Modelling constituent order despite symmetric associations in memory

Jeremy J. Thomas

Department of Psychology, University of Alberta

Jeremy B. Caplan

Department of Psychology, and Neuroscience and Mental Health Institute, University of Alberta

Abstract

Mathematical models of association memory (study AB, given A, recall B) either predict that knowledge for constituent order of a word pair (AB vs. BA) is perfectly unrelated, or completely dependent on knowledge of the pairing itself. Data contradict both predictions; when a pair is remembered, constituent-order is above chance, but still fairly low. Convolution-based models are inherently symmetric and can explain associative symmetry, but cannot discriminate AB from BA. We evaluated four extensions of convolution, where order is incorporated as item features, partial permutations of features, item-position associations, or by adding item and position vectors. All approaches could discriminate order within behaviourally observed ranges, without compromising associative symmetry. Only the permutation model could disambiguate AB from BC in double-function lists, as humans can do. It is possible that each of our proposed mechanisms might apply to a different, particular task setting. However, the partial permutation model can thus far explain the broadest set of empirical benchmarks.

Keywords: association memory; order memory; mathematical models; verbal memory; convolution; associative symmetry;

Introduction

Memory for associations forms the cognitive basis for a large portion of behaviour (Murdock, 1974; Lashley, 1951). In many cases, such as remembering face-name relationships at a dinner party, or that colorful snakes are poisonous, it is sufficient to remember

---

This research was supported by the Natural Sciences and Engineering Research Council of Canada. Corresponding author: Jeremy J. Thomas, jjthomas@ualberta.ca, Department of Psychology, Biological Sciences Building, University of Alberta, Edmonton, Alberta T6G 2E9, Canada, Tel:+1.780.492.5361, Fax: +1.780.492.1768. Model code is posted at `https://osf.io/r3ywf/`.

that stimuli are associated to each other. But sometimes it is important to remember an association along with its constituent-order (AB versus BA). Indeed, many examples of order-sensitive associations exist in language, such as modifier-head relationships in compound words, PAN CAKE versus CAKE PAN, or HOUSE GUEST versus GUEST HOUSE (Dressler, 2006; Caplan, Boulton, & Gagné, 2014). However, memory for order has typically not been a focus in the experimental study of verbal association memory. Standard tests of association memory ask participants to study pairs of words (AB), followed by cued recall (given A, respond with B). Participants can respond with B when given A, and vice versa, without knowing the constituent-order of the pairing. Moreover, memory for order is typically studied with separate tasks such as serial recall (study A, B, C, D, recall the list in order).

Consequently, mathematical models of association memory are quite poor at accounting for constituent-order, either assuming that associations are stored with perfect order, or with no order at all. Models based on convolution (Kelly, Blostein, & Mewhort, 2013; Murdock, 1982; Metcalfe Eich, 1982; Plate, 1995), and recent models within the REM framework (Cox & Criss, 2017, 2020; Criss & Shiffrin, 2005), assume associations are stored with no order. Thus, AB is mathematically equivalent to BA. The face-value prediction is that memory for constituent-order will be at chance. However, given evidence that participants can remember constituent-order above-chance (Greene & Tussing, 2001; Kato & Caplan, 2017; Kounios, Smith, Yang, Bachman, & D'Esposito, 2001; Kounios, Bachman, Casasanto, Grossman, & Smith, 2003; Yang et al., 2013), one might rescue convolution, and other symmetric models, by allowing for some additional source of information to support order judgments, such as an additional term in the model. The consequence of storing order separately from associations is that the models would predict that memory for constituent-order should be unrelated to memory for the pairing itself. The second type of prediction, that associations are stored with perfect order, comes from matrix models

(Anderson, 1970; Humphreys, Bain, & Pike, 1989; Osth & Dennis, 2015b; Pike, 1984) and models that concatenate the two item vectors (Hintzman, 1984; Shiffrin & Steyvers, 1997). These models can infer order with no ambiguity, predicting that memory for constituent-order (AB versus BA) should be perfect given that the association itself can be recalled.

Kato and Caplan (2017) tested these predictions with a task which we refer to as order recognition (Greene & Tussing, 2001; Kounios et al., 2001, 2003; Yang et al., 2013). Order recognition tests memory for constituent-order directly by presenting pairs in their original (AB) or reversed order (BA). Participants then provide a forced-choice judgment whether the probe is intact or reverse. One group of participants were tested with cued recall, and then order recognition for each studied pair, and compared to another group tested with associative recognition after cued recall instead.[1] Matrix models predict that order recognition performance should be perfect for correctly recalled pairs. Convolution models predict that order recognition performance should be equivalent for correct and incorrectly recalled pairs. Contradicting both predictions, order recognition was significantly better when cued recall was correct, but well below maximum, and well below associative recognition for correctly recalled pairs.[2] These results indicate that verbal associations are neither encoded with perfect order, nor are completely order-absent, inconsistent with assumptions in all models.[3]

Another clue about the representation of associations and their constituent order comes from from double function lists in Rehani and Caplan (2011), where cued recall was direction-specific. Double function lists (Howard, Jing, Rao, Provyn, & Datey, 2009;

[1]Both cued recall and associative recognition test memory for pairings between words and tend to be highly correlated. Thus, the cued recall-associative recognition group provided a realistic upper ceiling to compare the order recognition group against.

[2]Kato and Caplan (2017) also addressed the possibility that testing with cued recall influenced order recognition. In their second experiment they withheld half the pairs from cued recall testing, and in their third experiment moved cued recall to the end of the session. In both cases they found that the order-cued recall relationship persisted.

[3]Thomas, Ayuno, Kluger, and Caplan (2022) also came to similar conclusions, as we elaborate below.

Primoff, 1938; Rehani & Caplan, 2011; Slamecka, 1976), contain pairs, where each constituent item appears in two pairs, once in the left position, and once in the right position (AB, . . . , BC, . . . , CA, . . .). Consider a trial where B is presented as a cue on the left-hand side. Correctly responding with C requires knowledge of relative position/order, for example, that B appeared on the left in pair BC, but not AB. Performance is compared to single function pairs that do not share items (EF, . . . , GH, . . . , IJ, . . .). Because of their extreme assumptions about order, matrix and convolution models generate direct predictions about this task. A convolution model has no information to select between A and C. Thus, assuming the model guesses between two possible responses, convolution predicts cued recall accuracy for double function pairs will be one-half that of single-function pairs. In contrast, matrix-based models suffer no interference between AB and BC (see below). Therefore, the model predicts equal accuracy for double and single-function pairs. Contradicting both matrix and convolution model predictions, Rehani and Caplan (2011) found double-function cued recall accuracy was somewhat lower, but well above one-half of single-function accuracy, converging with evidence from the order recognition task that associations are neither stored order-absent, nor with perfect directionality.[4]

In sum, participants can discriminate AB versus BA during a word pair task (Greene & Tussing, 2001; Kato & Caplan, 2017; Kounios et al., 2001, 2003; Yang et al., 2013), and even use order/item-position information to aid cued recall (B ?) to solve AB versus BC interference (Rehani & Caplan, 2011). Taken together, this suggests that the constituent-order of verbal associations is explicitly stored, and in a way that is moderately dependent on memory for the pairing itself.

[4]One could argue that the ability to disambiguate double function pairs does not come from memory for order, but rather, because each item in these pairs was repeated, and thus more available in memory. However, Caplan, Rehani, and Andrews (2014) found when participants were able to respond with both associates for double function pairs, double and single function cued recall accuracy was equivalent, arguing against this confound.

Associative Symmetry

Despite evidence that associations are stored with moderate levels of order, there is also a sense in which verbal associations are rather symmetric. Initial support for idea, known as associative symmetry, arose from the stable tendency for forward cued recall accuracy (APPLE ?) and backward cued recall (? OVEN) accuracy to be equal on average (Asch & Ebenholtz, 1962; Horowitz, Brown, & Weissbluth, 1964; Kahana, 2002; Kato & Caplan, 2017; Murdock, 1962). However, Kahana (2002) showed that an asymmetric model could produce symmetry in mean cued recall accuracy, suggesting this result is not diagnostic of symmetric associations. Instead, Kahana (2002) proposed that associative symmetry should be tested at the pair level, with two cued recall trials for each word, and where test 1 and test 2 is either forward or backward cued recall. Indeed, multiple studies have returned a near-perfect correlation for incongruent conditions (forward-backward, backward-forward), that are remarkably close to what are essentially test-retest correlations for congruent conditions (forward-forward, backward-backward) (Kahana, 2002; Kato & Caplan, 2017; Rehani & Caplan, 2011; Rizzuto & Kahana, 2000, 2001; Sommer, Schoell, & Büchel, 2008). These findings either suggest forward and backward cued recall are testing the same bi-directional association in memory, or, that there are distinct forward and backward associations for a given pair, but these are highly correlated in their strengths (Kahana, 2002).

We were particularly interested in associative symmetry here because of the potential paradox between association memory that is highly symmetric, yet supports memory for its constituent-order. As we elaborate below, it was especially challenging in previous attempts to modify matrix models to simultaneously produce moderate order memory and associative symmetry (Kato & Caplan, 2017). A strong account of association memory should be able to account for both constraints, and thus, we include this as an additional

benchmark for all models.

Attempts to produce order and symmetry in current models

Given that associations are symmetric, yet support a moderate ability to judge constituent-order, how do existing models account for the potential tension between these constraints?

Matrix-based models. Associations are encoded as follows, $M = \mathbf{ab}^\intercal$, where M denotes the memory matrix, $\mathbf{a}$ and $\mathbf{b}$ represent item vectors, and $\intercal$ denotes transpose. Bold-face indicates column vectors. Cued recall is modelled with matrix multiplication, for example, $M\mathbf{b} \approx \mathbf{a} + noise$. Matrix multiplication is direction sensitive, meaning that $\mathbf{b}^\intercal M \approx 0 + noise$. By comparing the outputs of $M\mathbf{b}$ and $\mathbf{b}^\intercal M$, the model can unambiguously infer that item $\mathbf{b}$ appeared in the left position. For similar reasons, matrix models also have a perfect ability to solve double function interference. If two pairs that share an item are stored in memory, $M = \mathbf{ab}^\intercal + \mathbf{bc}^\intercal$, the direction specificity of forward and backward cued recall means that a given item vector $\mathbf{b}$ can cue completely different pairs in memory based on direction, $M\mathbf{b} \approx \mathbf{a}$ and $\mathbf{b}^\intercal M \approx \mathbf{c}$. One can eliminate this directionality by simultaneously storing the forward and reverse association, $\alpha_f \mathbf{a}^\intercal \mathbf{b} + \alpha_b \mathbf{b}^\intercal \mathbf{a}$, where α_f and α_b are scalar random values that represent variable encoding strengths. Assuming that α_f and α_b are perfectly correlated, and that $\mathrm{E}\left[\alpha_f\right] = \mathrm{E}\left[\alpha_b\right]$, this model can produce perfect associative symmetry (Kahana, 2002), but as a direct consequence, cannot discriminate AB from BA (Kato & Caplan, 2017) or solve double function lists (Rehani & Caplan, 2011). To regain some ability to disambiguate AB from BA, $\mathrm{E}\left[\alpha_f\right]$ could be increased relative to $\mathrm{E}\left[\alpha_b\right]$, so that the forward association is stronger in memory; however, the model now produces a forward recall advantage violating associative symmetry, and predicts order recognition performance would positively correlate with the difference between forward and backward cued recall performance. Kato and Caplan (2017) found no evidence for

the latter prediction; these correlations were not significant. Kato and Caplan (2017) also tested a matrix model that always stored a definite order, but sometimes encoded pairs in the incorrect order with probability p_{rev}. Increasing p_{rev} reduced the model's order recognition performance, even to the moderate levels seen in behaviour. However, the model assumes that even wrong order judgments are made with perfect certainty, because they come from perfectly directional associations in memory. The resulting prediction is that participants should be unlikely to switch their response if they are tested twice for order recognition, correct-correct or incorrect-incorrect judgments should be most frequent. This prediction was also unsupported in Kato and Caplan's (2017) data—participants did not stick with their order judgments more frequently than they switched their order judgments. Along with evidence from other analyses, order judgments seem to not be made with perfect certainty, but are rather more like uncertain, noisy decisions that are prone to change on retest.

Convolution-based models. Convolution models do not store order at all. Associations are stored as follows, $\mathbf{m} = \mathbf{a} * \mathbf{b}$, where $\mathbf{a}$ and $\mathbf{b}$ denote item vectors, $\mathbf{m}$ denotes the memory vector, and $*$ denotes circular convolution.[5] If $\mathbf{a}$ and $\mathbf{b}$ are n-dimensional vectors, with each element sub-scripted from 0 to $k-1$, circular convolution is defined as follows,

$$\mathbf{m}_i = \sum_{k=0}^{n-1} \mathbf{a}_k \mathbf{b}_{(i-k) \bmod n}, \tag{1}$$

Where $\mathbf{m}$ is an n-dimensional vector. Importantly, convolution is strictly commutative, $\mathbf{a} * \mathbf{b} \equiv \mathbf{b} * \mathbf{a}$. This property causes convolution to naturally produce associative symmetry (Kahana, 2002), but also means that there is no way to recover the constituent-order of the pair after encoding. To retain order information in a convolution model, one could permute the elements of item-vectors before encoding (Jones & Mewhort, 2007; Kelly et

[5] Models such as Plate (1995) have used circular convolution. From now on, convolution refers specifically to circular convolution.

al., 2013; Plate, 1995; Recchia, Jones, Sahlgren, & Kanerva, 2010), expressed as follows, $\mathbf{m} = p_l(\mathbf{a}) * p_r(\mathbf{b})$, where p denotes permutation operator, and subscript l and r indicate the position-specific permutation pattern applied to each vector. Permutation allows convolution to encode order-sensitive relationships (Jones & Mewhort, 2007), along with other useful side-effects (Kelly et al., 2013). In published implementations, the whole vector is permuted which effectively implements a non-commutative operation, more like a matrix-outer product, $p_l(\mathbf{a}) * p_r(\mathbf{b}) \neq p_r(\mathbf{a}) * p_l(\mathbf{b})$. For this reason, fully permuting item vectors may be incompatible with empirical data in a similar way as an unmodified matrix model. However, we do test this idea, with a small twist, below.

Four ways to extend convolution to store order

In sum, the concurrent empirical constraints of associative symmetry and moderate order memory prove difficult for all existing models. Convolution models and modified matrix models can produce perfect associative symmetry, but disregard order, while non-commutative versions of both matrix and convolution models over-predict the degree to which order is remembered. One could address these challenges with two possible approaches, either modify non-commutative models to have reduced order memory, or extend symmetric models to store order. In the present article we take the latter approach.

Our objective here is not to fundamentally alter basic model mechanisms, but design modifications that store order while preserving useful characteristics, like associative symmetry, that make convolution a rich account of verbal association memory. To this end, all of our four models (Illustrated in Figure 1) are intentionally very simple, each consisting of only three free parameters. Furthermore, each model parameterizes order discrimination ability with one free parameter, as we describe below.

