Psychological Review
2006, Vol. 113, No. 2, 201–233

0033-295X/06/$12.00 DOI: 10.1037/0033-295X.113.2.201

Short-Term Memory for Serial Order: A Recurrent Neural Network Model

Matthew M. Botvinick
University of Pennsylvania

David C. Plaut
Carnegie Mellon University

Despite a century of research, the mechanisms underlying short-term or working memory for serial order remain uncertain. Recent theoretical models have converged on a particular account, based on transient associations between independent item and context representations. In the present article, the authors present an alternative model, according to which sequence information is encoded through sustained patterns of activation within a recurrent neural network architecture. As demonstrated through a series of computer simulations, the model provides a parsimonious account for numerous benchmark characteristics of immediate serial recall, including data that have been considered to preclude the application of recurrent neural networks in this domain. Unlike most competing accounts, the model deals naturally with findings concerning the role of background knowledge in serial recall and makes contact with relevant neuroscientific data. Furthermore, the model gives rise to numerous testable predictions that differentiate it from competing theories. Taken together, the results presented indicate that recurrent neural networks may offer a useful framework for understanding short-term memory for serial order.

Keywords: working memory, short-term memory, serial order, computational models

Short-term memory for serial order has been a central topic in psychology almost since the discipline's inception. The earliest published studies on the topic go back to the 19th century (Nipher, 1876), and by the 1930s, enough work had been done to warrant the publication of a literature review (Blankenship, 1938). Today, the PsycINFO database lists over 800 articles under the search term *serial recall*, with over 250 of these also found under *short-term* or *working memory*, and over 150 focusing on the single, hallmark task of immediate serial recall (ISR). Taken together, existing work yields an extremely detailed characterization of human performance, describing effects of list length and composition, effects of presentation modality, detailed characteristics of errors, and many other features of serial recall (for reviews, see Marshuetz, 2005; Neath, 1998).

From the beginning, research on serial memory[1] has placed a strong emphasis on the development of explicit theories. Over the years, such theories have evolved from verbal, analogy-based accounts (e.g., Conrad, 1965; Murdock, 1974) to box-and-arrow schematics (e.g., Baddeley, 1986), to abstract mathematical characterizations and algorithms (Anderson & Matessa, 1997; Drewnowski, 1980; Estes, 1972; Lewandowsky & Murdock, 1989; Nairne, 1990). In the most recent generation of models, the trend has been toward connectionist or neural network models, which seek to characterize the mechanisms involved in serial memory in terms that can be mapped onto neural hardware (e.g., G. Brown, Preece, & Hulme, 2000; Burgess & Hitch, 1999; Houghton, 1990; Page & Norris, 1998).

Considering the sophistication and explanatory power of recent models, one might assume that the problem of serial order in short-term memory has essentially been solved. However, this is not at all the case. Although substantial progress has been made toward clarifying the ramifications of various theoretical perspectives (see Henson, 1999), the mechanisms underlying serial recall are still vigorously debated, even at the most fundamental levels.

In the current article, we propose a new theory of short-term memory for serial order. Our approach takes, as its foundation, previous work in both psychology and neuroscience, examining the function of recurrent neural networks. Through computer simulation, we have applied a recurrent network to the task of ISR, evaluating its ability to account for a core set of empirical benchmarks. The model accounts well for a wide range of phenomena, including findings that had been thought to rule out recurrent networks as candidate models of serial recall. At the same time, the model is capable of accounting for a set of findings, relating to interactions between short- and long-term memory, that has presented difficulty for competing accounts.

In what follows, we provide a context for the work to be reported, by contrasting two general frameworks for understanding sequence memory. Following this, we introduce the details of our own approach, and then turn to the simulation results.

Matthew M. Botvinick, Departments of Psychiatry and Psychology and Center for Cognitive Neuroscience, University of Pennsylvania; David C. Plaut, Departments of Psychology and Computer Science and Center for the Neural Basis of Cognition, Carnegie Mellon University.

The present work was supported by National Institutes of Health Awards MH16804 to Matthew M. Botvinick and MH64445 to James McClelland and David C. Plaut, et al.

Correspondence concerning this article should be addressed to Matthew M. Botvinick, Center for Cognitive Neuroscience, University of Pennsylvania, 3720 Walnut Street, Philadelphia, PA 19104-6241. Email: mmb@mail.med.upenn.edu

[1] Throughout the article, we use the terms *serial memory* and *serial recall* interchangeably, setting aside the fact that *serial memory* refers more to a mental faculty and *serial recall,* to a specific task, which is used to tap that mental faculty.

