathbf{c} \oplus \mathbf{d})$ (assuming, for the moment, that all half-length vectors, $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$, $\mathbf{d}$ are mutually orthogonal and normalized), the match to studied items, $\mathbf{m} \cdot (\mathbf{a} \oplus \mathbf{b}) = \mathbf{m} \cdot (\mathbf{c} \oplus \mathbf{d}) = 1$, but the matches to items composed of recombining the parts are exactly the same, $\mathbf{m} \cdot (\mathbf{a} \oplus \mathbf{d}) = \mathbf{m} \cdot (\mathbf{c} \oplus \mathbf{b}) = 1$. Having studied (e.g., Mayes, Montaldi, & Migo, 2007) snowman and sandstorm, the model would just as readily endorse snowstorm and sandman as belonging to the study set. This is one reason modelers have developed more sophisticated models of recognition memory, such as MINERVA 2 (Hintzman, 1984) and REM (Shiffrin & Steyvers, 1997). Both these models are essentially local-trace models that include nonlinearities to produce greater matching evidence when all features of a single trace match than when matching features are distributed across traces. Alternatively, item memory can be stored associatively, in auto-associative terms, as is common in artificial neural network models, but also in a convolution-based model, CHARM (Metcalf Eich, 1982). The spinoff recognition-memory ability demonstrated here would be challenged by intra-item recombined recognition probes. We are not suggesting a return to the vector-sum as a model of item-memory. We are only suggesting that this spin-off information could explain (a) the relatively high prevalence of within-list intrusions in cued recall and (b) how item memory could be derived from stored associations without any additional process required to encode items. Thus, the item-recognition capability of the Association Model without mean-centering may be useful, and may be a good account of human memory behavior in certain conditions. When a particular item-recognition situation demands more fine-grained comparisons, it may quickly become insufficient, and additional, separate encoding terms may be required.

We chose to use the same value of μ during encoding, cued recall and matching to the lexicon (the process that presumably drives redintegration). This choice was in part for simplicity, and in part because we felt it plausible that if item representations need to include a bias of μ during encoding due to some physiological constraints, the same constraints probably hold always. It would be interesting to examine what would happen if the bias could take on different values at different times. However, one quickly appreciates that a $\mu' \neq \mu$ at retrieval may have little effect. Probing the Association Model with a constant vector, $\mu'\mathbf{I}$, Eq. (18) becomes

$$\mathbf{m}_r = M(\mu'\mathbf{I}) = \sum_{i=1}^{L} \alpha_i(\mathbf{g}_i + \mu\mathbf{I})(\mathbf{f}_i + \mu\mathbf{I})^{\mathsf{T}}\mu'\mathbf{I}, \tag{34}$$

but μ' is clearly just a scalar that will be present in all subsequent calculations of matching strengths, thus not affecting the competitive advantage of studied items over non-studied items.

Next, we consider an issue first raised by Ratcliff, Sheu, and Gronlund (1992). They criticized the matched-filter model (embedded within TODAM; Murdock, 1982) for predicting that the variance of strength for targets and for lures to be nearly equal. This contrasted with empirical measures that estimated lure:target variance ratios of approximately 0.8, derived from ROC curves (Ratcliff, McKoon, & Tindall, 1994; Ratcliff et al., 1992), and corroborated with analyses of response-time distributions (Osth, Dennis, & Heathcote, 2017; Starns & Ratcliff, 2014).[1] A result

[1] We thank Adam Osth for noting the connection to this prior work.

we had not anticipated was that the variance of recognition-strengths for target items was larger than that for lure items, particularly as μ increased. This is evident in Fig. 4, but it is true both of recognition from the Association Model (probed with constant input) and from the Item Model. First, we note that the ratio of variances will always be strictly less than 1. In the Item Model, Eq. (10), σ^2_{target} differs from σ^2_{lure} only as V_{ii} differs from V_{ij}. Eqs. (6) and (8) show that V_{ii} includes V_{ij} plus additional non-negative terms, ensuring that it will always be greater. Specifically, for the Item Model with μ included:

$$\frac{\sigma^2_{\text{lure}}}{\sigma^2_{\text{target}}} = \frac{LV_{ij}}{(L-1)V_{ij}+V_{ii}} = \frac{L(1/n+2\mu^2)}{(L-1)(1/n+2\mu^2)+2/n+4\mu^2} = \frac{L}{L+1}, \tag{35}$$

which (a) does not depend on the value of μ, (b) depends on L and (c) agrees with the solution found by Ratcliff et al. (1992).