- **Model A** (Figure 1a): Order is encoded as explicit associations between item vectors and "position" vectors, bearing some resemblance to positional-coding models of

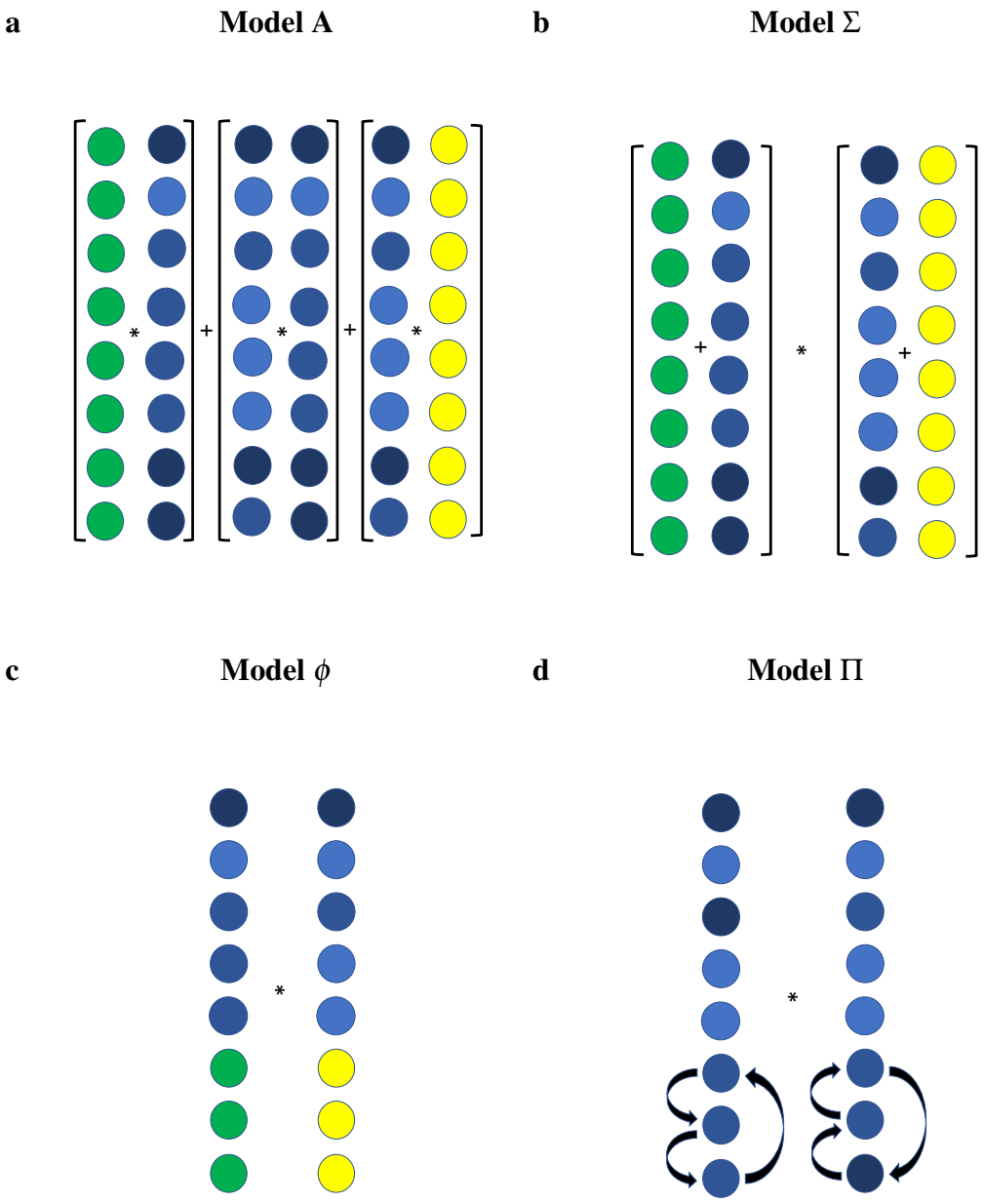


Figure 1. Four different mechanisms to store the constituent-order of associations within a convolution model. Blue circles denote item features, green circles correspond to the left position, and yellow circles correspond to the right position. Note, the color of the circles were for illustrative purposes, and do not indicate that features were the same value.

serial recall (Conrad, 1960; Brown, Neath, & Chater, 2007; Burgess & Hitch, 1999; Farrell, 2012; Henson, 1998), or item-context associations in the Temporal Context Model (Howard & Kahana, 1999) but with just two unique position vectors. These two associations for the left and right positions are stored along with the item-item association,

$$\mathbf{m}_{\mathrm{A}} = \sum_{i=1}^{L} \alpha_i \left((\mathbf{f}_i * \mathbf{l}) + (\mathbf{g}_i * \mathbf{r}) + (\mathbf{f}_i * \mathbf{g}_i) \right), \tag{2}$$

where $\mathbf{f}_i$, $\mathbf{g}_i$ are n-dimensional item-vectors, and $\mathbf{l}$ and $\mathbf{r}$ are n-dimensional position vectors, and L denotes list length or number of pairs stored in the memory vector $\mathbf{m}_{\mathrm{A}}$. Features values for all vectors are sampled from $N(0, \sigma^2)$, and then vectors are strictly normalized. Item-position, and item-item associations share an associative encoding strength α_i, which is a scalar value sampled from $N(\mu, \sigma_\alpha)$, and where σ_α, and μ are free parameters. Model A infers order by comparing a dot product between a correct item-position pair to the memory vector, $((\mathbf{f}_i * \mathbf{l}) + (\mathbf{g}_i * \mathbf{r})) \cdot \mathbf{m}_A$, and a dot product between an incorrect item-position pair and the memory vector, $((\mathbf{f}_i * \mathbf{r}) + (\mathbf{g}_i * \mathbf{l})) \cdot \mathbf{m}_A$. In our implementation of model A, we parameterize order discrimination ability by modifying the strength of item-position associations with single parameter, the mean associative encoding strength μ. By modifying the mean associative encoding strength μ, we can increase or decrease the match of a correct item-position pair to memory. Finally, because item-position and item-item associations share an associative encoding strength α_i, this ensures that memory for the association co-varies with memory for its order.

- **Model** Σ (Figure 1b): Similar to model A, position vectors are used to represent order but are instead added element-wise to each item before convolving, which is

mathematically similar to extensions of TODAM (Murdock, 1995) that summed item vectors before convolving,

$$\mathbf{m}_\Sigma = \sum_{i=1}^{L} \alpha_i((\mathbf{f}_i + \mathbf{l}) * (\mathbf{g}_i + \mathbf{r})), \tag{3}$$

where L, α_i, $\mathbf{f}_i$, $\mathbf{g}_i$, $\mathbf{l}$, and $\mathbf{r}$ are identical to their definitions in equation 2, and $\mathbf{m}_\Sigma$ denotes the memory vector. Interestingly, by expanding the encoding equation, $\mathbf{m}_\Sigma = \sum_{i=1}^{L} \alpha_i((\mathbf{f}_i * \mathbf{g}_i) + (\mathbf{g}_i * \mathbf{l}) + (\mathbf{f}_i * \mathbf{r}) + (\mathbf{l} * \mathbf{r}))$, we can see that this model is equivalent to model A (equation 2) with an additional noise term, $\mathbf{l} * \mathbf{r}$. This equivalency means that we can parameterize order discrimination ability in the same way as model A, by modifying the strength of item-position associations with a single parameter μ. Thus, if the model infers order by comparing a dot product between a correct item-position pair, $((\mathbf{f}_i + \mathbf{l}) + (\mathbf{g}_i + \mathbf{r})) \cdot \mathbf{m}_\Sigma$, and incorrect item-position pair to the memory vector, $((\mathbf{f}_i + \mathbf{r}) + (\mathbf{g}_i + \mathbf{l})) \cdot \mathbf{m}_\Sigma$, modifying the mean associative encoding strength μ can modify the match of a correct and incorrect item-position pair to memory.

- **Model ϕ** (Figure 1c): Order is encoded by incorporating dedicated positional feature values into the item vector alongside item-unique features. This bears some resemblance to the ways in which numerous models have incorporated attributes such as list context as specialized features. All items in the left position receive the same set of positional feature values, and likewise for right position items,

$$\mathbf{m}_\phi = \sum_{i=1}^{L} \alpha_i((\mathbf{f}_i \oplus \mathbf{l}) * (\mathbf{g}_i \oplus \mathbf{r})), \tag{4}$$

where L is defined as before, and $\mathbf{l}$ and $\mathbf{r}$ consist of n_p positional features that are concatenated (denoted by $\oplus$) onto item vectors $\mathbf{f}_i$ and $\mathbf{g}_i$ respectively, and $\mathbf{m}_\phi$ denotes the memory vector. Encoding strength α_i is drawn from $N(1, \sigma_\alpha)$, but note the follow-

ing difference from models A and Σ—σ_α is a free parameter and mean associative encoding strength is fixed at 1. This is because order discrimination is parameterized with the number of positional features n_p, instead of mean associative encoding strength (see below). Vectors $\mathbf{f}_i$ and $\mathbf{g}_i$ each consist of unique item features, and have $n - n_p$ dimensions to ensure that resulting dimensions of the full vector, with position features, is always equal to n. All feature values, including position features, are independently sampled from $N(0, \sigma^2)$, and item vectors, with position features, are strictly normalized. The order discrimination ability of model ϕ is parameterized by a single parameter, the number of positional features n_p. The model can infer order by comparing a dot product between a pair of items with correct position features to the memory trace, $(\mathbf{f}_i \oplus \mathbf{l}) * (\mathbf{g}_i \oplus \mathbf{r}) \cdot \mathbf{m}_\phi$, and a dot product between pair of items with incorrect position features to the memory trace, $(\mathbf{f}_i \oplus \mathbf{r}) * (\mathbf{g}_i \oplus \mathbf{l}) \cdot \mathbf{m}_\phi$. Increasing n_p increases the difference between these two matches, and thus overall order discrimination ability.

- **Model** Π (Figure 1d): To encode order, item-unique feature values are shuffled or permuted in a pattern that is specific to the position of that item vector. This mechanism is inspired by the use of permutation in other models (Jones & Mewhort, 2007; Kelly et al., 2013; Plate, 1995), but the key difference in our implementation is that here, only a subset of features are permuted, rather than the entire vector,

$$\mathbf{m}_\Pi = \sum_{i=1}^{L} \alpha_i (p_l(\mathbf{f}_i) * p_r(\mathbf{g}_i)), \tag{5}$$

where L is defined as before, and $\mathbf{f}_i$ and $\mathbf{g}_i$ are n-dimensional item vectors, of which n elements are independently sampled from $N(0, \sigma^2)$. Vectors are then strictly normalized, and $\mathbf{m}_\Pi$ denotes the memory vector. A distinct pattern of permutation is

applied to every left position item, denoted by p_l, and another pattern of permutation is applied to the right position item, denoted by p_r. Just like model ϕ, encoding strength α_i is drawn from $N(1, \sigma_\alpha)$, where σ_α is a free parameter and mean associative encoding strength is fixed at 1. The order discrimination ability of model Π is parameterized by a single parameter, the number of permuted features n_{perm}. Thus, similar to model ϕ, and unlike models A and Σ, μ is not a free parameter. The model can infer order by comparing a dot product between a pair of items with the correct position permutations to the memory vector, $p_l(\mathbf{f}_i) * p_r(\mathbf{g}_i) \cdot \mathbf{m}_\Pi$, and a dot product between a pair of items with incorrect position permutations and the memory vector, $p_r(\mathbf{f}_i) * p_l(\mathbf{g}_i) \cdot \mathbf{m}_\Pi$. Increasing n_{perm} increases the difference between these two matches, and thus overall order discrimination ability.

Summary of modelling approach

A major focus in this article is the challenge presented by order recognition data (Kato & Caplan, 2017; Thomas et al., 2022), which to our knowledge, has not been previously fit by models. To investigate whether each of our extensions of convolution can address this challenge, we fit each to both aggregate data and single participants, as two separate benchmarks. Single-participant data is better in the sense that it is less likely to arise from a mixture of mechanisms, and more likely to be model-pure, reducing the chance that the wrong model is favored. The disadvantage is that each participant has less data, so they are, in principle, more noisy than aggregated data. By including both single participant and aggregate model-selection, we could look for broad agreement between both benchmarks, increasing the robustness of any conclusions we make.

We also evaluate whether each model can account for double function list performance (Rehani & Caplan, 2011). While double function lists also provide a challenge to models for similar reasons as order recognition data, our intention is not to provide a com-

prehensive account of this task. Instead, we used double function lists to help characterize conditions under which certain order-encoding mechanisms may be more preferable. Accordingly, we kept this section brief, opting to use algebraic arguments and simulations, rather than quantitative fits to data. Our evaluation of these models will proceed as follows. First, we simulate order recognition, cued recall, and associative recognition with each model, to show the relationship between performance and key model parameters. Next, we fit models to order recognition data at the aggregate level, to determine if each can produce a moderate relationship between order recognition and cued recall, while preserving the near-perfect correlation between forward and backward cued recall (benchmark 1a). Next, we fit models to order recognition data for individual participants (benchmark 1b). Finally, we evaluate each model against double function lists (benchmark 2).

Empirical benchmark 1: Order recognition and associative symmetry

We first wondered whether each of these models could produce above-chance order recognition performance, with a moderate relationship to cued recall performance, and alongside the near-perfect symmetry between forward and backward cued recall.

Target data-set

We fit models to data from experiment 1 in Thomas et al. (2022).[6] This experiment included 227 participants total, with two experimental groups, a strategy-instruction group ($N = 117$) where participants received instructions to use a memory strategy, and a control group ($N = 114$). We only used data from the control group in the following fits, as it was most comparable to conditions in Kato and Caplan (2017).

The design of this experiment is illustrated in Figure 2, and the full methods can be found in Thomas et al. (2022). Briefly, each participant completed study, cued recall, and

[6] Data is posted at `https://osf.io/x78gp/`

recognition for eight lists total (excluding a practice list at the beginning of the experiment). For each list, participants viewed eight word pairs in sequence, where each pair presented for 2850 ms, with a 150 ms inter-trial interval. This was followed by eight cued recall trials which tested all studied pairs. Then, depending on the condition that each participant was assigned, this was followed by either eight order recognition trials ($N = 56$) or eight associative recognition trials ($N = 58$) which tested all studied pairs.[7] Interleaved between study, cued recall and recognition trials were five trials of a mathematical distractor task. Cued recall direction (forward versus backward), associative recognition probe type (intact versus recombined), and order recognition probe type (intact versus reverse) was counter-balanced across all lists. The design of this experiment was very similar to Kato and Caplan (2017) except that cued recall was only tested once per pair, meaning that the correlation between forward and backward cued recall could not be measured at the level of individual pairs. Instead, as we elaborate below, we used general ranges from previous experiments to check whether the correlations in the present models are consistent with previous reports. We fit models to data averaged across all eight test lists.

We derived two empirical benchmarks (1a and 1b) from this data-set to evaluate models. Benchmark 1a was order recognition performance separated by cued recall correctness (and associative recognition as a control), alongside the near-perfect correlation between forward and backward cued recall. Benchmark 1b was individual differences in order recognition performance, that occupied a range around the means observed in benchmark 1a. Here we wondered if models could not only produce means that characterized empirical data, but account for individual participants within the data-set using different parameter sets.

[7] Initially, both cued recall and recognition trials had 15000 ms time-limit, but this was removed halfway through data collection.

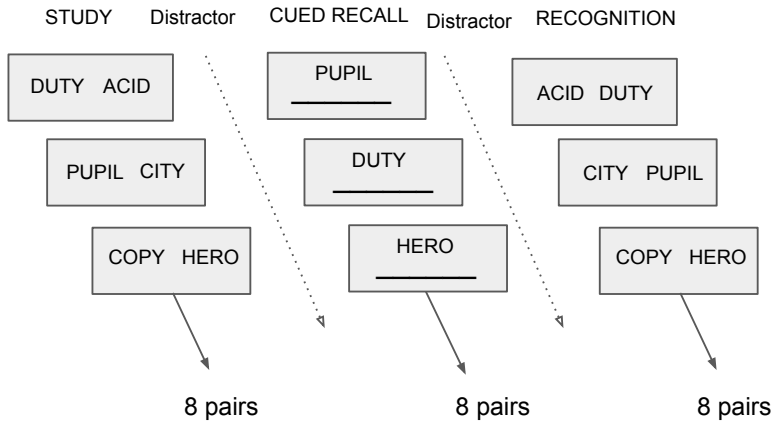


Figure 2. The design of one cycle from experiment 1 in Thomas et. al. (2022), adapted from figure 1 of Thomas et. al. (2022). The recognition task performed (associative versus order) was a between-subject factor. This procedure was repeated for a total of eight cycles, after which participants completed other tasks and a questionnaire not pictured here.

Simulation Methods

To begin, we describe how order recognition, cued recall and associative recognition are simulated in each of our models.

Encoding. Each model encodes a memory vector $\mathbf{m}$ according to its respective encoding expression defined above (Equations 2–5). Each memory vector $\mathbf{m}$ stores $L = 8$ unique pairings of 16 different items, matching Thomas et al. (2022).

Order recognition. Two dot products are used to assess model order recognition performance. First, a dot product between items with the correct position and the memory trace, defined as follows for models A, Σ, ϕ, and Π respectively,

$$\iota_A = \quad ((\mathbf{f}_i * \mathbf{l}) + (\mathbf{g}_i * \mathbf{r})) \cdot \mathbf{m}_A, \tag{6}$$

$$\iota_\Sigma = \quad ((\mathbf{f}_i + \mathbf{l}) + (\mathbf{g}_i + \mathbf{r})) \cdot \mathbf{m}_\Sigma, \tag{7}$$

$$\iota_\phi = \quad ((\mathbf{f}_i \oplus \mathbf{l}) * (\mathbf{g}_i \oplus \mathbf{r})) \cdot \mathbf{m}_\phi, \tag{8}$$

$$\iota_\Pi = \quad (p_l(\mathbf{f}_i) * p_r(\mathbf{g}_i)) \cdot \mathbf{m}_\Pi, \tag{9}$$

And then, a dot product between items with incorrect positions and the memory trace, simulated for each of our four models using the following expressions respectively,

$$\rho_A = \quad ((\mathbf{f}_i * \mathbf{r}) + (\mathbf{g}_i * \mathbf{l})) \cdot \mathbf{m}_A, \tag{10}$$

$$\rho_\Sigma = \quad ((\mathbf{f}_i + \mathbf{r}) + (\mathbf{g}_i + \mathbf{l})) \cdot \mathbf{m}_\Sigma, \tag{11}$$

$$\rho_\phi = \quad ((\mathbf{f}_i \oplus \mathbf{r}) * (\mathbf{g}_i \oplus \mathbf{l})) \cdot \mathbf{m}_\phi, \tag{12}$$

$$\rho_\Pi = \quad (p_r(\mathbf{f}_i) * p_l(\mathbf{g}_i)) \cdot \mathbf{m}_\Pi, \tag{13}$$

For equations 8 and 12 (model ϕ), position features are first concatenated to item vectors, and then the entire vector is strictly normalized. For all other models (equations 6, 7, 9, 10, 11, and 13), all item vectors and position vectors are strictly normalized. Each equation above is computed for all L pairs in memory, which returns L samples per list. Overall order recognition sensitivity (d') is computed from these L samples across all lists according to $d' = \frac{(\mathrm{E}[\iota_M] - \mathrm{E}[\rho_M])}{\sqrt{0.5(\mathrm{Var}[\iota_M] + \mathrm{Var}[\rho_M])}}, M \in \{\mathrm{A}, \Sigma, \phi, \Pi\}$, where E and Var respectively denote the mean and variance of matching strengths.