Two Frameworks for Understanding Memory for Serial Order

Recurrent Neural Networks and Activation-Based Memory

One of the earliest proposals concerning the mechanism underlying serial recall was made by Hebb (1949). Hebb suggested that serial memory depends on what he termed "activity traces." Specifically, the proposal was that information about the identities of items in a sequence, and their serial order, are represented through sustained patterns of neural activation. From early on, it was suggested that such sustained activity might emerge as the result of recurrent connectivity in the brain, which created "reverberatory loops" (Conrad, 1959; Lashley, 1951).

Over the years since these original proposals, the view of serial recall as based on activation traces in a recurrent architecture has garnered considerable empirical support, particularly from neuroscientific data. It is now widely accepted that sustained neural activation plays a central role in short-term, or working, memory (Fuster, 1997; Goldman-Rakic, 1995; E. K. Miller & Cohen, 2001; O'Reilly, Braver, & Cohen, 1999) and that recurrent synaptic connectivity plays a critical role in supporting such activation (Compte, Brunel, Goldman-Rakic, & Wang, 2000; Zipser, Kehoe, Littlewort, & Fuster, 1993). Neurophysiologic studies with nonhuman primates suggest that the same considerations may apply to memory for serial order. Specifically, during ISR, neurons in prefrontal cortex have been found to encode item and position information and to remain active between list encoding and recall, providing a distributed representation of the target sequence (Barone & Joseph, 1989; Funahashi, Inoue, & Kubota, 1997; Ninokura, Mushiake, & Tanji, 2003, 2004). Several computational models have shown how the patterns of activity observed in these neurophysiologic studies of serial recall can be accounted for on the basis of the dynamics of recurrent neural networks connecting cortex, basal ganglia, and thalamus (e.g., Beiser & Houk, 1998; Dominey, Arbib, & Joseph, 1995; see Figure 1). Such studies have demonstrated the sufficiency of recurrent neural networks to support the recall of sequences. Indeed, the capacity of recurrent networks to encode sequence information has become an area of study in its own right within computer science and engineering (Jaeger, 2001; Maass, Natschlager, & Markram, 2002; White, Lee, & Sompolinsky, 2004).

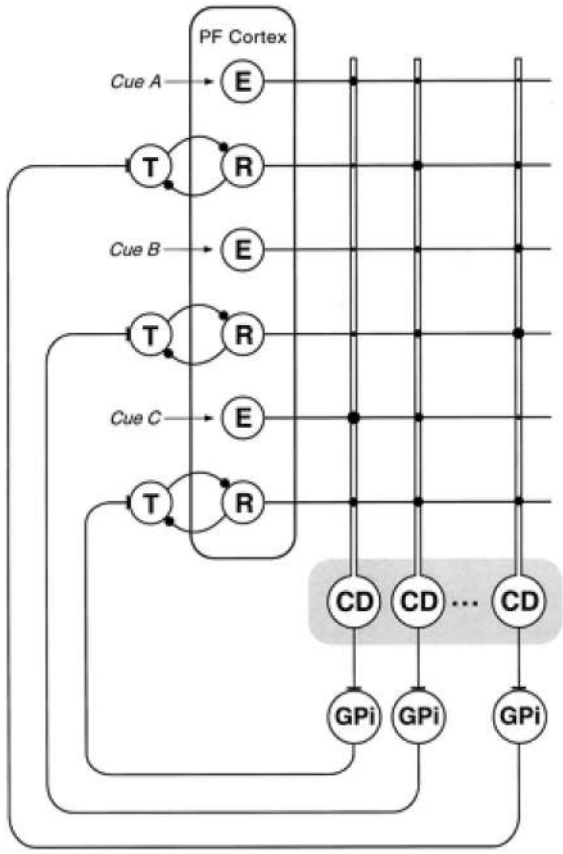

Figure 1. A recurrence-based model of the neural mechanisms underlying immediate serial recall, proposed by Beiser and Houk (1998). T = thalamus; PF cortex = prefrontal cortex; CD = caudate; GPi = globus pallidus interna; R = recurrent prefrontal unit; E = encoding prefrontal unit. From "Model of Cortical–Basal Ganglionic Processing: Encoding the Serial Order of Sensory Events," by D. G. Beiser and J. C. Houk, 1998, *Journal of Neurophysiology, 79,* p. 3170. Copyright 1998 by the American Physiological Society. Reprinted with permission.

Arguments Against Recurrent Networks

Given the accumulating neuroscientific evidence for the involvement of recurrent neural circuitry in supporting serial recall, it is striking to find that many psychologists in fact reject recurrent networks as candidate models in this domain. The central argument that has been leveled against recurrent networks forms an important backdrop for the work we present here, so it is worth laying out in some detail.