Second, the same argument holds for the distributions of target and lure variances of item-recognition strengths retrieved from the Association Model, when one compares Eqs. (24)–(26) to Eqs. (27)–(29). Solving for the ratio of variances by substituting the expressions for variances into Eq. (10):

$$\frac{\sigma^2_{\text{lure}}}{\sigma^2_{\text{target}}} = \frac{\left(L^2+2L\right)n^3\mu^4+\left(L\,n^2+2\,L\,n\right)\,\mu^2+L}{\left(L^2+4L\right)n^3\mu^4+\left((L+3)n^2+(2L+2)n\right)\mu^2+L+n+1}. \tag{36}$$

For non-zero μ, as n becomes large, the $O(n^3)$ terms dominate, and the ratio of variances approaches $(L+2)/(L+4)$, which again, approaches 1 for large L, but is considerably below 1 for small L. The ratio of variances, thus, is strictly less than 1, but for non-zero μ, approaches 1 (equal variance) as the list length, L, increases. While not conclusive, this demonstrates that the criticism of approximately equal variances for the standard matched-filter model applies to the mean-centered model, but if one relaxes mean-centering, the model has the flexibility (via trade-offs of μ, L and n) to produce a wide range of ratios. This could be seen as another serendipitous outcome of the lack of mean-centering. That said, the dependence on list length of the ratio of variances is not supported empirically (Ratcliff et al., 1994), so the match of the non-mean-centered match-filter model to data is not clear-cut, and would require at least some additional modification.

The idea that item-memory is stored as a side-effect of association-memory encoding, but not necessarily vice versa, is reminiscent of Hockley and Cristi (1996). They found that participants performed far better on tests of associative recognition when they were instructed to study associatively, in anticipation of an association-memory test, than when they were instructed to study items in isolation, in anticipation of a subsequent item-memory test. However, item-recognition was just as good when participants studied in anticipation of an associative recognition test as when they were expecting an item-memory test. That said, the connection to Hockley and Cristi's findings is not straight-forward. Our derivations showed that the non-mean-centered model, while it did provide the ability to perform item-recognition above chance (and quite well, for certain parameter values), would always underperform the original Item Model. This raises the possibility that Hockley and Cristi's findings might be alternatively viewed as actually reflecting a deficit in item memory when participants study for associations, that is approximately offset by the spillover item-memory capability due to the non-zero μ.[2] Our reasoning suggests that if the component items were mix-and-match sets (such as snowstorm, snowman, sandstorm and sandman), studying for associations would, indeed, compromise item-recognition — a prediction that could be tested empirically (for a preliminary suggestion of this result, see Cox & Criss, 2017).

Although we have considered only a constant-valued vector ($\mu\mathbf{I}$), which is equivalent to violating mean-centering, the "spinoff" item-memory effects would presumably also be found with any shift of origin. One could envision a model in which each list "context" was implemented by shifting the origin for the set of items associated on the list.[3] An interesting possibility would be to combine such an extended model, whereby a shift in origin is used to represent context, with a model like the dynamic model of Cox and Shiffrin (2017), that probes jointly with item and context features. In such a model, context, conceived of as a shift in origin, might essentially pre-activate items studied within the particular context.

Another common representation in neural network models is sparse-coding (Tsodyks & Feigel'man, 1988), which can produce vectors that are mutually orthogonal while not mean-centered. However, the "background" discharge rate of cortical neurons introduces an additional mean across the population which, although small, cannot be ignored given that the standard deviation of element values with sparse-coding is also small. Consequently, one expects the same phenomena reported here with non-sparse item representations: an elevated rate of intrusions to incorrect target items and the ability to activate an item-memory vector that can support old/new recognition of all studied target items.

Finally, a major competing mathematical framework for association-memory, also assuming items are represented as vectors, is convolution. Convolution is a mathematical operation, denoted $*$ that combines two vectors into a new vector, thus $\mathbf{m} = \mathbf{f} * \mathbf{g}$. It is the central operation in TODAM (Murdock, 1982) and CHARM (Metcalf Eich, 1982). Although in most ways, convolution and matrix outer product can account for behavioral data equivalently (e.g., Murdock, 1985; Pike, 1984; Plate, 1995), a major difference is that convolution is commutative. That is, $\mathbf{f} * \mathbf{g} \equiv \mathbf{g} * \mathbf{f}$. Convolution-based models are thus oblivious to the order of items within the association. Matrix outer product, in contrast, is non-commutative; specifically, $\mathbf{gf}^{\intercal} = (\mathbf{fg}^{\intercal})^{\intercal}$. The non-commutativity of the outer product is what, in the modeling work presented here, caused the target items, but not the probe items, to be activated by cued-recall probes, as well as the constant-input probe, $\mathbf{I}$. In contrast, a bias introduced into a convolution-based model might, in fact, retrieve a weighted sum of all would-be cue and target items. Future work could address the same model phenomena within convolution models. Whether within-list intrusions are predominantly target items, or both cue and target items, could be settled empirically.

References

Anderson, J. A. (1970). Two models for memory organization using interacting traces. *Mathematical Biosciences*, *8*, 137–160.

Anderson, J. A. (1973). A theory for the recognition of items from short memorized lists. *Psychological Review*, *80*(6), 417–438.