Cued recall. As in previous models such as Plate (1995), cued recall is implemented with the correlation operation, denoted with #. If $\mathbf{f}$ and $\mathbf{m}$ are n-dimensional vectors with subscripts 0 to $k-1$, where $\mathbf{f}$ is the cue vector, and $\mathbf{m}$ is the memory vector, correlation

is defined as,

$$\mathbf{g}_i = \sum_{k=0}^{n-1} \mathbf{f}_k \mathbf{m}_{(k+i) \bmod n}, \tag{14}$$

Where $\mathbf{g}$ is an n-dimensional vector. In the following simulations, forward cued recall $\mathbf{f}_i \# \mathbf{m} \approx \mathbf{g}_i$, and backward cued recall, $\mathbf{g}_i \# \mathbf{m} \approx \mathbf{f}_i$, are simulated for all studied pairs. For each cued recall trial, a dot product is computed between the retrieved vector and all $2L$ item vectors representing possible candidate responses, which we refer to as lexicon vectors. Lexicon vectors are strictly normalized. The highest match is selected as the response with a winner-take-all rule, and is scored correct if it matches the target item. Following the procedure used by Thomas et al. (2022), where cue words had no positional information, in all models position is excluded for the cue and lexicon vectors; 1) Models A and Σ, positional vectors $\mathbf{l}$ and $\mathbf{r}$ are omitted from all cued recall operations. 2) Model ϕ, position features for cue vectors and lexicon candidate item vectors are replaced with noise, sampled from $N(0, \sigma^2)$ for each item. 3) Model Π, the cue vector and lexicon vectors are not permuted, departing from previous implementations (Kelly et al., 2013).

Associative recognition. The following two dot products are used to assess model associative recognition performance,

$$\iota = \quad (\mathbf{f}_i * \mathbf{g}_i) \cdot \mathbf{m}, \tag{15}$$

$$\rho = \quad (\mathbf{f}_i * \mathbf{g}_j) \cdot \mathbf{m} \tag{16}$$

Where $i \neq j$. Equation 15 is a dot product between the memory vector and an old (studied) pairing of list items, and equation 16 is a dot product between the memory vector and a new pairing of list items. For equation 16, this dot product is repeated for L unique new

pairings between left and right items from the studied list. All item vectors $\mathbf{f}_i$ and $\mathbf{g}_i$ are strictly normalized. Overall associative recognition performance (d') is computed using the outputs of these two dot products repeated for L pairs, across all lists, according to $d' = \frac{(\mathrm{E}[\iota_M] - \mathrm{E}[\rho_M])}{\sqrt{0.5(\mathrm{Var}[\iota_M] + \mathrm{Var}[\rho_M])}}$. Just as for cued recall, we assume no positional information in both equations; 1) Model A, and Σ, positional vectors $\mathbf{l}$ and $\mathbf{r}$ are omitted from intact and recombined probes. 2) For model ϕ, this means that position features for item vectors in intact and recombined probes are replaced with noise, sampled from $N(0, \sigma^2)$. 3) Model Π, item vectors in intact and recombined probes are not permuted.

Procedure. Following Thomas et al. (2022), encoding, cued recall, order recognition, and associative recognition are repeated for eight word lists. For recognition tasks, this results in $8L$ intact probe matches (OR: Equations 6–9, AR: Equation 15), and reverse/recombined probe matches (OR: Equations 10–13, AR: Equation 16), from which order and associative recognition sensitivity (d') is computed. Similarly, cued recall accuracy was computed across $8L$ trials for forward cued recall, and $8L$ trials for backward cued recall.

Parametric plots of model performance.

Before fits to data, we wanted to understand the sensitivity of each model to parameters, with special attention to the single parameter that directly modifies each model's ability to discriminate order. This parameter was μ in models A and Σ, n_p in model ϕ, and n_{perm} in model Π. These parameters are now called the "order parameter" of each model. We simulated cued recall, order recognition, and associative recognition at the following values of each model's order parameter; for models A and Σ, $\mu = \{0, 0.1, 0.2 ...,1.0\}$, model ϕ, $\frac{n_p}{n} = \{0, 0.1, 0.2 ...,1.0\}$, and in model Π, $\frac{n_{perm}}{n} = \{0, 0.1, 0.2 ...,1.0\}$. Simulations were repeated for $\sigma_\alpha = \{\ 0.1, 0.5\ ,1.0\}$ (SD of associative encoding strength α). Total item vector features was held constant at $n = 100$ for all simulations, and all procedures were accord-

ing to the specifications stated above. Simulations at each parameter set were iterated 100 times, and predicted values were averaged across these 100 iterations.

Results. Parameter μ in models A and Σ had a positive relationship to performance in all memory tasks (figure 3). In contrast, parameter n_{perm} in model Π had a positive relationship with order recognition performance, but a negative relationship to both associative recognition and cued recall performance. This meant that if order recognition becomes too accurate, association memory becomes unrealistically low, below levels seen in behaviour. This suggests that for model Π there may be a particular optimum, whereas for models A and Σ, performance on one task does not come at the expense of performance on another. Parameter n_p in model ϕ was similar to n_{perm} in this way, but negatively affected order recognition performance after roughly half of the item vectors consisted of position features. With a few exceptions, reducing the value of σ_α, and therefore overall noise, improved performance for all models and memory tasks. Some of these relationships between model parameters and performance could be changed if models were implemented in different ways. For example, in model Π, we did not permute the cue vector based on position, because cue words did not contain position information in Thomas et al. (2022). However, if we did permute the cue vector, all of the features of the cue vector would be diagnostic for cued recall, rather than just the $n - n_p$ non-permuted features, and there would then be a positive relationship between n_{perm} and cued recall performance.

Empirical benchmark 1a: The moderate within-subject relationship between order recognition and cued recall correctness

Thomas et al. (2022) found that order recognition performance was significantly better when cued recall for that pair was correct, but well below associative recognition for correctly recalled word pairs (Figure 4). To test if each model could account for these within-subject patterns we performed quantitative fits to means in figure 4, along with

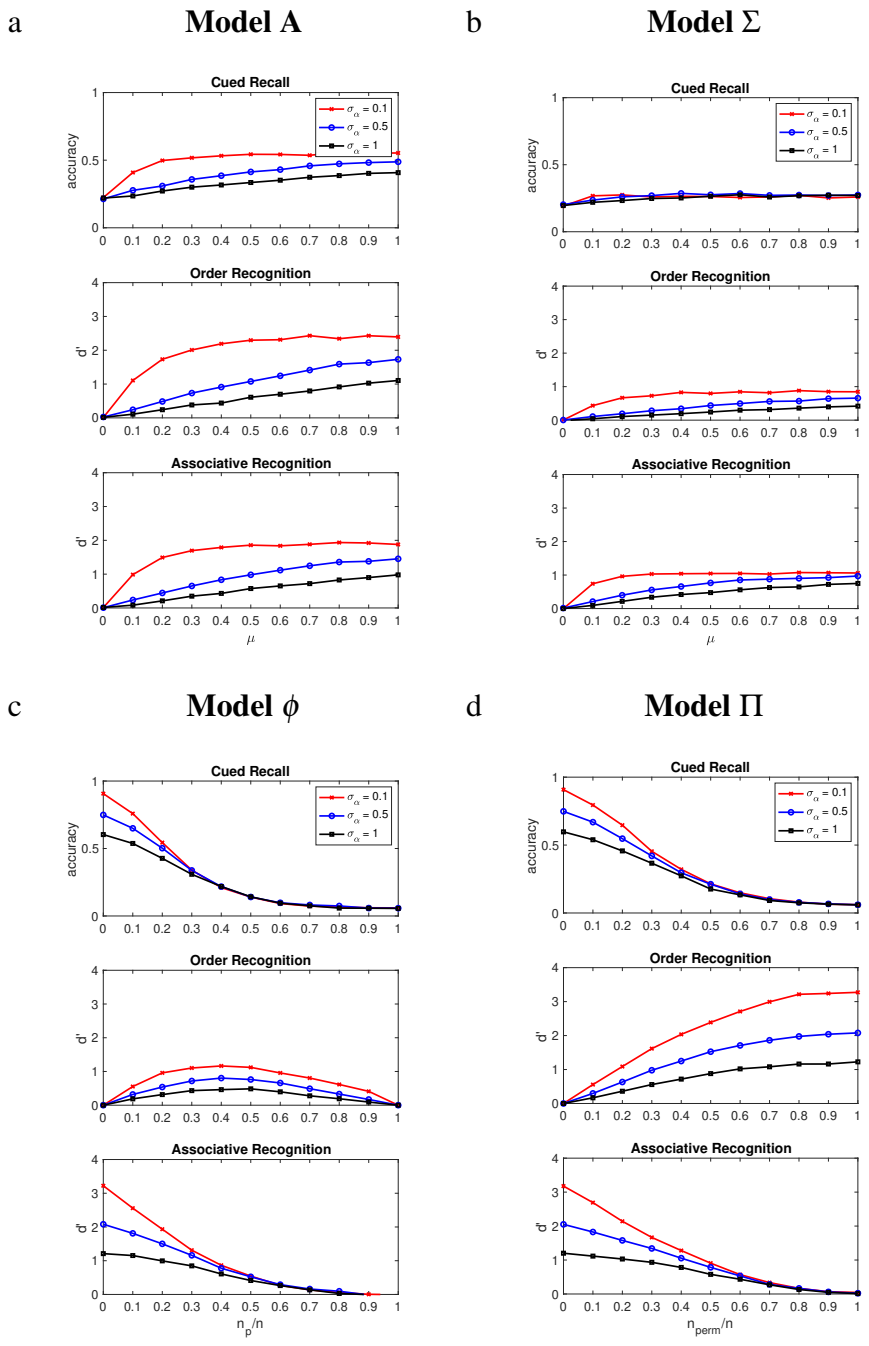


Figure 3. Parametric plots of cued recall, order recognition and associative recognition performance for each model as a function of mean associative encoding strength (μ) for models A and Σ, number of position features (n_p) for model ϕ, and the number of permuted features (n_{perm}) for model Π. Total item vector features was held constant at $n = 100$ for all simulations. Simulations were repeated for $\sigma_\alpha = \{\, 0.1, 0.5\ ,1.0\}$.

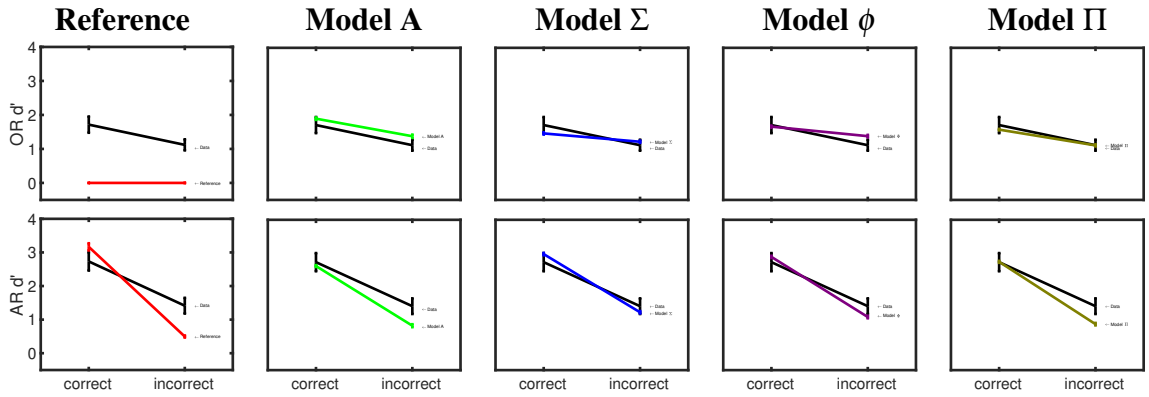


Figure 4. Benchmark 1a: Each model's best fit to order recognition d' (top row) and associative recognition d' (bottom row) for correct versus incorrectly recalled pairs from Thomas et. al. (2022). Error bars in all panels represent 95% confidence intervals based on standard error of the mean.

some other empirical constraints described below.

Given the challenge associative symmetry posed in previous efforts to modify models (see introduction), we also checked whether models maintained symmetry in the following fits. We quantitatively fit the symmetry between forward and backward cued recall accuracy using data from Thomas et al. (2022). Additionally, previous studies (e.g., Kahana, 2002; Kato & Caplan, 2017) have used Yule's Q (Bishop, Fienberg, & Holland, 1975), to measure the within-pair correlation between forward and backward cued recall. Yule's Q quantifies the relationship between two tests with dichotomous outcomes, and like a Pearson correlation, ranges from -1 to 1. Thomas et al. (2022) could not compute Yule's Q because pairs were only tested with cued recall once. However, given that high Yule's Q more diagnostic of associative symmetry than average performance (Kahana, 2002), we still checked whether each model could produce values in the basic empirical range, such as Q $\approx$ .85 in Kato and Caplan (2017).

Parameter search and methods. Each model was fit to the following empirical values from Thomas et al. (2022); 1) order recognition d' for correctly recalled pairs, 2) associative recognition d' for correctly recalled pairs, 3) the *difference* between order recog-

nition d' for correct and incorrectly recalled pairs, 4) the *difference* between associative recognition d' for correct and incorrectly recalled pairs, 5) forward cued recall accuracy, 6) backward cued recall accuracy. The empirical values for each of these measures are reported in Table 2.

The closest fit for each model was determined via direct search, meaning that each model was simulated at each combination of parameter values for the following parameters and parameter ranges; (1) $\sigma_\alpha = \{0, 0.05, 0.1 ...,1.0\}$, (2) $n = \{10, 20, 30 ...,500\}$. (3) Again, the third free parameter was known as the order parameter, and was specific to each model. For models A and Σ, μ (mean of associative encoding strength)= $\{0, 0.025, 0.05 ...,1.0\}$, model ϕ, number of positional features, $\frac{n_p}{n} = \{0, 0.025, 0.05 ...,1.0\}$, and in model Π, the number of permuted features, $\frac{n_{perm}}{n} = \{0, 0.025, 0.05 ...,1.0\}$.[8] This resulted in a $41\times50\times21$ matrix of model predictions for each model which we call the "direct search matrix".

Forward cued recall, backward cued recall, order recognition and associative recognition were simulated as described above, for 8 lists of L pairs. These simulations were iterated 300 times for each cell of the direct search matrix, and model predicted values were averaged across these 300 iterations. Root-Mean-Squared Error (RMSE) was computed between empirical and model predicted values for the four means plotted in Figure 4. RMSE was then transformed to Bayesian Information Criterion (BIC) values via an estimation of log-likelihood (Burnham & Anderson, 2004). The BIC minimum was selected from the direct search matrix to find the best fitting parameter set. By convention, if ΔBIC > 2 the models are considered meaningfully different. We have included heat maps of BIC values around the best fitting parameter sets in supplementary materials (Figure S1 and S2).

To compute model predictions for Yule's Q, we used the outcome of cued recall simu-

[8] Note, that in model ϕ and Π, certain values of the order parameters $\frac{n_{perm}}{n}$ and $\frac{n_p}{n}$ resulted in decimal values of permuted or positional features (e.g., $\frac{n_{perm}}{n} = 0.325$ at $n = 130$ would result in 42.25 permuted features). In this case, the number of permuted or positional features was rounded to the nearest whole number.

lations for each pair in the backward and forward direction. Predicted Yule's Q values were generated for each of the 300 iterations at each cell of the direct search matrix as follows. The frequency of the following four outcomes was tallied; a = # of pairs where forward and backward cued recall were correct, b = # of pairs where forward cued recall was correct and backward cued recall was incorrect, c = # of pairs where forward cued recall was incorrect and backward cued recall was correct, d = # of pairs where both backward and forward cued recall were incorrect. Yule's Q is then calculated according to $(ad-bc)/(ad+bc)$, and can range from -1 to 1. We added 0.5 observations to each outcome (a,b,c,d) to prevent infinities. Yule's Q values for each cell were then log-odds transformed, averaged, and then inverse log-odds-transformed[9] to generate a single predicted value at each cell of the direct search matrix.

For comparison we also plotted and reported the performance of a reference model by simulating 300 iterations of the Model Π at $\frac{n_{perm}}{n} = 0$, $\sigma_\alpha = 0.5$, and $n = 400$. At these parameter values this model is equivalent to an unmodified convolution model with no information for item position/order. This model is unable to produce order recognition d' above 0, and would thus be unconstrained by empirical order recognition performance, unlike the other models. As a result, we did not fit the reference model to data.