The primary objection derives from the view that recurrent networks can produce sequences only by relying on interitem associations, a mechanism referred to as *chaining* (G. Brown et al., 2000; Burgess & Hitch, 1999; Henson, Norris, Page, & Baddeley, 1996; Houghton, 1990; Houghton & Hartley, 1995). This assumption—inaccurate, we argue—is based on the observation that recurrent networks, because of their connectivity, are influenced by their own earlier internal states. It has also been encouraged by the fact that recurrent networks have, to date, been most frequently applied to tasks requiring prediction of items in a sequence on the basis of their predecessors (e.g., Cleeremans, 1993; Elman, 1990).

If recurrent networks were in fact simply chaining models, it is true that this would exclude them as models of serial recall. For although it was long considered a possibility that short-term memory for serial order might be based on chaining (Lewandowsky & Murdock, 1989; Wickelgren, 1966), subsequent empirical work has succeeded in conclusively ruling out this hypothesis. The critical observation was made by Baddeley (1968; see also Bjork & Healy, 1974; Henson et al., 1996). Here, subjects were asked to recall six-item consonant lists that alternated between acoustically confusable and nonconfusable items. Not surprisingly, recall for the nonconfusable items was superior to that for confusable ones, yielding a "sawtooth" pattern (shown in Figure 9). The pivotal finding, however, was that recall for nonconfusable items in these alternating lists was as accurate as recall for items in the same positions in lists made up completely of nonconfusable items (see Figure 9). If recall were conducted on the basis of chaining, one would expect the similarity between confusable items to lead to exchanges among their nonconfusable successors (see Henson et

al., 1996, for a detailed presentation of this argument). However, no such tendency was observed.

Henson et al. (1996) have provided another argument against chaining, which has to do with what Wickelgren (1966) called "associative intrusions" and what the Henson group calls "relative errors." Here, two items that appear adjacent to one another in the input list are moved to new positions at recall, but remain adjacent and in their original order. Chaining models predict that such errors should occur at above-chance levels, because when an item migrates to a new position, item-to-item associations should often lead that item's successor to follow. Henson et al. (1996) provided empirical evidence contradicting this prediction, showing that relative errors were not as frequent as a chaining account would necessarily predict.

Context-Based Accounts

The accumulation of evidence against chaining has led to a shift, within the psychological literature, away from models based on interitem associations (e.g., TODAM; Lewandowsky & Murdock, 1989) and toward a framework that depends instead on transient associations between item representations and a time-dependent representation of context. Within this framework (see, e.g., G. Brown et al., 2000; Burgess & Hitch, 1992, 1999; Grossberg, 1986; Hartley & Houghton, 1996; Henson, 1996, 1998; Houghton, 1990; Houghton & Hartley, 1995), recall is accomplished by resetting the context representation to the state it held at the beginning of list encoding and then stepping through the context representation's successive states, using them to retrieve the associated item representations. The main point of variation among the relevant accounts concerns the nature of the context representation, which may be understood as a representation of list position (Anderson & Matessa, 1997), distance from list start or end (Henson, 1998; Houghton, 1990), or the state of a set of neural oscillators (G. Brown et al., 2000; Burgess & Hitch, 1999). In each case, however, serial recall depends on transient links between item and context representations.

The majority of context-based models has been presented in the form of neural networks. Here, the associative links between item and context representations are established by changing the connection weights between processing units (analogous to the synapses between neurons in a biological neural network). When cast in neural terms, the context-based account can thus be characterized as using a *weight-based* method for encoding and maintaining serial order information, a point that strongly differentiates it from the *activation-based* framework considered earlier.

Unlike recurrent, activation-based models, context-based models have been applied to a wide range of detailed behavioral findings, including list-length effects, transposition gradients, effects of item similarity and modality, effects of grouping, and changes in performance over development, not to mention numerous others. Their considerable success in accounting for such data has made context-based models the standard against which any competing account of serial recall must presently be compared.