Barkai, E., Bergman, R. E., Horwitz, G., & Hasselmo, M. E. (1994). Modulation of associative memory function in a biophysical simulation of rat piriform cortex. *Journal of Neurophysiology*, *72*(2), 659–677.

Cox, G. E., & Criss, A. H. (2017). Parallel interactive retrieval of item and associative information from event memory. *Cognitive Psychology*, *97*, 31–61.

Cox, G. E., & Shiffrin, R. M. (2017). A dynamic approach to recognition memory. *Psychological Review*, *124*(6), 795–860.

[2] We thank Greg Cox for the more subtle dimensions of this logic.

[3] We thank Michael J. Kahana for this suggestion.

Dayan, P., & Abbott, L. F. (2001). *Theoretical neuroscience: Computational and mathematical modeling of neural systems*. Cambridge, MA, USA: MIT Press.
Eliasmith, C., Stewart, T. C., Choo, X., Bekolay, T., DeWolf, T., Tang, Y., et al. (2012). A large-scale model of the functioning brain. *Science*, *338*, 1202–1205.
Franklin, D. R. J., & Mewhort, D. J. K. (2015). Memory as a hologram: an analysis of learning and recall. *Canadian Journal of Experimental Psychology*, *69*(1), 115–135.
Hintzman, D. L. (1984). MINERVA 2: A simulation model of human memory 2: A simulation model of human memory. *Behavior Research Methods, Instruments, & Computers*, *16*(2), 96–101.
Hockley, W. E., & Cristi, C. (1996). Tests of encoding tradeoffs between item and associative information. *Memory & Cognition*, *24*, 202–216.
Howard, M. W., & Kahana, M. J. (1999). Contextual variability and serial position effects in free recall. *Journal of Experimental Psychology. Learning, Memory, and Cognition*, *25*(4), 923–941.
Howard, M. W., & Kahana, M. J. (2002). A distributed representation of temporal context. *Journal of Mathematical Psychology*, *46*(3), 269–299.
Humphreys, M. S., Bain, J. D., & Pike, R. (1989). Different ways to cue a coherent memory system: A theory for episodic, semantic, and procedural tasks. *Psychological Review*, *96*(2), 208–233.
Izhikevich, E. M. (2003). Simple model of spiking neurons. *IEEE Transactions on Neural Networks*, *14*(6), 1569–1572.
Mayes, A., Montaldi, D., & Migo, E. (2007). Associative memory and the medial temporal lobes. *Trends in Cognitive Sciences*, *11*(3), 126–135.
Metcalf Eich, J. (1982). A composite holographic associative recall model. *Psychological Review*, *89*(6), 627–661.
Murdock, B. B. (1982). A theory for the storage and retrieval of item and associative information. *Psychological Review*, *89*(6), 609–626.
Murdock, B. B. (1985). Convolution and matrix systems: a reply to Pikeike. *Psychological Review*, *92*(1), 130–132.
Osth, A. F., & Dennis, S. (2015). Sources of interference in item and associative recognition memory. *Psychological Review*, *122*(2), 260–311.
Osth, A. F., Dennis, S., & Heathcote, A. (2017). Likelihood ratio sequential sampling models of recognition memory. *Cognitive Psychology*, *92*, 101–126.
Pike, R. (1984). Comparison of convolution and matrix distributed memory systems for associative recall and recognition. *Psychological Review*, *91*(3), 281–294.
Plate, T. A. (1995). Holographic reduced representations. *IEEE Transactions on Neural Networks*, *6*(3), 623–641.
Ratcliff, R., McKoon, G., & Tindall, M. (1994). Empirical generality of data from recognition memory receiver-operating characteristic functions and implications for the global memory models. *Journal of Experimental Psychology. Learning, Memory, and Cognition*, *20*(4), 763–785.
Ratcliff, R., Sheu, C. F., & Gronlund, S. D. (1992). Testing global models using ROC curves curves. *Psychological Review*, *99*(3), 518–522.
Shiffrin, R. M., & Steyvers, M. (1997). A model for recognition memory: REM—retrieving effectively from memory— retrieving effectively from memory. *Psychonomic Bulletin & Review*, *4*, 145–166.
Starns, J. J., & Ratcliff, R. (2014). Validating the unequal-variance assumption in recognition memory using response time distributions instead of ROC functions: a diffusion model analysis functions: a diffusion model analysis. *Journal of Memory and Language*, *70*, 36–52.
Stein, R. B., Gossen, E. R., & Jones, K. E. (2005). Neuronal variability: noise or part of the signal? *Nature Reviews Neuroscience*, *6*(5), 389.
Tsodyks, M. V., & Feigel'man, M. V. (1988). The enhanced storage capacity in neural networks with low activity level. *Europhysics Letters*, *6*(2), 101–105.
Weber, E. U. (1988). Expectation and variance of item resemblance distributions in a convolution-correlation model of distributed memory. *Journal of Mathematical Psychology*, *32*(1), 1–43.