Results. All four modified convolution models improved substantially on the reference model fits (Table 1), and Model A (item-position associations), and Model Π (position-specific permutation) performed substantially better than model Σ (both $\Delta BIC > 2$). For all other differences $\Delta BIC < 2$. Although there were differences in quantitative fits, in general, models could produce greater order recognition performance for correctly recalled pairs compared to incorrectly recalled pairs, indicating a moderate relationship between order memory and association memory. Models Σ and ϕ produced a smaller dif-

[9]This followed analyses of empirical Yule's Q in Kato and Caplan (2017), who log-odds transformed Yule's Q to ensure that these measures met parametric assumptions.

ference than other models and what is seen in behaviour (Table 2); however, because the current fits were highly constrained, and required models to fit both order and associative recognition with the same parameters (despite being performed by different participants in the behavioural data-set), this does not necessarily indicate that models Σ and ϕ cannot produce a moderate order recognition-cued recall relationship, and would perhaps produce closer fits to behavioural values under different fitness measures and parameters spaces.

All models could produce order recognition performance that was well below associative recognition for correctly recalled pairs, successfully producing the less than maximum relationship between order and association memory seen in behaviour (Figure 4).

All models were also successful at preserving associative symmetry. Additionally, all models exhibited equal forward and backward cued recall accuracy, with values that were close to empirical observations. There was a marked reduction in Yule's Q for all models compared to the reference model (Table 2), closer to what is observed empirically (Yule's Q > 0.85 for all groups in Kato & Caplan, 2017). Previous methods to reduce model Yule's Q in models such as adding noise between successive tests simultaneously reduced test-retest correlations (forward-forward, backward-backward) which, in contrast, tend to be close to 1 in behavioural data.

In sum, all modifications extended convolution to support moderate order recognition, without compromising associative symmetry and indeed producing Yule's Q values that closer to empirical observations.

Empirical benchmark 1b: Fits to individual differences in order recognition performance

Fits to aggregate data can be informative, but can lead to misleading conclusions if participants vary substantially, where some participants are better fit by one model and others, by a different model. Indeed, even though order recognition performance exhibits a

Table 1

Best-fitting model parameters for fits to benchmark 1a. Data was obtained from experiment 1 in Thomas et. al. (2022). All models produced substantially closer fits compared to the reference model (Δ BIC>2). Additionally, model A and model Π performed substantially better than models Σ. All other differences in BIC values were not greater than 2. For all models σ_α and n were free parameters. The order parameter was a free parameter unique to each model—for model ϕ this was n_p, for model Π this was n_{perm}, and for models A and Σ this was μ. Note, for models ϕ and Π, parameter μ was held constant.

Model	n	σ_α	Order parameter	BIC
Reference	400	0.5	N/A	5.07
Model A	60	0.4	$\mu = 0.925$	–13.05
Model Σ	400	0.2	$\mu = 0.775$	–9.77
Model ϕ	380	0.3	$\frac{n_p}{n} = 0.375$	–11.25
Model Π	130	0.45	$\frac{n_{perm}}{n} = 0.325$	–11.85

Table 2

Data from experiment 1 in Thomas et. al. (2022) along with predictions generated by each model at best fitting parameters. Values under correct and incorrect are data and model predictions for recognition performance for correctly and incorrectly recalled respectively. Under the label difference are data and model predictions for recognition performance for correctly recalled pairs minus performance for incorrectly recalled pairs, and provides a measure of the dependence of recognition on cued recall performance. We also report model predicted values for Yule's Q, although we did not quantitatively fit models to empirical Yule's Q values.

	Order recognition d'			Associative recognition d'			Cued recall accuracy		
	correct	incorrect	difference	correct	incorrect	difference	forward	backward	Yule's Q
Data	1.71	1.12	0.59	2.73	1.41	1.32	0.44	0.40	-
Reference	0	0	0	3.18	0.40	2.78	0.91	0.91	.99
Model A	1.90	1.38	0.52	2.62	0.83	1.79	0.38	0.38	.85
Model Σ	1.47	1.22	0.25	2.97	1.23	1.74	0.59	0.59	.85
Model ϕ	1.67	1.39	0.28	2.89	1.09	1.8	0.51	0.51	.87
Model Π	1.58	1.11	0.47	2.74	0.88	1.86	0.46	0.46	.87

moderate relationship to cued recall, individual participants in Thomas et al. (2022) occupied a range around these mean values (See Figure S5 in Supplementary Materials). Thus, we tested how each model could fit individual differences.[10]

Parameter search. We fit models to individual participant values for; 1) order recognition d' for correctly recalled pairs, 2) order recognition d' for incorrectly recalled pairs, 3) log-odds transformed cued recall accuracy.[11] Re-using simulated model predictions from the direct search matrix for benchmark 1a, best fits were selected by minimizing BIC for each participant.[12]

Results. Figure 5 plots residuals for fits to individual participant's order recognition performance for incorrectly recalled pairs (x axis) and correctly recalled pairs (y axis). Model Π exhibited a tight, spherical scatter around the origin, indicating highly accurate fits to individual participants on both measures. There were, however, two clear outliers for model Π, indicating that certain participants were more challenging for this model to account for. Figure S4 shows that these two participants had exceptionally high OR d' for correctly recalled pairs. Next, residuals for models A and ϕ exhibited a larger and less spherical scatter compared to model Π, although there were no clear outliers. Finally, model Σ had the largest scatter, with a clearly non-spherical pattern, suggesting that this model provides a weak account of individual participants relative to other models.

Additional insights can be garnered from Figure S4 as follows—Model Σ could only produce a narrow range of performance values and was the poorest at accounting for individual differences across all models. In contrast, models A and ϕ produced a range of predictions, although model A was biased towards predicting high order recognition d' for

[10] We also examined whether models could produce empirical *between-subject* correlations between recognition and cued recall performance (page S6)

[11] Log-odds transformed cued recall was included in the fitness measure because we also used the following model fits for benchmark 1c (see below).

[12] The distribution of parameter values that each model used to fit individual participants is reported in Table S1 and plotted in Figure S3.

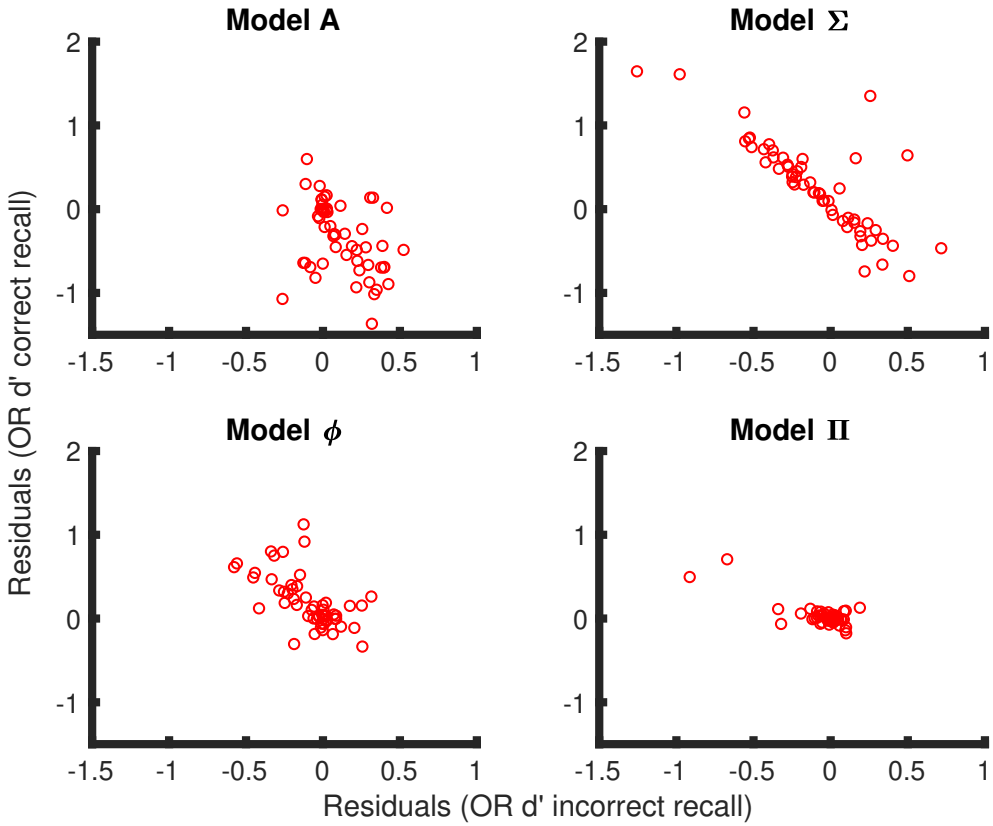


Figure 5. Residuals for fits to individual participant data from Thomas et. al. (2022). Coordinates for each circle are the difference between data and model predictions for an individual participant's order recognition d' for incorrectly recalled pairs (x axis), versus correctly recalled pairs (y axis).

correctly recalled pairs (relative to the central tendency of the empirical data), while model ϕ tended to predict values closer to the center or even on the lower end of empirical order recognition d' for correctly recalled pairs. Finally, model Π produced the broadest range of predictions, suggesting it was the most flexible out of all models.

As an additional way to compare model fits, we used a winner-take-all rule, tallying the number of times each model produced the lowest BIC value for a given participant. If a model did not win by significant margin compared to the other three models ($\Delta\text{BIC} > 2$) we omitted that participant from reported counts. 37 participants were excluded on this basis. Figure S5 plots this analysis.

Model Π provided the strongest account of benchmark 1b, producing the best fit to

15 participants which were located throughout the scatter (Figure S5). This was followed by model ϕ which provided the best fit to four participants. Finally, models A and Σ did not win for any participant. This may indicate that, in addition to providing a good account of mean order recognition performance, model Π provided the closest fit to the largest number of individuals. However, given that certain participants were better described by other models, and that 37 participants did not have a winning model, this suggests that participants may, in fact, judge order in more than one way.

Empirical benchmark 2: double-function lists

Although models varied in their ability to account for individual differences, we have shown that simple modifications to convolution can produce moderate order memory without compromising its inherent symmetry. As a further test of each model, we leveraged another paradigm that demands memory for constituent-order, double function lists (AB..., BC..., CA...). Recall that standard convolution models are unable to disambiguate double function pairs during cued recall (see introduction). For example, if A is presented as a cue to a convolution model, both B and C are retrieved equally, and the model must guess. We start with algebraic expressions to come to general conclusions about how each model may solve this task, and then test these conclusions with simulations.

Model A. First assume that three double-function pairs are encoded in memory, AB, BC, and CA. Following Rehani and Caplan (2011), each item appears in two pairs, exactly once in the left position and exactly once in the right position. Dropping the associative encoding strength (α_i), this is expressed in model A as follows,

$$\begin{aligned}\mathbf{m} = \mathbf{a}*\mathbf{b} + \mathbf{a}*\mathbf{l} + \mathbf{b}*\mathbf{r} \\ +\mathbf{b}*\mathbf{c} + \mathbf{b}*\mathbf{l} + \mathbf{c}*\mathbf{r} \\ +\mathbf{c}*\mathbf{a} + \mathbf{c}*\mathbf{l} + \mathbf{a}*\mathbf{r}\end{aligned} \tag{17}$$

where **a**, **b**, and **c** denote item vectors, and **l** and **r** denote left and right position vectors respectively. Cued recall is expressed as follows,

$$\mathbf{a} \# \mathbf{m} = \mathbf{b} + \mathbf{c} + \mathbf{l} + \mathbf{r} \tag{18}$$

We see that the retrieved vector is essentially a sum of **b** and **c** with noise. As a result, there is no information to help the model select between competing items, resulting in perfect double function interference. To address this, one could incorporate the positional vector into the cue,

$$(\mathbf{a} + \mathbf{l}) \# \mathbf{m} = 2\mathbf{b} + 2\mathbf{c} + \mathbf{a} + \mathbf{l} + \mathbf{r} \tag{19}$$

However, the retrieved vector is still equally similar to **c** and **b**. This is because the positional vector **l** is associated to every item exactly once in the list, and provides no information to solve double function interference.

Model Σ. Model Σ cannot solve double function interference for the same reason. Again, assume that pairs AB, BC, and CA are encoded in memory,

$$\begin{aligned} \mathbf{m} = &(\mathbf{a} + \mathbf{l}) * (\mathbf{b} + \mathbf{r}) \\ &+ (\mathbf{b} + \mathbf{l}) * (\mathbf{c} + \mathbf{r}) \\ &+ (\mathbf{c} + \mathbf{l}) * (\mathbf{a} + \mathbf{r}) \end{aligned} \tag{20}$$

Expanding the above expression shows that model Σ is equivalent to model A, with an additional noise term $(\mathbf{l} * \mathbf{r})$ generated for each pair,

$$\begin{aligned} \mathbf{m} = &(\mathbf{a} * \mathbf{b}) + (\mathbf{a} * \mathbf{r}) + (\mathbf{b} * \mathbf{l}) + (\mathbf{l} * \mathbf{r}) \\ &+ (\mathbf{b} * \mathbf{c}) + (\mathbf{b} * \mathbf{r}) + (\mathbf{c} * \mathbf{l}) + (\mathbf{l} * \mathbf{r}) \\ &+ (\mathbf{c} * \mathbf{a}) + (\mathbf{c} * \mathbf{r}) + (\mathbf{a} * \mathbf{l}) + (\mathbf{l} * \mathbf{r}) \end{aligned} \tag{21}$$

In its expanded form, we can see that both positional vectors are associated to every item

in the list. As a result, if cued recall is simulated with the cue $\mathbf{a}+\mathbf{l}$,

$$(\mathbf{a}+\mathbf{l})\#\mathbf{m} = 2\mathbf{b}+2\mathbf{c}+\mathbf{a}+\mathbf{l}+4\mathbf{r} \tag{22}$$

the retrieved vector is equally similar to the target item **b**, and non-target item **c**. Just as in model A, positional vector **l** provides no diagnostic ability.

Model ϕ. Position features also cannot be used to solve double function interference. AB, BC, and CA would be encoded as follows,

$$\mathbf{m} = (\mathbf{a}\oplus\mathbf{l}*\mathbf{b}\oplus\mathbf{r})+(\mathbf{b}\oplus\mathbf{l}*\mathbf{c}\oplus\mathbf{r})+(\mathbf{c}\oplus\mathbf{l}*\mathbf{a}\oplus\mathbf{r}) \tag{23}$$

Assuming $n = 3$ and $n_p = 1$, this can also be expressed in its expanded form,

$$\begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix} = \left(\begin{bmatrix} a_1 \\ a_2 \\ l_3 \end{bmatrix} * \begin{bmatrix} b_1 \\ b_2 \\ r_3 \end{bmatrix}\right) + \left(\begin{bmatrix} b_1 \\ b_2 \\ l_3 \end{bmatrix} * \begin{bmatrix} c_1 \\ c_2 \\ r_3 \end{bmatrix}\right) + \left(\begin{bmatrix} c_1 \\ c_2 \\ l_3 \end{bmatrix} * \begin{bmatrix} a_1 \\ a_2 \\ r_3 \end{bmatrix}\right) \tag{24}$$

$$\begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix} = \begin{bmatrix} a_1b_1+a_2r_3+l_3b_2+ & b_1c_1+b_2r_3+l_3c_2+ & c_1a_1+c_2r_3+l_3a_2 \\ a_1b_2+a_2b_1+l_3r_3+ & b_1c_2+b_2c_1+l_3r_3+ & c_1a_2+c_2a_1+l_3r_3 \\ a_1r_3+a_2b_2+l_3b_1+ & b_1r_3+b_2c_2+l_3c_1+ & c_1r_3+c_2a_2+l_3a_1 \end{bmatrix} \tag{25}$$

We can see that positional features l_3 and r_3 are distributed throughout the memory vector after convolution, appearing in terms with item features from every item of the list. Ultimately, this means that positional features are no longer specific to any item. This becomes clearer if we proceed with cued recall, which is expressed as, $\mathbf{x} = (\mathbf{a}\oplus\mathbf{l})\#\mathbf{m}$, where **x** is the retrieved vector. The expanded form of equation is expressed as follows,

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} a_1 \\ a_2 \\ l_3 \end{bmatrix} \# \begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix} \tag{26}$$

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} a_1 m_1 + a_2 m_2 + l_3 m_3 \\ l_3 m_1 + a_1 m_2 + a_2 m_3 \\ a_2 m_1 + l_3 m_2 + a_1 m_3 \end{bmatrix} \tag{27}$$

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1(a_1^2 + a_2^2 + l_3^2) + c_1(a_1^2 + a_2^2 + l_3^2) + a_1 l_3^2 + noise \\ b_2(a_1^2 + a_2^2 + l_3^2) + c_2(a_1^2 + a_2^2 + l_3^2) + a_2 l_3^2 + noise \\ r_3(a_1^2 + a_2^2 + l_3^2) + r_3(a_1^2 + a_2^2 + l_3^2) + r_3 l_3^2 + noise \end{bmatrix} \tag{28}$$

The retrieved vector $\mathbf{x}$ is essentially an equal sum of $\mathbf{b} \oplus \mathbf{r}$ and $\mathbf{c} \oplus \mathbf{r}$, and to a lesser extent $\mathbf{a} \oplus \mathbf{r}$. As a result, dot products to both candidate items will be equal ($\mathrm{E}[\mathbf{x} \cdot (\mathbf{b} \oplus \mathbf{r})] = \mathrm{E}[\mathbf{x} \cdot (\mathbf{c} \oplus \mathbf{r})]$), regardless of the number of positional features n_p. Thus, because position features are repeated for multiple items, they cannot be used to cue a specific item in memory. Thus, the position-feature model cannot solve double function interference.