It is, however, important to point out that there is at least one area that has proven challenging for context-based models. This involves cases in which serial recall is influenced by long-term or background knowledge about sequential structure. One classic example of such an influence is the *bigram frequency effect*, first reported by Baddeley (1964; see also Baddeley, Conrad, & Hull, 1965; Kantowitz, Ornstein, & Schwartz, 1972; Mayzner & Schoenberg, 1964). Here, strings of letters are better recalled if adjacent items are also likely to appear together in English words. Similar effects have been demonstrated with other kinds of material: G. A. Miller and Selfridge (1951) found that lists of words were better recalled if they contained pairwise transitions that are likely to occur in actual sentences. More recently, Botvinick (2005; Botvinick & Bylsma, 2005) has shown that the same effect can be observed using a set of pseudowords sequenced on the basis of an artificial grammar; after extended experience with sequences generated on the basis of the grammar, subjects were better able to recall orderings that were associated with a high probability under the grammar, than less probable orderings. Finally, it has been shown that lists of nonwords are better recalled if the nonwords contain high-frequency phoneme-to-phoneme transitions, suggesting that short-term memory is influenced by background knowledge concerning phonotactic structure (Gathercole, 1995; Gathercole, Frankish, Pickering, & Peaker, 1999; Gathercole, Willis, Emslie, & Baddeley, 1991; Grant et al., 1997; Roodenrys & Hinton, 2002; Van Bon & Van der Pijl, 1997). In these examples, short-term memory for serial order is seen to depend on background knowledge concerning domain-specific regularities in sequential structure. In each case, recall for highly probable sequences was better than for less probable ones.

Note that, in each of the cases just reviewed, the relevant background knowledge involves transition probabilities among specific items. It is this that makes the observed effects difficult for context-based models to address. Given the strong evidence against chaining in short-term memory for serial order, context-based models have eschewed any role for item-to-item associations. Although this allows such models to account for phenomena such as the sawtooth error pattern described earlier, it makes it difficult for them to account simultaneously for effects of long-term sequence knowledge, in which information about transition probabilities appears critical. The difficulty has prompted at least one proponent of the context-based approach to acknowledge that such "interactions between short- and long-term memory pose problems for most models of serial recall" (Henson, 1998, p. 115).

The Present Work

We have contrasted two general approaches to understanding short-term memory for serial order. One, involving weight-based associations between context and item representations, has been successfully applied to a broad range of behavioral phenomena but faces difficulty in accounting for effects of background knowledge. The other, involving an activation-based memory mechanism, supported by recurrent connectivity, is consistent with neuroscientific findings but has not been tested in any detailed way against behavioral data, a fact partly attributable to specific doubts concerning its viability.

In the present article, we argue for the latter, recurrent activation-based account of short-term memory for serial order. Our approach is to implement the general account in the form of a simple recurrent neural network and to use this model to simulate a set of critical behavioral phenomena, all relating to the task of ISR. The results we present support two general conclusions. The first is that, contrary to the opinion frequently expressed in the existing literature, recurrent networks can, in fact, account for central benchmark phenomena, including the findings that rule out

chaining as a mechanism for short-term serial recall. The second conclusion is that, unlike context-based models, recurrent networks also provide a viable account for effects of background knowledge on serial recall.

The Model: Core Features

Our model is a deliberately minimal implementation of the recurrence- and activation-based account of serial memory. The approach was simplified wherever possible, to focus on the implications of a core set of theoretical assumptions. The key aspects of the model are summarized as follows:

1. *Neural network implementation*. The model takes the form of a connectionist, or artificial neural, network, composed of simple, interconnected processing units with continuously varying activation values. The units in the model are divided into an input group, used to represent list items presented during encoding; an output group, which represents the system's responses during list encoding and during recall; and an internal or "hidden" group, which mediates between input and output (see Figure 2).

2. *Recurrent connectivity*. The model's architecture is characterized by massively recurrent connectivity, forming feedback loops over which activation can reverberate. Specifically, feedback connections run among the units in the model's hidden layer, allowing the pattern of activation in this layer on each time step to influence its behavior on the next. A second (less critical) feedback loop involves connections running from the output layer back to the hidden layer, allowing the model's outputs to influence its subsequent states.

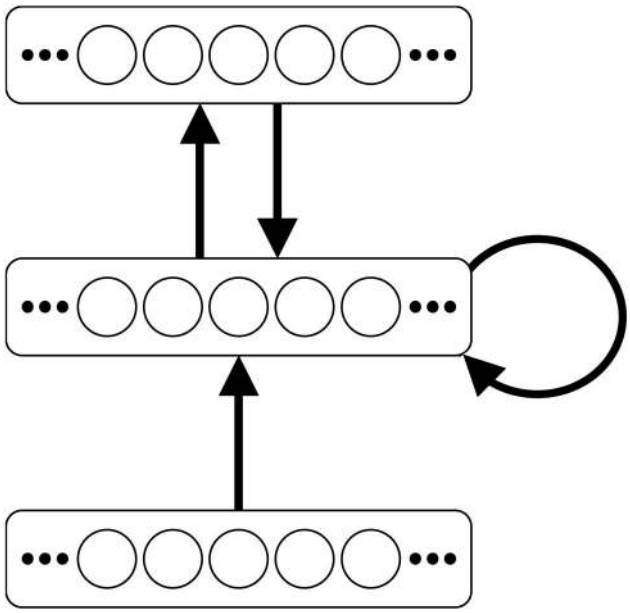

Figure 2. Schematic representation of the simulation model. Arrows represent all-to-all projections.