Model Π. Permutation, in contrast, can be used solve double function interference. First assume that pairs AB, BC, and CA are encoded as follows,

$$\mathbf{m} = p_l(\mathbf{a}) * p_r(\mathbf{b}) + p_l(\mathbf{b}) * p_r(\mathbf{c}) + p_l(\mathbf{c}) * p_r(\mathbf{a}) \tag{29}$$

To understand why interfering pairs can be disambiguated with permutation, consider a case where the whole item is permuted ($\frac{n_{perm}}{n} = 1$) before encoding. Given this, $p_r(\mathbf{a})$ and

$p_l(\mathbf{a})$ will behave as distinct, orthogonal items (assuming large n). As a result, if cued recall proceeds with the following expression,

$$p_l(\mathbf{a}) \# \mathbf{m} \tag{30}$$

$p_l(\mathbf{a})$ will only evoke pair, $p_l(\mathbf{a}) * p_r(\mathbf{b})$in memory, and grant the model perfect ability to disambiguate double function pairs. If we assume only a subset of item vectors are permuted ($\frac{n_{perm}}{n} < 1$), The degree to which vector $p_l(\mathbf{a})$ retrieves the target $p_r(\mathbf{b})$ is proportional to n_{perm}. This is because the non-permuted portion of $p_l(\mathbf{a})$ is identical to the non-permuted portion of $p_r(\mathbf{a})$, it will also evoke pair $p_l(\mathbf{c}) * p_r(\mathbf{a})$. As we demonstrate below, changing n_{perm} allows the position-specific permutation model to produce a range of performance values ranging from zero (like an unmodified convolution model) to perfect (like a matrix model) ability to solve double function pairs.

Simulation methods

To test the insights gained from algebraic expressions, we also simulated double function lists with each of our four models.

Encoding. Assume that each model stores double function pairs AB, BC, and CA. Encoding for each model proceeds as follows,

$$\mathbf{m}_A = \alpha_1(\mathbf{a}*\mathbf{b}+\mathbf{a}*\mathbf{l}+\mathbf{b}*\mathbf{r}) + \alpha_2(\mathbf{b}*\mathbf{c}+\mathbf{b}*\mathbf{l}+\mathbf{c}*\mathbf{r}) + \alpha_3(\mathbf{c}*\mathbf{a}+\mathbf{c}*\mathbf{l}+\mathbf{a}*\mathbf{r}) \tag{31}$$

$$\mathbf{m}_\Sigma = \alpha_1((\mathbf{a}+\mathbf{l})*(\mathbf{b}+\mathbf{r})) + \alpha_2((\mathbf{b}+\mathbf{l})*(\mathbf{c}+\mathbf{r})) + \alpha_3((\mathbf{c}+\mathbf{l})*(\mathbf{a}+\mathbf{r})) \tag{32}$$

$$\mathbf{m}_\phi = \alpha_1(\mathbf{a}\oplus\mathbf{l}*\mathbf{b}\oplus\mathbf{r}) + \alpha_2(\mathbf{b}\oplus\mathbf{l}*\mathbf{c}\oplus\mathbf{r}) + \alpha_3(\mathbf{c}\oplus\mathbf{l}*\mathbf{a}\oplus\mathbf{r}) \tag{33}$$

$$\mathbf{m}_\Pi = \alpha_1(p_l(\mathbf{a})*p_r(\mathbf{b})) + \alpha_2(p_l(\mathbf{b})*p_r(\mathbf{c})) + \alpha_3(p_l(\mathbf{c})*p_r(\mathbf{a})) \tag{34}$$

where **a**, **b**, and **c** are word vectors with n features, **l**, **r** represent positional vectors in models A and Σ with n features, and α_1, α_2, and α_3 represent associative encoding strengths.

Cued recall. Assuming that A is a left-position cue, cued recall proceeds as follows,

$$(\mathbf{a}+\mathbf{l})\#\mathbf{m}_A \tag{35}$$

$$(\mathbf{a}+\mathbf{l})\#\mathbf{m}_\Sigma \tag{36}$$

$$(\mathbf{a}\oplus\mathbf{l})\#\mathbf{m}_\phi \tag{37}$$

$$p_l(\mathbf{a})\#\mathbf{m}_\Pi \tag{38}$$

Then for all models, a dot product is computed between the retrieved vector, and each of the vectors, **b**, and **c**, which represent candidate items B and C. Note that for model Π, the output of equation 37 is permuted with the inverse of the right permutation pattern to reproduce the original non-permuted item, following previous implementations of permutation (Jones & Mewhort, 2007; Kelly et al., 2013). If a model can disambiguate double function pairs, the retrieved item should be more similar to item B than to item C.

Procedure. For item vectors, **a**, **b**, and **c**, and position vectors **l**, and **r**, $n = 100$. Vector features were drawn from $N(0,\sigma^2)$, where $\sigma^2 = \frac{1}{n}$. Associative encoding strengths (α_1,α_2,α_3) were drawn from $N(\mu,\sigma_\alpha)$, where $\sigma_\alpha = 1$, and $\mu = 1$ for models ϕ and Π. We varied the number of item position features (n_p) in model ϕ, permuted features (n_{perm}) in model Π, and mean associative encoding strength (μ) in models A and Σ according to the following ranges $\frac{n_p}{n} = \{0, 0.1, 0.2 ...,1.0\}$, $\frac{n_{perm}}{n} = \{0, 0.1, 0.2 ...,1.0\}$, $\mu = \{0, 0.1, 0.2 ...,1.0\}$. For each model, dot products between the retrieved vectors from equations 35-38, and candidate items **b** and **c** were averaged across 10000 iterations, for each value of n_p, n_{perm}, and μ.

Results

The main results from these simulations are plotted in figure 6. Confirming our arguments above, models ϕ, A, and Σ, could not solve interference between **b** and **c**, even when parameters $\frac{n_p}{n}$ and μ were increased. In contrast, for model Π the difference in matching strengths between the retrieved vector to both **b** and **c** increased with parameter $\frac{n_{perm}}{n}$. At $\frac{n_{perm}}{n} = 1$, this difference reached the maximum possible value, where the matching strength to **c** reached the minimum dot product between two normalized vectors (≈ 0). This indicates that model Π is able to mimic both zero, and perfect ability to solve double function interference, and all values in between. Taken together, this confirms the idea that models ϕ, A, and Σ suffer from cue-overload when tested with cued recall for stored double function pairs. Permutation (model Π) overcame this challenge because permuting a given item by two different patterns (e.g., $p_l(\mathbf{a})$ versus $p_r(\mathbf{a})$) decreased similarity between both versions, in proportion to the number of permuted features. This meant that a cue vector with a certain positional permutation predominantly activated a single pair, overcoming interference from the other pair containing that cue item.

Relating these simulations back to previous model fits, when we fit models to averaged order recognition data (benchmark 1a), model Π achieved its best fit at $\frac{n_{perm}}{n} = 0.325$. The distributions of matching strengths at $\frac{n_{perm}}{n} = 0.3$, plotted in Figure 7, show a clear separation between the means of the target item and non-target item matching strength distributions that indicates some ability to disambiguate double function pairs, but also an overlap between the distributions that is consistent with high rates of errors observed to the non-target item in behaviour (Rehani & Caplan, 2011). In other words, model Π may not need to deviate from the fits to the other benchmark data to be able to perform well on double function lists.

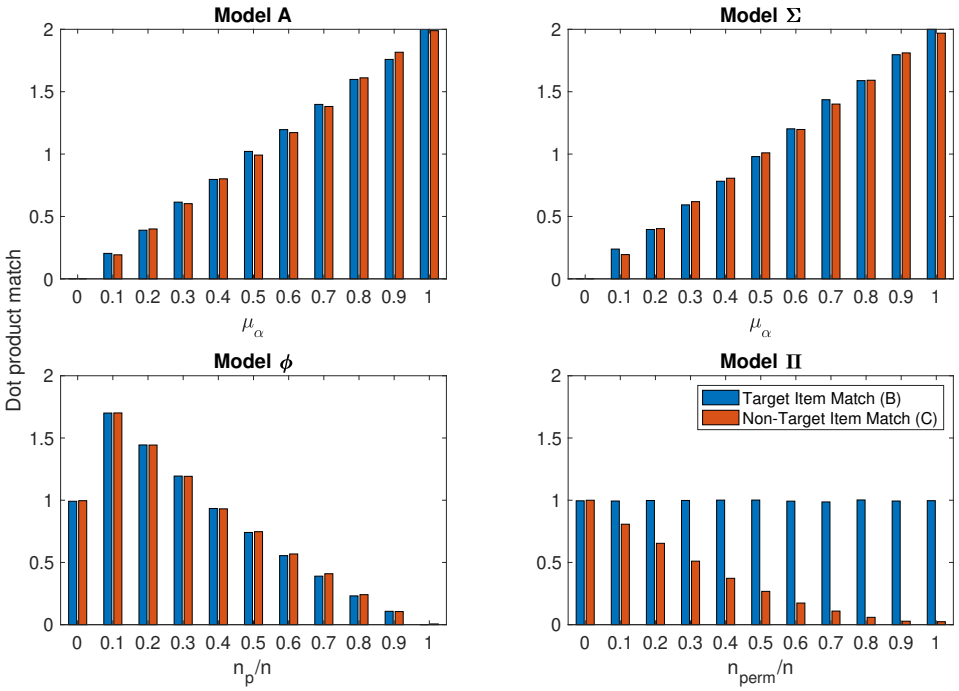


Figure 6. Double function list simulations: For each model, dot products between the retrieved vectors from equations 35-38, and candidate items **b** and **c** were averaged across 10000 iterations, for each value of n_p, n_{perm}, and μ. For models ϕ, A, and Σ matching strengths are identical for the target and non-target at all parameter values. For model Π, the difference between the target and non-target item match, and therefore the ability to solve interference, increases with the proportion of permuted features. At $n_{perm}/n = 0$, model Π is equivalent to an unmodified convolution model and has no ability to solve this interference. At $n_{perm}/n = 1$, model Π is essentially non-commutative like in previous implementations of permutation (e.g., Kelly et. al., 2013), and has perfect ability to solve interference.

Discussion

We started with the following puzzle: the perfect symmetry of convolution-based models matched behavioural data well, but offered no ability to discriminate the constituent-order of associations. As a result, convolution models could not account for more recent empirical data, which revealed that the constituent order of an association could be judged above-chance, and that this ability was moderately dependent on remembering the pairing itself. To address this challenge we designed and evaluated four extensions of convolution. Despite being extremely simple, consisting of only three free parameters each,

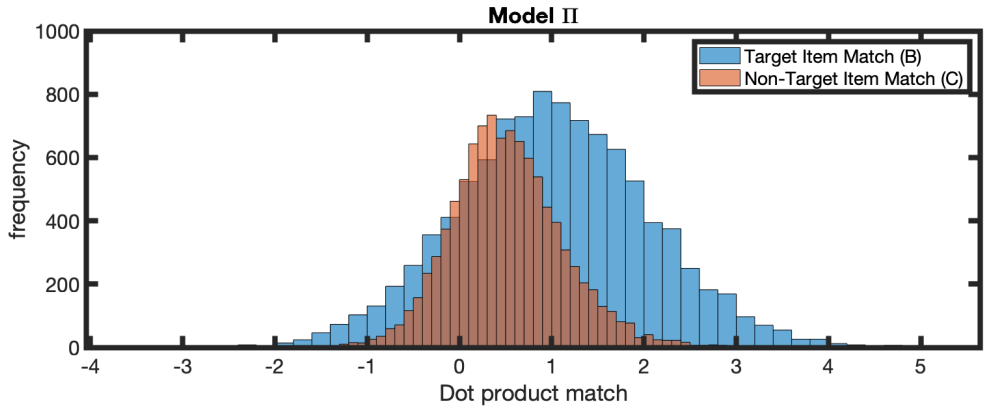


Figure 7. Double function list simulations: Distributions of dot products computed between a retrieved vector (from cued recall), and the target versus non-target item for model Π at $n_{perm}/n = 0.3$. At this proportion of permuted features, model Π provided its best fit to the previous benchmark 1a.

all models provided reasonable accounts of order recognition data without compromising the inherent symmetry of convolution. Our second benchmark, double function lists, could only be accounted for by model Π. While it may be tempting to conclude that model Π won on this basis, we emphasize here and below that our purpose was to build an understanding of how certain models might fit or miss certain aspects of behaviour, rather than conclusively rule out certain models. Permutation showed its strengths in conditions involving order-related interference during cued recall, but we also discuss conditions where other mechanisms may be favored below.

Simple modifications can produce memory for order with symmetric associations. We tested several possible mechanisms that could extend a convolution model to store order. All models provided a reasonable fit to order recognition and associative recognition data, although the item-position association model (model A) performed the best quantitatively. Position-specific, partial permutation (model Π) produced its best fits to data at only 32.5% permuted features, departing from previous implementations (e.g., Jones & Mewhort, 2007; Kelly et al., 2013). This suggests convolution can be modified quite easily to produce moderate order recognition performance.

We also checked whether each model could preserve the inherent symmetry of convolution while fitting recognition data. This was especially important consider given the difficulty this additional constraint posed in previous efforts to modify matrix models. Matrix models start out asymmetric, but can be modified to produce associative symmetry by storing both the forward and backward associations (and with highly correlated forward and backward associative encoding strengths), although this removes any information for order. To regain some order, one could increase the forward association strength, but this causes the model to violate associative symmetry, along with generating additional erroneous predictions (see introduction). In contrast, all four of our models here maintained symmetry between forward and backward cued recall accuracy, although not perfect, nor as high as values from unmodified convolution model (Table 2). Interestingly, less-than-perfect forward-backward Yule's Q is, in fact more consistent with empirical findings. In data, the test/re-test correlation (both tests forward or both backward) is typically nearly perfect, whereas the forward-backward correlation is typically well below 1, around 0.8–0.9 (Kahana, 2002; Kato & Caplan, 2017; Rehani & Caplan, 2011; Rizzuto & Kahana, 2000, 2001; Sommer et al., 2008). In a symmetric model, one way to reduce the forward-backward correlation would be add noise between successive tests; however, this would also reduce the test/re-test correlation. In contrast, all four of our models produced correlations well below 1 without such a mechanism. Additionally, because we did not incorporate testing effects into our models, test/re-test correlations for all models are trivially equal to 1. Thus, it seems that deviating from the perfectly commutative convolution operation can also explain why forward and backward cued recall are slightly decoupled from one another (compared to testing twice in the same direction), without losing other desirable characteristics of convolution, such as equal accuracy in the forward and backward direction on average.

The success of our models shows that symmetric item-item associations can still

support order judgements. Furthermore, the paradox between associative symmetry and moderate order memory may be particular to an unmodified matrix model, which assumes order is derived directly from a perfectly directional association. However, these results do not necessarily argue against matrix models, but suggest modifications to these models need to take a different approach. For example, one could incorporate partial-permutation into a symmetric matrix model as follows, $M = \alpha(p_l(\mathbf{a})p_r(\mathbf{b})^\intercal + p_r(\mathbf{b})p_l(\mathbf{a})^\intercal)$, where the forward and backward association share the same associative encoding strength α to produce high Yule's Q. The model could infer order by retrieving an item, then computing a dot product to a copy of this item with the correct position, $p_r(\mathbf{b}) \cdot (Mp_l(\mathbf{a}))$, and incorrect position, $p_l(\mathbf{b}) \cdot (Mp_l(\mathbf{a}))$. Order recognition performance, as the difference between these two dot products, would be proportional to number of permuted features. This version of the matrix model may be able to function similarly to its cousin implemented with convolution (model Π).

Like convolution, some recent models within the REM framework such as Criss and Shiffrin (2005) and Cox and Criss (2017, 2020) also disregard the order within associations. The design principles of models Π and ϕ could also be applied to these models quite easily. Indeed, Cox and Criss (2020) suggested something to this effect, where features representing the spatial locations of each item could be incorporated into item vectors in their model to produce some memory for order. Partial permutation could be applied to REM-based models with the same logic as with matrix models (described above). However, because item-item associations are represented with concatenation within the REM framework, the implementation of model A would be formally equivalent to the implementation of model ϕ.

Modelling other asymmetric relationships. An interesting future application of each of our models here may be asymmetric relationships beyond memory for the constituent-order of random word pairs. For example, adjective-verb relationships,

modifier-head relationships within compound words (GUEST HOUSE), and spatial relationships (above-below, front-behind) are asymmetric, where individual items have unique identities in relation to each other. Convolution augmented with any of our examined mechanisms (e.g., partial-permutation, item-position associations) may also provide alternative accounts of this type of information, which could be explored in future work.