3. *Role of domain-specific experience*. The system's connection weights are set through the operation of a learning procedure, which adjusts the weights—gradually, over many trials—so as to reduce output error during performance of the serial recall task. This aspect of the simulations has two key implications. First, it means that the network's weights are shaped only by the basic demands of the ISR task; they are in no way chosen to yield the benchmark performance patterns (e.g., primacy, recency, and error patterns) that we consider. Second, and more important, it means that if the model is trained in a domain involving regularities of sequential structure, this exposure can, in principle, influence the mechanism that develops for performing the serial recall task.

4. *Activation-based, rather than weight-based, encoding*. Simulations using the model are divided into training and testing phases, each consisting of many individual trials. Although, as just noted, the system's weights change during training, this is not the case in the testing phase, in which the performance of the fully trained model is evaluated. Here, all of the system's connection weights are held constant. Because the weights are not allowed to change over the course of testing trials, the network cannot use a weight-based mechanism for encoding and preserving sequence information. Its successful performance of the ISR task must rely, instead, on sustained patterns of unit activation, supported by the system's recurrent connectivity.[2]

5. *Random variability*. The internal representations of the model (i.e., patterns of activations over hidden units) display a degree of random variability. In some simulations, this variability derives from noise added to the model's hidden unit activations. In others, as explained later, it derives from sampling variability in the training process itself. In both cases, the presence of variability in the model's encodings plays a central role in its recall errors.

[2] Our choice to freeze the system's weights during testing should not be interpreted as an assertion that short-term memory mechanisms in the brain become incapable of learning at some point. The purpose of this measure was simply to emphasize the distinction between weight-based and activation-based encoding. In further simulations, to be reported elsewhere (Botvinick & Huffstettler, 2006), we have considered the performance of the model when learning is permitted to continue during testing. Under these conditions, the model displays the Hebb effect (Hebb, 1949). That is, if a specific target list is presented on every third trial, performance on this list gradually improves, whereas performance on intervening filler lists remains stable. In recent empirical work, Cumming, Page, and Norris (2003) compared performance on Hebb (repeated) lists to transfer lists, which overlapped with the Hebb lists only at every other list position. As Cumming et al. pointed out, positional theories (including the weight- and context-based theories we have grouped together here) predict that recall accuracy on these transfer lists should be higher at the positions matching the Hebb list. However, this was not observed empirically. Instead, performance on the transfer lists closely resembled performance on ordinary filler lists. As we plan to report in detail elsewhere (Botvinick & Huffstettler, 2006), the present model displays the same pattern of performance.

Again, our approach was to simplify wherever possible, so as to focus on the implications of these core stipulations. This strategy has the advantage of making the basis for the model's behavior relatively clear. However, it also means that the model bears a rather abstract relationship to the behavioral processes and neural mechanisms it is meant to elucidate. The role of learning requires specific comment, in this regard. As noted above, the model is trained on the task of ISR. Obviously, it is not our claim that the human capacity to perform ISR develops through direct practice on the task. Rather, the training procedure we used is meant to arrive at a pattern of connectivity that, in the human, would arise from a combination of innate constraints and experience with behavioral tasks that place a premium on sequence information—perhaps most important, language comprehension and production. This point, as well as the overall role of learning in our simulations, are discussed further in the General Discussion section.

Full details of the model's implementation are presented in the next section. Following this, we provide a general description of the sequencing mechanism the model developed through training. Next, we detail the specific results of four simulations. Simulations 1 and 2 address a basic set of behavioral phenomena, including list-length effects, primacy and recency effects, transposition gradients, and effects of interitem confusability. In Simulations 3 and 4, the same basic model was applied to phenomena involving interactions between short- and long-term memory.

General Method

Model Architecture

In all instantiations, the network was composed of simple units assuming, on each step of processing, an activation level between zero and one. These were partitioned into an input group of between 13 and 43 units, depending on the simulation; an output group containing the same number of units as the input group; and an internal or hidden layer of 200 units. Input units were externally set to represent the input appropriate to the current time step. Activation values for units in the other two layers were set according to a standard approach (Rumelhart & McClelland, 1986). Specifically, these units assumed activation levels based on their current net input, calculated as follows:

$$net_j = \sum_i a_i w_{ij}, \tag{1}$$

where a_i is the activation of unit i, and w_{ij} is the weight of the connection from unit i to unit j. For units in the hidden layer, activations were based on the logistic function:

$$a_j = \frac{1}{1 + e^{-net_j}}. \tag{2}$$

As described below (see the *Sources of Variability* section), a small amount of random noise was added to hidden unit activations in Simulation 4. Activations in the output layer were based on the softmax function

$$a_j = \frac{e^{net_j}}{\sum_k e^{net_k}}, \tag{3}$$

where k indexes all of the units in the output layer. The softmax activation function simulates a form of competition within the layer, ensuring that overall activation in the layer sums to one. Moreover, when used with the divergence error metric described below, it can be understood as representing the posterior probability of a particular event from a set of mutually exclusive alternatives (Rumelhart, Durbin, Golden, & Chauvin, 1996).

The connectivity of the network, in all instantiations, was as illustrated in Figure 2. Activation fed forward from the input layer to the hidden layer, and from the hidden layer to the output layer. Recurrent connections also allowed activation to flow from the output layer to the hidden layer (see the Training and Testing sections for additional information on this projection) and from units in the hidden layer to all other units in the same layer. For every projection, all units in the sending layer were connected to all units in the receiving layer. Each unit in the hidden and output layers also received an input from a single bias unit, with a fixed activation of one.

Weights were initialized to random values between −1 and 1 (for recurrent connections, ±0.5; weights from the bias unit to hidden units were initialized at −1, to avoid strong hidden unit activation early in training, which can slow the learning process). Operation of the network was in discrete time, with each step corresponding to an event in the task, either presentation of a list item at encoding or output of an item at recall. On each time step, activations in the hidden layer were determined prior to activations in the output layer. Recurrent connections were associated with a one time-step delay, as is conventional in the simple recurrent network implementation (Elman, 1990). As a result of this conduction delay, the pattern of activation in the hidden layer was determined by the joint influence of (a) the pattern of activation in the input layer, (b) the pattern of activation in the hidden layer on the previous time step, and (c) the pattern of activation in the output layer on the previous time step (following winner-take-all action selection, as described further below).

Task and Representations

The input and output representations used in the simulations were straightforward. Simulations 1, 3, and 4 used single units to represent individual list items (English letters in Simulations 1 and 3, pseudowords in Simulation 4). To simulate presentation of an item, we assigned the input unit standing for that item an activation of one, with all other input units set to zero. It should be noted that this localist representation scheme was not necessary; distributed representations could just as well have been used. Localist representations were used for simplicity and to avoid spurious similarities. As described in the Simulation 2 section, that study used two-dimensional item representations, allowing inclusion of confusable items (items identical on one dimension) and nonconfusable (nonoverlapping) items. All instantiations of the model included a special unit in the input layer to serve as the recall cue, and one in the output layer to signal the end of recall.

The task addressed in all simulations was forward, ISR. During encoding, individual item representations in the model's input layer were activated on successive time steps, with the task being to activate the corresponding item representation in the output layer. Following the final element in the target list, the recall cue unit in the input layer was activated. This unit remained activated throughout the recall phase, during which the task was to output the items in the target list, one per time step and in their original order. The recall cue was the only input unit activated during recall. At the end of recall, the task called for the network to activate a special output unit indicating that recall was complete. List lengths presented during training and testing varied in length from 1 to 6, 8, or 9, depending on the simulation. (To minimize training time, the maximum list length for each simulation was chosen to match the maximum list length in the target empirical studies. Training to longer list lengths made little difference in the model's behavior.) It is important to note that, in all simulations, the model was trained on multiple list lengths. This is not an incidental aspect of the simulations; training the model only on a single list length was found to produce results qualitatively different from those that are reported here.

Training

The model was trained on the ISR task as just described. Training began with a single-element list. Following this, list length increased by one

element per trial until a simulation-specific maximum length was reached. Following presentation of a list of maximum length, the list length returned to one and the cycle repeated. In what follows, we refer to the processing of a single list (including both encoding and recall) as a *trial*, and a single pass through the full range of list lengths as a training *cycle*.

Lists presented during training in Simulations 1 and 2 were composed of input elements selected randomly without replacement. Lists in Simulations 3 and 4 were generated on the basis of specific transition frequencies, as described in conjunction with the simulations themselves.