The influence of order/position on associative recognition. Across six experiments and under various conditions, Yang et al. (2013) found that associative recognition probes were judged faster, and with higher accuracy when presented in the correct order, consistent with results from previous studies (Haskins, Yonelinas, Quamme, & Ranganath, 2008; Wiegand, Bader, & Mecklinger, 2010). Our present models may help us understand how these results are still consistent with symmetric associations in memory. First, consider the position-specific permutation model (model Π). Assume the model stores the following pairs in memory, $\mathbf{m} = p_l(\mathbf{a}) * p_r(\mathbf{b}) + p_l(\mathbf{c}) * p_r(\mathbf{d})$. If the model "knows" that probes may be reversed at test, it is reasonable to assume that it will apply permutation to probe items to incorporate order into the recognition process. The model could implement a "forward" intact trial as follows, $p_l(\mathbf{a}) * p_r(\mathbf{b}) \cdot \mathbf{m}$, along with a "backward" intact trial, $p_r(\mathbf{a}) * p_l(\mathbf{b}) \cdot \mathbf{m}$. These two matches are identical to our implementation of order recognition in model Π (equations 9 and 13), so we already know that the model can produce an advantage for forward intact probes. The model could not produce a forward advantage for recombined probes, because both $\mathrm{E}[p_l(\mathbf{a}) * p_r(\mathbf{d}) \cdot \mathbf{m}]$ and $\mathrm{E}[p_r(\mathbf{a}) * p_l(\mathbf{d}) \cdot \mathbf{m}]$ are equal to 0. Thus, the permutation model would predict that forward asymmetries for associative recognition are only driven by asymmetries in intact probe trials.

Model ϕ (position-features) would function similarly. Again, we know the model can produce a correct-order advantage for intact pairs because the comparison between a forward and backward intact trial is identical to its implementation of order recognition (Equation 8 and 12), thus $\mathrm{E}[(\mathbf{a} \oplus \mathbf{l}) * (\mathbf{b} \oplus \mathbf{r}) \cdot \mathbf{m}] > \mathrm{E}[(\mathbf{a} \oplus \mathbf{r}) * (\mathbf{b} \oplus \mathbf{l}) \cdot \mathbf{m}]$. However, be-

cause $\mathrm{E}[(\mathbf{a} \oplus \mathbf{l}) * (\mathbf{d} \oplus \mathbf{r}) \cdot \mathbf{m}]$ and $\mathrm{E}[(\mathbf{a} \oplus \mathbf{r}) * (\mathbf{d} \oplus \mathbf{l}) \cdot \mathbf{m}]$ are both equal to 0, like in model Π, this model predicts no order-advantage for recombined pairs.

Model A (the item-position association model), and by extension model Σ, will produce an order-advantage for associative recognition. Given that encoding is as follows, $\mathbf{m} = (\mathbf{a} * \mathbf{b} + \mathbf{a} * \mathbf{l} + \mathbf{b} * \mathbf{r}) + (\mathbf{c} * \mathbf{d} + \mathbf{c} * \mathbf{l} + \mathbf{d} * \mathbf{r})$, the model produces a correct-order advantage to intact probes because, $\mathrm{E}[(\mathbf{a} * \mathbf{b} + \mathbf{a} * \mathbf{l} + \mathbf{b} * \mathbf{r}) \cdot \mathbf{m}] > \mathrm{E}[(\mathbf{a} * \mathbf{b} + \mathbf{a} * \mathbf{r} + \mathbf{b} * \mathbf{l}) \cdot \mathbf{m}]$. However, unlike model Π and ϕ, model A (and Σ) can also produce a correct-order advantage for recombined probes, $\mathrm{E}[(\mathbf{a} * \mathbf{d} + \mathbf{a} * \mathbf{l} + \mathbf{d} * \mathbf{r}) \cdot \mathbf{m}] > \mathrm{E}[(\mathbf{a} * \mathbf{d} + \mathbf{a} * \mathbf{r} + \mathbf{d} * \mathbf{l}) \cdot \mathbf{m}]$. Thus, models differ in their ability to produce correct-order advantages for recombined probes. In experiment 6 of Yang et al. (2013), associative recognition judgements were more accurate for intact pairs in the correct order (compared to incorrect order), but with no significant difference for recombined pairs. This may suggest that order did not influence judgements of recombined pairs, consistent with models Π and ϕ; however, because these were analyses of *accuracy* values, and not of d', they did not account for bias effects. In any case, our main point here is that symmetric associations can still cause associative recognition to depend on constituent-order if position/order is incorporated at encoding and at test.

Order incorporated into the item representation. Models were largely comparable when fitting order recognition data. However, the double function task posed a major challenge to models A, Σ, and ϕ. In these models, incorporating position into the cue vector retrieved a sum of the target item *and* items that shared the target item's position, ultimately providing no information to select between interfering pairs. Partial permutations (Model Π) overcame this issue because permuting an item by a given pattern does not increase similarity to other items permuted by that same pattern, allowing retrieval of a specific target, without retrieving items sharing the position of the target. The success of permutation here may provide some insight into how order is represented under certain conditions, revealing

unique advantages to embedding position information into item representations (although see section below).

The idea that item vectors are modified by their appearance in a word pair has precedence in existing memory models (Benjamin, 2010; Caplan, Chakravarty, & Dittmann, 2021; Criss & Shiffrin, 2004; Cox & Criss, 2020). For example, Benjamin's (2010) DRYAD model encoded context as a subset of each item's features to explain age-related memory deficits in context memory. If we assume that order/position is also part of context, this idea would be quite similar to our position-feature model. Caplan et al. (2021) proposed a model where certain features of a word were selectively attended to, while others were not, and set to zero. Attended features were based on the item it was paired with at study, implementing the idea certain meanings of a word are highlighted based on the context it appears in (e.g., BANK in RIVER BANK versus MONEY BANK). The model could use the pattern of attended features to judge pairings between items without storing any explicit associations. Vector permutation in our present model may be functionally related to this idea, although with an important difference. Like permutation, setting certain features of an item to zero effectively rotates item vectors in vector space, causing them to be dissimilar from the original word. However, unlike permutation, the effective dimensionality of item vectors is reduced, decreasing the information stored in memory.

Nonetheless, this brings up an important point. There are likely many other ways to embed positional information into an item vector that share the properties of permutation. As another example, one could imagine a hybrid of model A and ϕ where the positional features are the convolution of a subset of an item's features and a subset of positional vector features. This would make each item's positional features unique, meaning no interference between items that occupy the same position.

Experimental conditions may influence order-encoding strategies. Before ruling out models, it is important to consider experimental conditions under which modifying

item representations based on position (e.g., with permutation) may not be an optimal strategy. For example, if participants studied word pairs, but were only required to judge their constituent-order (rather than item-item pairings), it might be optimal to ignore associations between items and instead remember the individual positions of each item. In this case, a participant's cognitive strategy might be more consistent with the item-position association model (model A), or addition of item and position vectors (model Σ). Indeed, while model Π was quite successful with our current benchmarks, there were still 37 participants who did not have a clear winning model (benchmark 1b), and model ϕ won for four participants (Figure S5). This may already suggest that participants adopt qualitatively different order-encoding strategies that cannot be accounted for by a single model.

Non-commutative convolution in the brain. Both Plate (2000) and Kelly, Mewhort, and West (2017) noted that precisely implementing convolution in the brain would require intricate patterns of neural connectivity. However, even if a network of neurons is wired perfectly to compute convolution, it is unlikely that the synaptic strengths would be perfectly equal within the network. Unequal synaptic strengths may actually be useful from the perspective of encoding order. Consider the following expression, $\mathbf{a} * \mathbf{b}$, which can be expanded as follows,

$$\begin{bmatrix} a_1b_1 + a_2b_3 + a_3b_2 \\ a_1b_2 + a_2b_1 + a_3b_3 \\ a_1b_3 + a_2b_2 + a_3b_1 \end{bmatrix} \tag{39}$$

Now consider a network of neurons that computes this operation, but by random chance, has one synapse that is stronger than the rest, represented with the coefficient ζ ,

$$\begin{bmatrix} m_{f1} \\ m_{f2} \\ m_{f3} \end{bmatrix} = \begin{bmatrix} a_1b_1 + \underline{\zeta a_2b_3} + a_3b_2 \\ a_1b_2 + a_2b_1 + a_3b_3 \\ a_1b_3 + a_2b_2 + a_3b_1 \end{bmatrix} \tag{40}$$

If same network computes the reversed association, $\mathbf{m}_b = \mathbf{b} * \mathbf{a}$,

$$\begin{bmatrix} m_{b1} \\ m_{b2} \\ m_{b3} \end{bmatrix} = \begin{bmatrix} b_1a_1 + \underline{\zeta b_2a_3} + b_3a_2 \\ b_1a_2 + b_2a_1 + b_3a_3 \\ b_1a_3 + b_2a_2 + b_3a_1 \end{bmatrix} \tag{41}$$

We can infer constituent-order by comparing $\mathbf{m}_f$ and $\mathbf{m}_b$ as follows. First consider the dot product $\mathbf{m}_f \cdot \mathbf{m}_f$,

$$\begin{aligned} \mathbf{m}_f \cdot \mathbf{m}_f = (a_1b_1 + \zeta a_2b_3 + a_3b_2)^2 + \\ (a_1b_2 + a_2b_1 + a_3b_3)^2 + \\ (a_1b_3 + a_2b_2 + a_3b_1)^2 \end{aligned} \tag{42}$$

$$\begin{aligned} = (a_1^2b_1^2 + \zeta^2a_2^2b_3^2 + a_3^2b_2^2) + noise + \\ (a_1^2b_2^2 + a_2^2b_1^2 + a_3^2b_3^2) + noise + \\ (a_1^2b_3^2 + a_2^2b_2^2 + a_3^2b_1^2) + noise \end{aligned} \tag{43}$$

The expectation of this dot product can be expressed as the sum of specific products between random variables. The noise terms can be dropped for expectation calculations because they consist of odd powers of random variables, and the expectation for odd powers of standard normal distributed variables is 0 (Anderson, 1970). Thus, the expectation of this dot product can be expressed as follows,

$$= \mathrm{E}\left[\mathbf{m}_f \cdot \mathbf{m}_f\right] = (n^2-1)\mathrm{E}\left[X^2Y^2\right] + \zeta^2\mathrm{E}\left[X^2Y^2\right] \tag{44}$$

Where X and Y denote random variables drawn from $N(0,\sigma^2)$. Following Weber (1988), expectations of squared random variables can be substituted,

$$= \mathrm{E}\left[\mathbf{m}_f \cdot \mathbf{m}_f\right] = (n^2-1)\sigma^4 + \zeta^2\sigma^4 \tag{45}$$

Assuming that each element from **a** and **b** is drawn from $N(0,\sigma^2)$, and $\sigma^2 = \frac{1}{n}$, which produces approximately normalized vectors, this equation can be simplified even further,

$$= \mathrm{E}\left[\mathbf{m}_f \cdot \mathbf{m}_f\right] = \frac{(n^2-1)+\zeta^2}{n^2} \tag{46}$$

Equation 46 reveals that $\mathrm{E}\left[\mathbf{m}_f \cdot \mathbf{m}_f\right]$ has a quadratic relationship to ζ. For comparison, let us also derive $\mathrm{E}\left[\mathbf{m}_f \cdot \mathbf{m}_b\right]$, which is expanded as follows,

$$\begin{aligned}\mathbf{m}_f \cdot \mathbf{m}_b = (a_1b_1 + \zeta a_2b_3 + a_3b_2)(b_1a_1 + \zeta b_2a_3 + b_3a_2) + \\ (a_1b_2 + a_2b_1 + a_3b_3)(b_1a_2 + b_2a_1 + b_3a_3) + \\ (a_1b_3 + a_2b_2 + a_3b_1)(b_1a_3 + b_2a_2 + b_3a_1)\end{aligned} \tag{47}$$

$$\begin{aligned}= (a_1^2b_1^2 + \zeta a_2^2b_3^2 + \zeta a_3^2b_2^2) + noise + \\ (a_1^2b_2^2 + a_2^2b_1^2 + a_3^2b_3^2) + noise + \\ (a_1^2b_3^2 + a_2^2b_2^2 + a_3^2b_1^2) + noise\end{aligned} \tag{48}$$

Again dropping the noise terms because odd powers of random variables have expectations of zero, we can derive the expectation of this dot product as follows,

$$\mathrm{E}\left[\mathbf{m}_f \cdot \mathbf{m}_b\right] = (n^2-2)\mathrm{E}\left[X^2Y^2\right] + 2\zeta\mathrm{E}\left[X^2Y^2\right] \tag{49}$$

$$= (n^2 - 2)\sigma^4 + 2\zeta\sigma^4 \tag{50}$$

$$= \frac{(n^2 - 2) + 2\zeta}{n^2} \tag{51}$$

These equations reveal that $\mathrm{E}\left[\mathbf{m}_f \cdot \mathbf{m}_b\right]$ has a linear relationship to ζ, while $\mathrm{E}\left[\mathbf{m}_f \cdot \mathbf{m}_f\right]$ has a quadratic relationship, meaning that the difference between these dot product would increase with ζ. Thus, in this extremely simple implementation of a neural network with unequal synaptic strengths (with only one "strong" synapse), we can start to see how this type of mechanism could introduce differences between the forward and backward versions of an item-item associations that could be leveraged to infer order, and without any explicit mechanism to encode order. In sum, the simple assumption that convolution is not strictly commutative when implemented in the brain could provide a simple way to support order memory.

Applications to serial recall. Symmetric associations that support the ability to discriminate the item position could also be useful for models of serial recall. Associative chaining (e.g. Ebbinghaus, 1885/1913; Lewandowsky & Murdock, 1989) is a major class of model of serial recall, and assumes that a participant learns a list of words in order by forming direct item-item associations between neighbouring items. At test, the list is remembered by sequentially chaining through the items, using one item as the cue for its next. A major competitor to chaining models are positional coding models, which strictly avoid inter-item associations, and associate each item with a positional code (Conrad, 1960; Brown et al., 2007; Burgess & Hitch, 1999; Farrell, 2012; Henson, 1998). Although it is beyond the scope of this manuscript, evidence for and against both positional coding and associative chaining models has been reported (Caplan, 2015; Henson, Norris, Page, & Baddeley, 1996; Henson, 1998; Lindsey & Logan, 2019; Solway, Murdock, & Kahana,

2012), leading some researchers to propose hybrid or mixture models for a full account (Caplan, 2015; Logan & Cox, 2021; Kahana, 2012; Osth & Hurlstone, in press).

Implementations of chaining such as Lewandowsky and Murdock (1989); Solway et al. (2012), and Caplan, Ardebili, and Liu (2022) have used symmetric operations like convolution to encode item-item associations. Symmetric chaining models have clear strengths when accounting for certain behavioural benchmarks. For example, if participants accidentally skip an item during serial recall (A,... C), participants frequently go backward in the list and recall the missed word B (fill-in errors, Henson, 1998) rather than proceeding on with the word D (in-fill errors). As Osth and Dennis (2015a) argued, a chaining model such as Lewandowsky and Murdock (1989), with symmetric associations could skip an item during serial recall, and still retrieve missed item to produce fill-in errors with high frequency. [13] Symmetric chaining models may find it more difficult to account for findings that suggest item-item associations are somewhat directional. One clear example is that sequential probes of serial lists (given item i from the list, recall item $i+1$ or recall item $i-1$) are quite accurate, far above chance (e.g., Kahana & Caplan, 2002; Woodward & Murdock, 1968) suggesting that unlike TODAM, symmetry of association strengths does not entail the absence of order, but rather, something more like our current models. Applying any of our present order-encoding mechanisms to a symmetric chaining model (e.g., left-right patterns of partial permutation for each item-item association), would allow a model to have some ability to make use of the ability to discriminate the constituent-order of item-item association, while preserving the ability to progress backward and forward equally through

[13] Some studies have found that in-fill errors are more frequent (Caplan, 2015; Solway et al., 2012); however, Osth and Dennis (2015a) showed that if participants are not able or willing to explicitly mark omissions (e.g., typing "PASS"), this can inflate in-fill errors. This may not explain the near-equivalent ratios found in data where participants explicitly marked omissions by typing "PASS" (word lists) or using the spacebar (consonant lists), respectively Caplan (2015); Caplan et al. (2022). Surprenant, Kelley, Farley, and Neath (1999) found that the high fill-in:in-fill ratio did not interact with output position, which is problematic for positional-coding accounts of the phenomenon. Any model may need additional mechanisms to fit the data at this level of detail.

the list.

Conclusion

Multiple modifications to convolution preserved important properties of this model while adding some ability to judge constituent-order. While only position-specific permutations could successfully disambiguate double function pairs, future work should explore task conditions under which other order-encoding mechanisms (e.g., item-position associations) might be favored. Overall, this work demonstrates that there are a number of possible mechanisms by which symmetry can co-exist with some ability to judge constituent-order, but the partial permutation model accounted for the broadest set of empirical benchmarks.