Learning was accomplished using a version of recurrent backpropagation through time, adapted to the simple recurrent network architecture (Williams & Zipser, 1995), using binary target activations. The divergence error metric was used:

$$\sum_{j} t_j \log\left(\frac{t_j}{a_j}\right), \tag{4}$$

where j indexes across output units. When divergence error is combined with the softmax activation function in the setting of a multinomial classification problem, such as the present ISR task, the learning procedure yields a network that approximates a maximum a posteriori or Bayesian classifier (Rumelhart et al., 1996), a point that becomes important in some analyses of the model's behavior below.

The learning rate was set at 0.001 for all simulations. Weights were updated at the end of each trial. During training, teacher forcing was used in generating the feedback from output to hidden layers. That is, the activations propagated over the recurrent connections from output to hidden layers were based on the target values for the output units, not their actual activation. At the beginning of each trial, activations for all units in the hidden layer were set to 0.5, and for all units in the output layer, to 0.0 (determining the activations propagated over recurrent connections on the first time step).

For each simulation, training proceeded until the network reached a predetermined level of recall accuracy (proportion of lists of a selected length recalled correctly). For this purpose, performance was evaluated after every 10,000 trials. The reference accuracy level was drawn from the relevant empirical studies, as detailed in subsequent sections. It should be noted that, beyond the noise parameter used in Simulation 4, length of training was the only parameter in our simulations that was varied so as to optimize the fit to empirical data.

Testing

At test, the weights in the network were held constant, and further lists were presented. Characteristics of the test lists are described in conjunction with each simulation. As also detailed there, the probability of any list appearing during both training and testing was, in general, very small.

The network's response was identified by selecting the most highly activated unit in the output layer, ignoring the end-of-list unit. The number of responses collected was set equal to the length of the target lists. This testing method was analogous to presenting subjects in an ISR experiment with fill-in boxes for their responses and insisting that they provide a response in each box, forbidding omissions. In simulations in which recall was terminated upon selection of the end-of-list unit (not further reported here), the network responded with the incorrect list length infrequently. For example, in Simulation 1, this occurred on 0.0% of trials for lists of three elements, 1.7% of trials for lists of six elements, and 8.9% of trials for lists of nine elements. Further simulations in which a response threshold was imposed, allowing omission errors to occur, did not change the overall pattern of data to be reported here.

Unless otherwise noted, accuracy data presented in this report are based on averages over 5,000 trials (at the relevant list length), a sample size associated with negligible variance. Data presented for each simulation are based on a single set of weights, generated in a single training run. However, in all cases, the reported patterns of behavior were found to be highly reproducible, reliably emerging each time the model was trained.

Evaluation of Performance

The specific behavioral phenomena to be addressed in each simulation, and the associated approach to analysis, are introduced in conjunction with the simulations themselves. In each case, the behavioral data provide clear qualitative contrasts or trends, and model performance was evaluated on the basis of the degree to which it displayed the same qualitative patterns. Nevertheless, for completeness, root-mean-square error (RMSE) is reported for fits involving more than two empirical data points. Note that, given our use of large sample sizes, the data presented provide a fairly precise indication of the central tendencies characterizing model performance; our goal was not to model the degree of variability in empirical data sets or to address individual differences.

In selecting which phenomena to address, we applied three criteria. First, we included basic phenomena that have come to be accepted as standard benchmarks for computational models in the domain. Second, we included phenomena that have been considered to militate against the application of recurrent networks to serial recall or to provide special support for accounts different from the one presented here. Finally, we included phenomena that highlight unique aspects of the present account, in particular the behavioral data concerning the role of background knowledge in serial recall.

Sources of Variability

In some settings (in particular, Simulation 4), normally distributed noise, with mean zero, was added to the network's hidden unit activations during testing. In other simulations, noise was not added; however, even here, other factors led the network's internal representations to display a degree of random variability. The sources of this variability relate to the learning process. As described above, during training, the model's weights were modified in response to each list it processes. Because learning was based on error reduction, these modifications were guaranteed to benefit the processing of the just-presented list. However, the weight modifications made on each trial were not constrained by how they might affect performance on other possible lists. Because the learning rate was small, the weights converged on values that allowed the model to successfully process a very large set of stimuli. However, even after the model had converged in this way, further training caused the weights in the system to "bounce around" to a small degree, based on the model's recent training history. The result was essentially equivalent to intrinsic activation noise, in that the encodings of individual sequences varied stochastically on the basis of their relationships to the model's recent training history. This point is illustrated by Figure 3, which compares hidden unit activations following encoding of the same list on two separate trials, separated by presentations of other lists.