References

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

Asch, S. E., & Ebenholtz, S. M. (1962). The principle of associative symmetry. *Proceedings of the American Philosophical Society*, *106*(2), 135-163.

Benjamin, A. S. (2010). Representational explanations of "process" dissociations in recognition: The dryad theory of aging and memory judgments. *Psychological Review*, *117*(4), 1055-1079.

Bishop, Y. M., Fienberg, S. E., & Holland, P. W. (1975). *Discrete multivariate analysis: theory and practice.* Cambridge, MA: MIT Press.

Brown, G. D. A., Neath, I., & Chater, N. (2007). A temporal ratio model of memory. *Psychological Review*, *114*(3), 539-576.

Burgess, N., & Hitch, G. J. (1999). Memory for serial order: A network model of the phonological loop and its timing. *Psychological Review*, *106*(3), 551-581.

Burnham, K. P., & Anderson, D. R. (2004). Multimodel inference understanding AIC and BIC in model selection. *Sociological Methods & Research*, *33*(2), 261-304.

Caplan, J. B. (2015). Order-memory and association-memory. *Canadian Journal of Experiemntal Psychology*, *69*(3), 221-232.

Caplan, J. B., Ardebili, A. S., & Liu, Y. S. (2022). Chaining models of serial recall can produce positional errors. *Journal of Mathematical Psychology*, *109*, 102677.

Caplan, J. B., Boulton, K. L., & Gagné, C. L. (2014). Associative asymmetry of compound words. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *40*(4), 1163-1171.

Caplan, J. B., Chakravarty, S., & Dittmann, N. L. (2021). Associative recognition without hippocampal associations. *Psychological Review*.

Caplan, J. B., Rehani, M., & Andrews, J. C. (2014). Associations compete directly in memory. *Quarterly Journal of Experimental Psychology*, *67*(5), 955-978.

Conrad, R. (1960). Serial order intrusions in immediate memory. *British Journal of Psychology*, *51*(1), 45-48.

Cox, G. E., & Criss, A. H. (2017). Parallel interactive retrieval of item and associative information from event memory. *Cognitive Psychology*, *97*(5), 31–61.

Cox, G. E., & Criss, A. H. (2020). Similarity leads to correlated processing: A dynamic model of encoding and recognition of episodic associations. *Psychological Review*, *127*(5), 792–828.

Criss, A. H., & Shiffrin, R. M. (2004). Pairs do not suffer interference from other types of pairs or single items in associative recognition. *Memory & Cognition*, *32*, 1284–1297.

Criss, A. H., & Shiffrin, R. M. (2005). List discrimination in associative recognition and implications for representation. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *31*(6), 1199-1212.

Dressler, W. U. (2006). Compound types. In G. Libben & G. Jarema (Eds.), *The representation and processing of compound words* (p. 23-44). Oxford University Press.

Ebbinghaus, H. (1885/1913). *Memory: A contribution to experimental psychology*. New York: Teachers College, Columbia University.

Farrell, S. (2012). Temporal clustering and sequencing in short-term memory and episodic memory. *Psychological Review*, *119*(2), 223-271.

Greene, R. L., & Tussing, A. A. (2001). Similarity and associative recognition. *Journal of Memory and Language*, *45*, 573-584.

Haskins, A. L., Yonelinas, A. P., Quamme, J. R., & Ranganath, C. (2008). Perirhinal cortex supports encoding and familiarity-based recognition of novel associations. *Neuron*, *59*, 554-560.

Henson, R. N. A. (1998). Short-term memory for serial order: the Start-End Model. *Cognitive Psychology*, *36*(2), 73-137.

Henson, R. N. A., Norris, D. G., Page, M. P. A., & Baddeley, A. D. (1996). Unchained memory: Error patterns rule out chaining models of immediate serial recall. *Quarterly Journal of Experimental Psychology*, *49A*(1), 80-115.

Hintzman, D. L. (1984). MINERVA 2: A simulation model of human memory. *Behavior Research Methods, Instruments, & Computers*, *16*(2), 96-101.

Horowitz, L. M., Brown, Z. M., & Weissbluth, S. (1964). Availability and the direction of associations. *Journal of Experimental Psychology*, *68*(6), 541-549.

Howard, M. W., Jing, B., Rao, V. A., Provyn, J. P., & Datey, A. V. (2009). Bridging the gap: transitive associations between items presented in similar temporal contexts. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *35*(2), 391-407.

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.

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.

Jones, M. N., & Mewhort, D. J. K. (2007). Representing word meaning and order information in a composite holographic lexicon. *Psychological Review*, *114*(1), 1-37.

Kahana, M. J. (2002). Associative symmetry and memory theory. *Memory & Cognition*, *30*(6), 823-840.

Kahana, M. J. (2012). *Foundations of Human Memory*. USA: Oxford University Press.

Kahana, M. J., & Caplan, J. B. (2002). Associative asymmetry in probed recall of serial lists. *Memory & Cognition*, *30*(6), 841-849.

Kato, K., & Caplan, J. B. (2017). Order of items within associations. *Journal of Memory and Language*, *97*, 81-102.

Kelly, M. A., Blostein, D., & Mewhort, D. J. K. (2013). Encoding structure in holographic reduced representations. *Canadian Journal of Experimental Psychology*, *67*(2), 79-93.

Kelly, M. A., Mewhort, D. J. K., & West, R. L. (2017). The memory tesseract: mathematical equivalence between composite and separate storage memory models. *Journal of Mathematical Psychology*, *77*, 142-155.

Kounios, J., Bachman, P., Casasanto, D., Grossman, M., & Smith, W., Roderick W. Yang. (2003). Novel concepts mediate word retrieval from human episodic associative memory: evidence from event-related potentials. *Neuroscience Letters*, *345*, 157-160.

Kounios, J., Smith, R. W., Yang, W., Bachman, P., & D'Esposito, M. (2001). Cognitive association formation in human memory revealed by spatiotemporal brain imaging. *Neuron*, *29*, 297-306.

Lashley, K. S. (1951). The problem of serial order in behavior. In L. A. Jeffress (Ed.), *Cerebral mechanisms in behavior* (p. 112-146). New York, NY: John Wiley and Sons.

Lewandowsky, S., & Murdock, B. B. (1989). Memory for serial order. *Psychological Review*, *96*(1), 25-57.

Lindsey, D. R., & Logan, G. D. (2019). Item-to-item associations in typing: Evidence from spin list sequence learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *45*(3), 397-416.

Logan, G. D., & Cox, G. E. (2021). Serial memory: Putting chains and position codes in context. *Psychological Review*, *128*(6), 1197–1205.

Metcalfe Eich, J. (1982). A composite holographic associative recall model. *Psychological Review*, *89*(6), 627-661.

Murdock, B. B. (1962). Direction of recall in short-term memory. *Journal of Verbal Learning and*

Verbal Behavior, *1*(2), 119-124.

Murdock, B. B. (1974). *Human memory: Theory and data.* Potomac, MD: Lawrence Erlbaum and Associates.

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. (1995). Developing TODAM: three models for serial-order information. *Memory & Cognition*, *23*(5), 631-645.

Osth, A. F., & Dennis, S. (2015a). The fill-in effect in serial recall can be obscured by omission errors. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *41*(5), 1447-1455.

Osth, A. F., & Dennis, S. (2015b). Sources of interference in item and associative recognition memory. *Psychological Review*, *122*(2), 260-311.

Osth, A. F., & Hurlstone, M. J. (in press). Do item-dependent context representations underlie serial order in cognition? Commentary on Logan (2021). *Psychological Review*.

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.

Plate, T. A. (2000). Analogy retrieval and processing with distributed vector representations. *Expert Systems: The International Journal of Knowledge Engineering and Neural Networks*, *17*(1), 29-40.

Primoff, E. (1938). Backward and forward associations as an organizing act in serial and in paired-associate learning. *Journal of Psychology*, *5*, 375-395.

Recchia, G., Jones, M. N., Sahlgren, M., & Kanerva, P. (2010). Encoding sequential information in vector space models of semantics: comparing holographic reduced representation and random permutation. In S. Ohlsson & R. Catrambone (Eds.), *Proceedings of the 32nd cognitive science society* (p. 865-870).

Rehani, M., & Caplan, J. B. (2011). Interference and the representation of order within associations. *Quarterly Journal of Experimental Psychology*, *64*(7), 1409-1429.

Rizzuto, D. S., & Kahana, M. J. (2000). Associative symmetry vs. independent associations. *NeuroComputing*, *32-33*, 973-978.

Rizzuto, D. S., & Kahana, M. J. (2001). An autoassociative neural network model of paired-associate learning. *Neural Computation*, *13*, 2075-2092.

Shiffrin, R. M., & Steyvers, M. (1997). A model for recognition memory: REM— Retrieving Effectively From Memory. *Psychonomic Bulletin & Review*, *4*, 145-166.

Slamecka, N. J. (1976). An analysis of double-function lists. *Memory & Cognition*, *4*(5), 581-585.

Solway, A., Murdock, B. B., & Kahana, M. J. (2012). Positional and temporal clustering in serial order memory. *Memory & Cognition*, *40*(2), 177-190.

Sommer, T., Schoell, E., & Büchel, C. (2008). Associative symmetry of the memory for object–location associations as revealed by the testing effect. *Acta Psychologica*, *128*, 238-248.

Surprenant, A. M., Kelley, M. R., Farley, L. A., & Neath, I. (1999). Fill-in and infill errors in order memory. *Memory*, *13*(3/4), 267-273.

Thomas, J. J., Ayuno, K. C., Kluger, F. E., & Caplan, J. B. (2022). The relationship between interactive-imagery instructions and association-memory. *Memory & Cognition*.

Weber, E. U. (1988). Expectation and variance of item resemblance distributions in a convolution-correlation model of distributed memory. *Journal of Mathematical Psychology*, *32*, 1-43.

Wiegand, I., Bader, R., & Mecklinger, A. (2010). Multiple ways to the prior occurrence of an event: an electrophysiological dissociation of experimental and conceptually driven familiarity in recognition memory. *Brain Research*, *1360*, 106-118.

Woodward, A. E., & Murdock, B. B. (1968). Positional and sequential probes in serial learning. *Canadian Journal of Psychology*, *22*(2), 131-138.

Yang, J., Zhao, P., Zhu, Z., Mecklinger, A., Fang, Z., & Han, L. (2013). Memory asymmetry of forward and backward associations in recognition tasks. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *39*(1), 253-269.

Supplementary Materials

Benchmark 1a

Visualization of fits to benchmark 1a data. Figure S1 and S2 are heat maps of BIC values around the best fitting parameter sets for each model's fit to benchmark 1a (Figure 4). In order to visualize BIC values in 2-D space, we plotted two separate heat maps for each model. Plotted on the left are model order parameters (μ, n_{perm}, or n_p) versus n, and plotted on the right, each model's order parameter versus σ_α. For each heat-map, the third free parameter (which does not have an axis) is set at the model's best fitting value. In general, visual inspection of these plots show a gradual slope in BIC values around best fitting parameter sets, suggesting our best fits were robust and not due to noise.

Benchmark 1b

Distribution of best-fitting model parameters. Plotted in figure S3 are histograms of the distribution of best-fitting model parameters to benchmark 1b. We included separate plots for each free parameter and model, and also the descriptive statistics for these distributions in Table S1. These plots indicate that models used a wide range of parameter values to fit to individual participant performance, with roughly even distribution across the total explored parameter space (Figure S3, Table S1). There were some notable exceptions. For example, parameter distributions for n and σ_α in model Σ were noticeably skewed, using high values of n and low values of σ_α to fit participants, both of which reduce the overall noise in the memory trace. Interestingly, because model Σ had an additional noise term compared to model A, it may have used parameters n and σ_α to counteract this effect. For model ϕ, the distributions of parameter σ_α and n were similarly skewed (Figure S3).

Plots of individual participant fits from benchmark 1b. Figure S4 plots model predictions from fits to individual participant data.

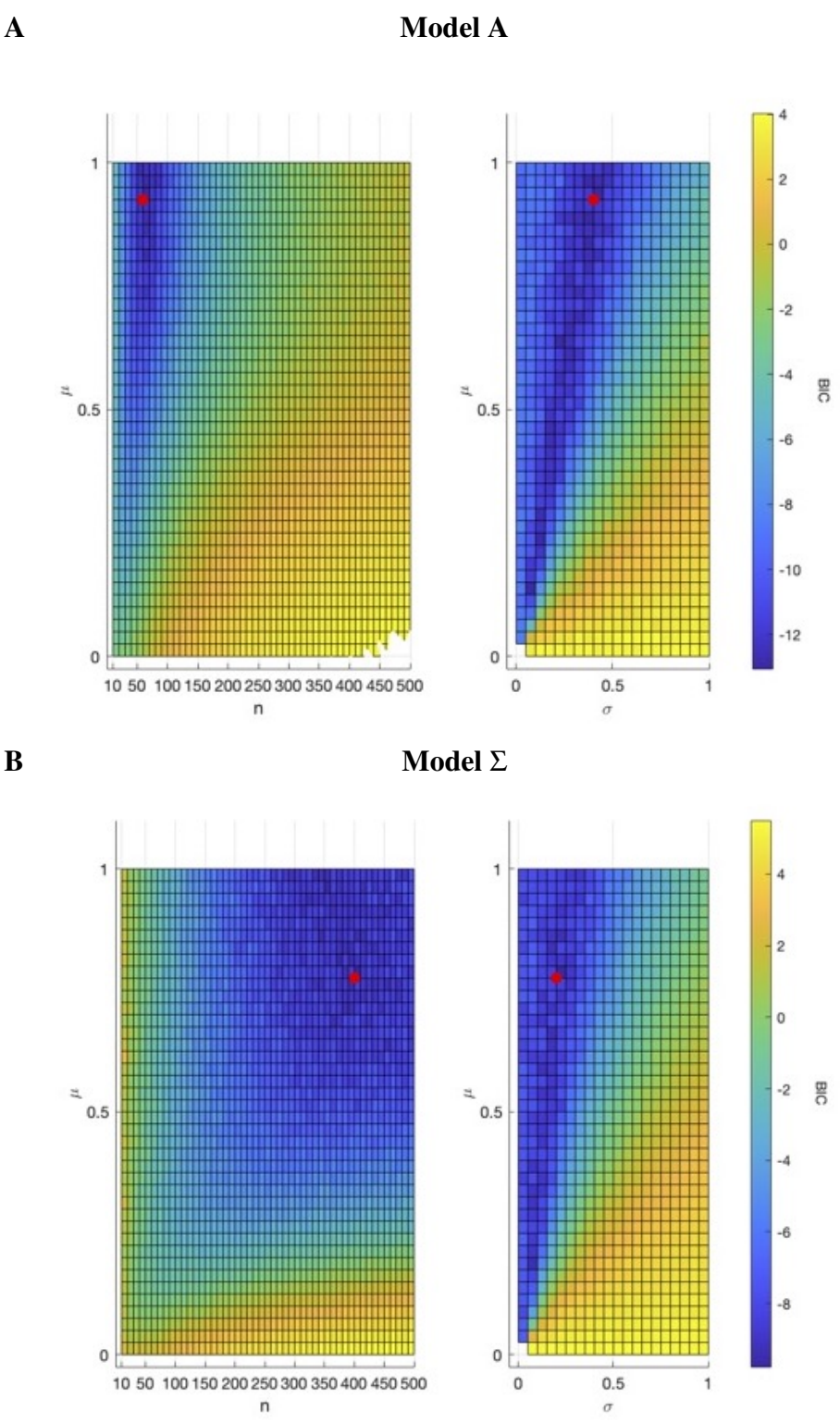
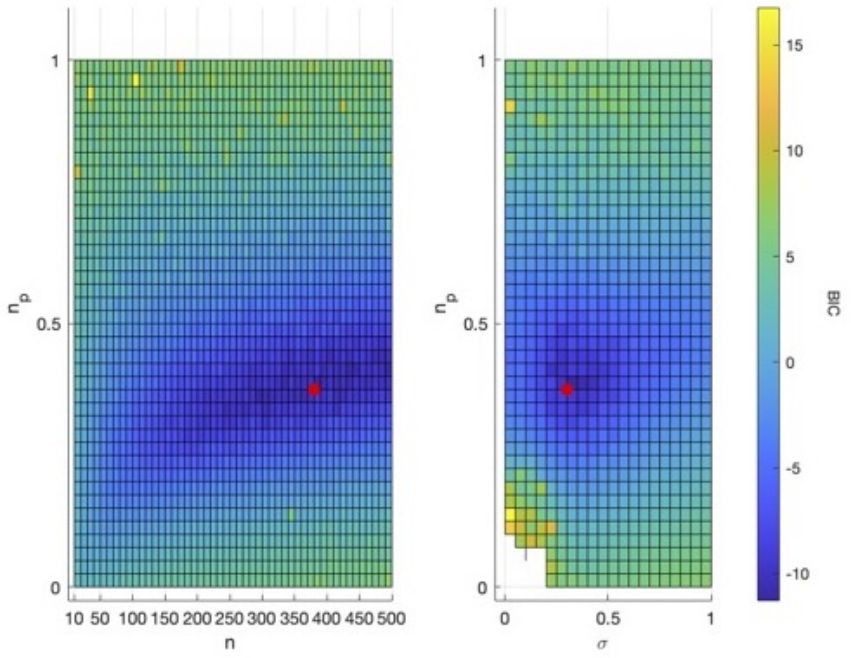


Figure S1. Heat maps depicting BIC values around best fitting parameter sets for models A and Σ. On the left is each model's order parameter versus n, and on the right, each model's order parameter versus σ_α. Red circles denote the minimum BIC value.

C **Model ϕ**



D **Model Π**

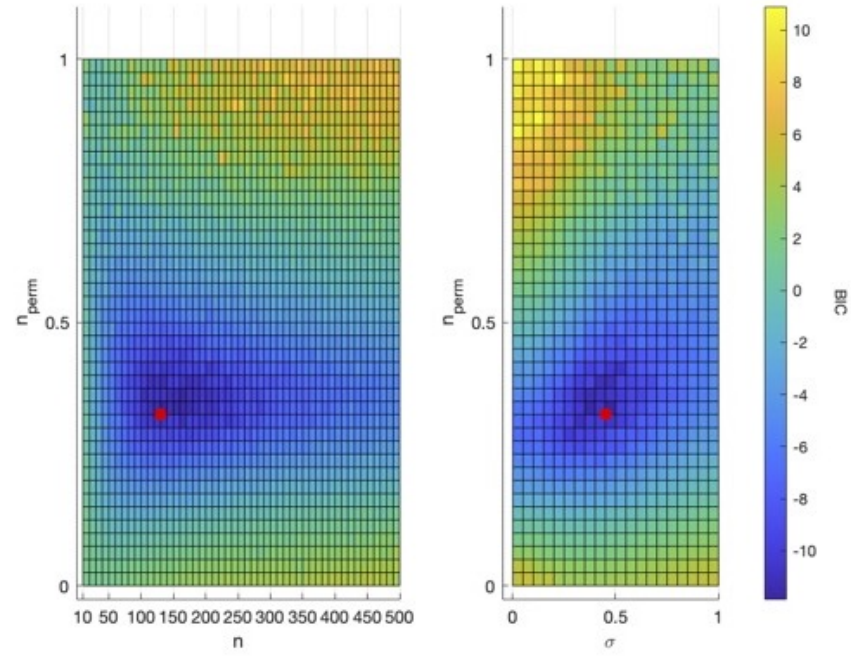


Figure S2. Heat maps depicting BIC values around best fitting parameter sets for model ϕ and Π. On the left is each model's order parameter versus n, and on the right, each model's order parameter versus σ_α. Red circles denote the minimum BIC value.

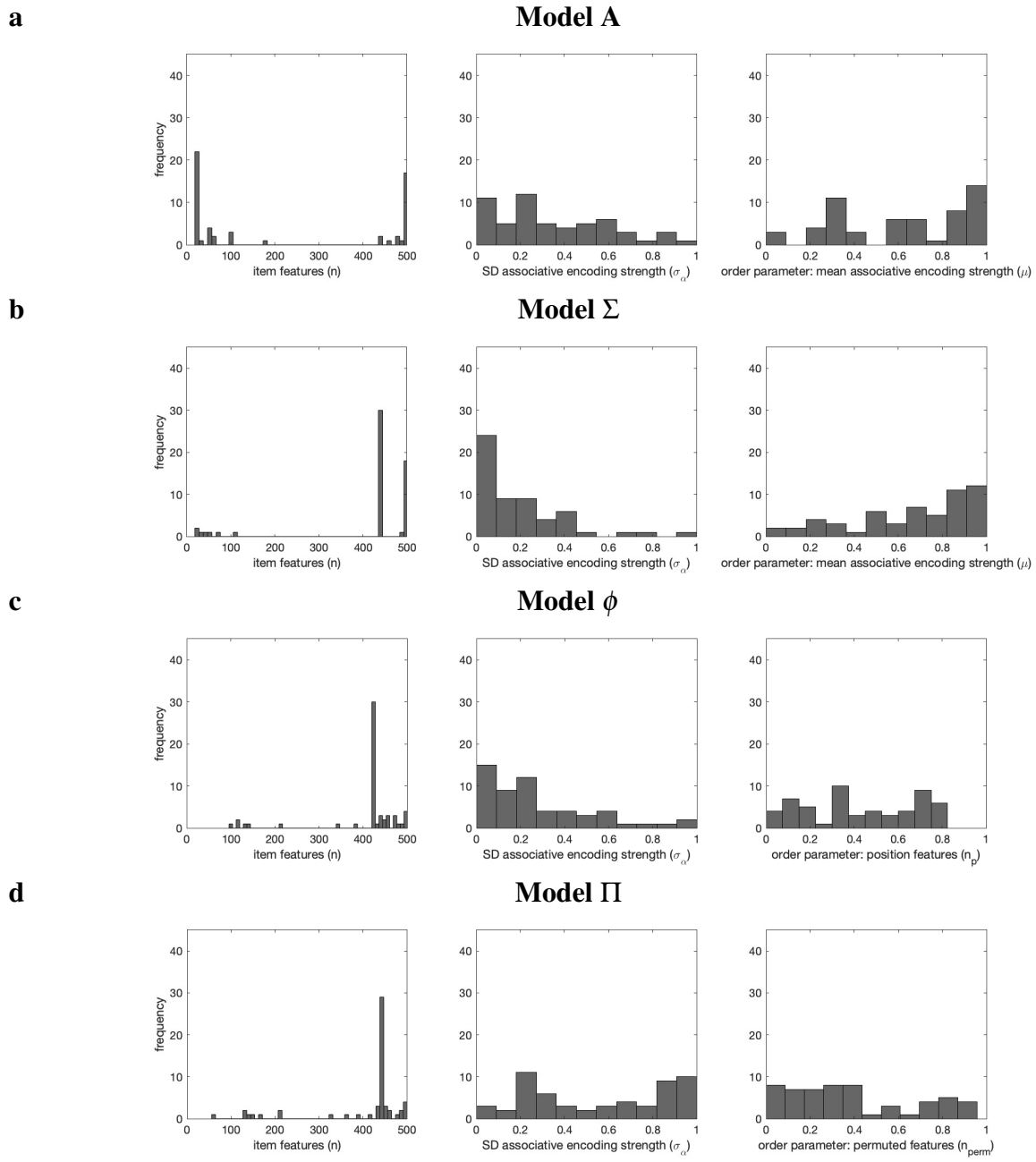


Figure S3. Histograms of the distributions of each model's free parameters for fits to benchmark 1b. Panels on the left plot the distribution of total item-features (n) used for model fits. Panels in the middle plot the distribution of the standard deviation of associative encoding strengths (σ_α) for model fits. Panels on the right plot the distribution of each model's order parameter values (μ,n_p,n_{perm}).

Table S1

The mean and standard deviation (Mean(SD)) for the distribution of each model's free parameters used for fits to individual participants in benchmark 1b, along with Mean(SD) of the distribution of BIC values. This distribution of parameters was also applied to benchmark 1c.

Model	n	σ_α	Order parameter	BIC
Model A	224 (227)	0.34 (0.26)	$\mu = 0.62(0.31)$	1.04 (4.97)
Model Σ	411 (141)	0.19 (0.21)	$\mu = 0.67(0.29)$	2.57 (4.09)
Model ϕ	403 (98)	0.27 (0.26)	$\frac{n_p}{n} = 0.42(0.25)$	–4.35 (6.29)
Model Π	401 (109)	0.56 (0.32)	$\frac{n_{perm}}{n} = 0.39(0.29)$	–7.09 (8.38)

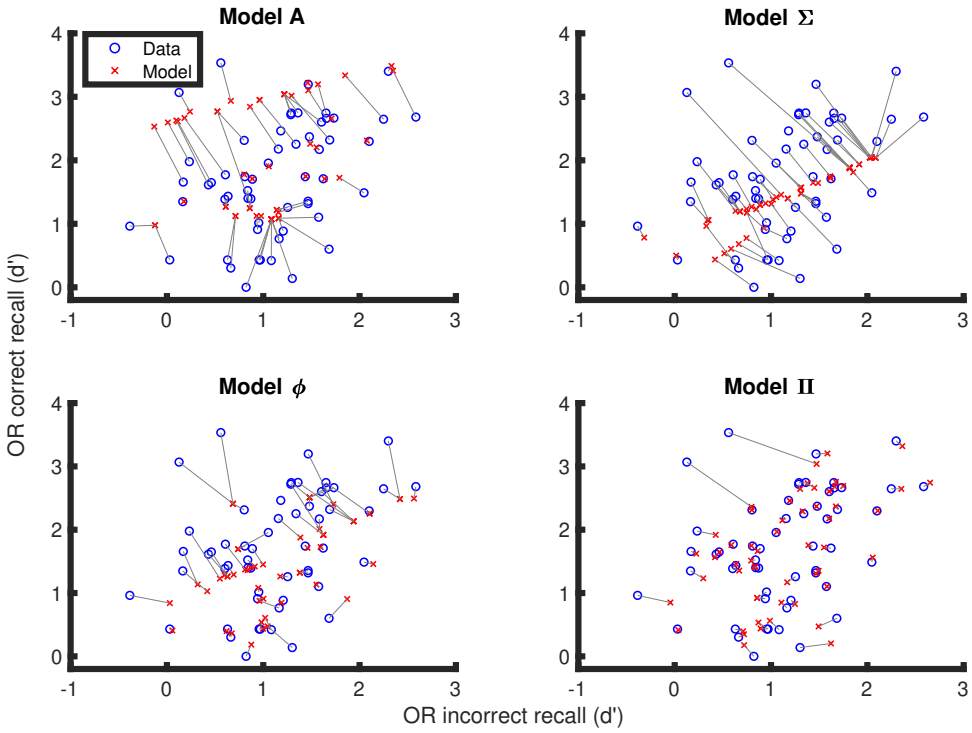


Figure S4. Scatter plots of individual participant order recognition d' data from Thomas et. al. (2022) for correct versus incorrectly recalled pairs, along with model fits to individual participants. Each circle is a participant, and crosses plot model predictions from best fits to each participant. Each grey line connects a participant and the best-fitting model prediction for that participant.

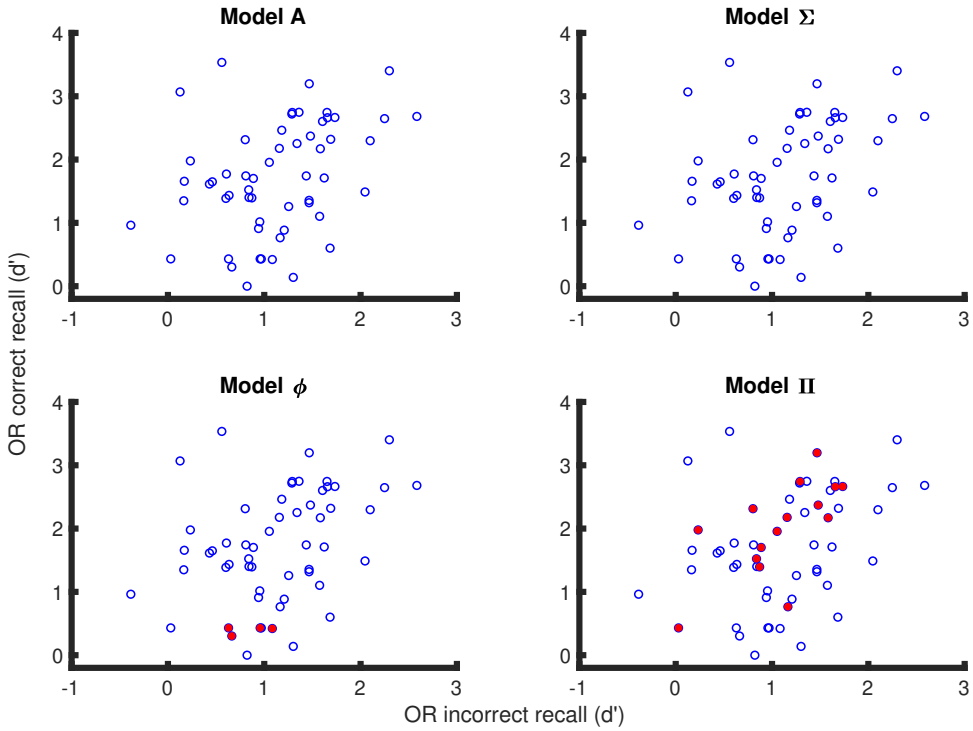


Figure S5. Scatter plots of individual participant order recognition d' data from Thomas et. al. (2022) for correct versus incorrectly recalled pairs. Each circle (open and filled) is a participant. Filled-in circles in each plot denote the participants that each model could fit substantially better than other models (by a margin of BIC$>$ 2). Open circles in each plot were participants which the model did not fit best. In total, 37 participants did not have clear winning model.

Plots comparing model fits to benchmark 1b with a winner-take-all rule. Figure S5 plots participants for which each model provided a substantially better fit than other models (ΔBIC$>$ 2).

Benchmark 1c: Between-subject correlations between recognition and recall extrapolated from fits to benchmark 1b

Thomas et al. (2022) also examined *between-subject* correlations between both recognition tasks and cued recall performance. These were consistent with benchmark 1a; there was a moderate correlation between order recognition and cued recall performance, but this was well below the correlation between associative recognition and cued

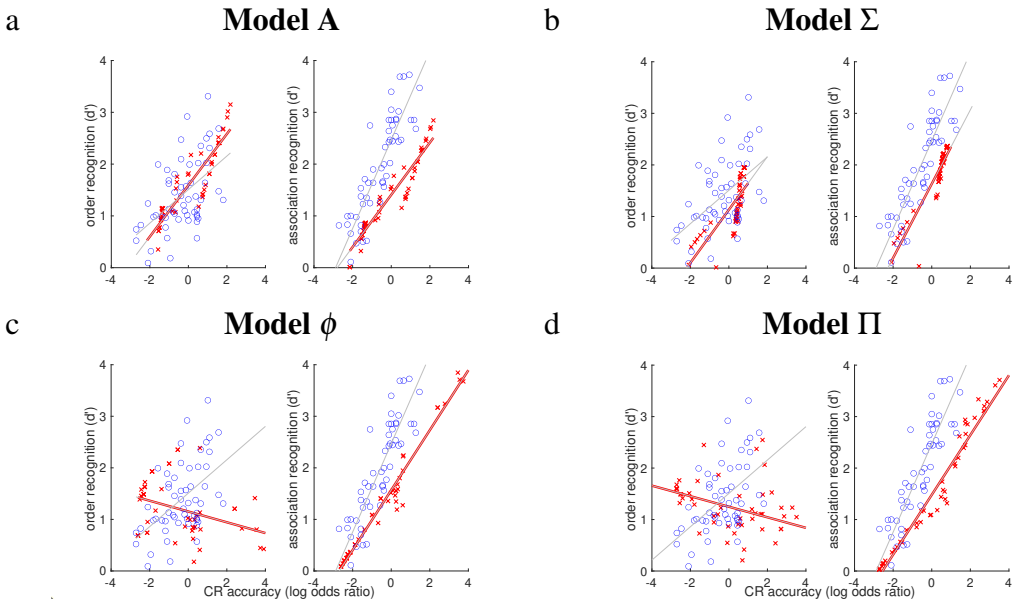


Figure S6. Model predictions and empirical data for order recognition versus log-odds transformed cued recall accuracy (left panels), and also associative recognition versus log-odds transformed cued recall accuracy (right panels). Model predicted values were generated in fits to benchmark 1b. Least squares lines for model predicted values are plotted in red, and least square lines for behavioural data are plotted in light grey. Each circle is a participant, and crosses plot model predicted values.

recall (Figure S6). We wondered if models would also exhibit a moderate between-subject relationship between cued recall and order recognition. Rather than re-fit the models, we simply plotted model output from previous fits to benchmark 1b. Note, this meant that plotted model predictions for associative recognition d' were not included in the original fitness measure at all. This placed models at a significant disadvantage when producing the associative recognition-cued recall correlation, especially considering that a completely different set of participants were tested for associative recognition in Thomas et al. (2022).

Results. For models ϕ and Π the order recognition-cued recall correlation was smaller than the associative recognition-cued recall correlation, but fell short in accounting for the magnitude of correlations observed in behaviour (Figure S6). Models A and Σ produced order recognition-cued recall correlations that were comparable to the associative

recognition-cued recall correlation, essentially predicting a maximal relationship between order recognition and cued recall. Despite the mismatch between model predictions and the magnitude of each correlation in behaviour, because models were not quantitatively fit to this data, the conclusion here is not that models are unable to account for benchmark 1c. In fact, although we do not report it here, models were quite good at producing the behavioural values for each correlation when directly fit to quantitative values for each participant. Rather, the conclusion here is that models ϕ and Π explain individual differences by dissociating order recognition from cued recall performance (more than associative recognition from cued recall), while for models A and Σ, performance in all tasks is correlated.