Given the presence of this second source of variability, we elected for simplicity to set the noise parameter to zero in most of our simulations.[3] Further work indicated that the addition of activation noise did not change

[3] It should be noted that during the testing phase in our simulations, the weights in the model were held constant. Thus, the learning-driven "bouncing around" just described was not occurring during testing. However, the impact of this learning-driven variability is nonetheless evident in the performance of the model when aggregated over a large set of target lists, simply because a proportion of such lists will be incorrectly recalled as a result of learning-induced weight changes occurring toward the end of training. In most of our simulations, average recall performance over a large sample of sequences could thus be used as an indirect measure of the probability of recall for any specific list, just as in empirical studies. A special note pertains to Simulation 4. Here, the space of possible lists was small enough that sequence-specific learning effects had a detectable impact on performance measured in the aggregate. To compensate for this, activation noise was used in this simulation, allowing internal representations (and consequently performance) to vary across repeated presentations of the same target sequence.

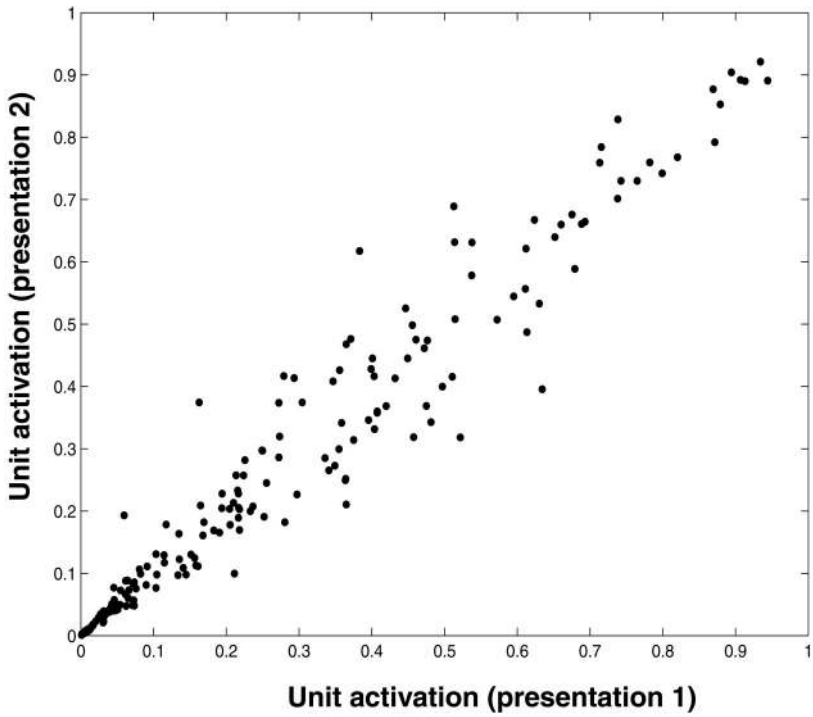

Figure 3. Variability in sequence encoding. The *x*-axis indicates the activation of each hidden unit on an arbitrarily selected presentation; the *y*-axis indicates the activation of the same unit, for the same stimulus list, following 100 cycles of further training.

the qualitative behavior of the model, as characterized in the simulations reported in the present article. All simulations were run on a Dell Precision computer, with dual Pentium 4 processors, using the LENS simulator (Rohde, 1999).[4]

Initial Analyses: How the Trained Model Works

Although the general operation of the model has been described, the specific mechanisms that allowed the model to perform the ISR task were not built in a priori but instead resulted from the learning process. A set of analyses, described in the Appendix, revealed that the learning process consistently resulted in a particular solution to the ISR task. In the present section, we provide an overview of this solution, with an emphasis on points relevant to understanding the model's overt performance.

To facilitate the following discussion, it will help to define several terms. We use the term *element* to refer to a single member of a target list. *Item* will be used to refer to the content of a list element, considered in isolation from that element's *position*. Thus, for illustration, in the list *BKRLM*, we would say that the element at Position 3 contains Item *R*.

Internal Representation of Sequences and List Elements

A basic demand of the ISR task is the ability to establish and maintain a representation of the target sequence. In the model, such a representation is carried by the units in the hidden layer. At the onset of recall, the pattern of activation in this layer must encode information concerning each list item and its respective list position. The model's solution to this problem involves a *superpositional* coding of list elements. Each list element is represented with a particular pattern of activation, and the list is represented as a superposition or summation of these patterns. The end result is a single vector of activation over the model's hidden units, but a vector that can be decomposed into element vectors each representing a single list element. In what follows, we refer to these components as *element vectors* or *element representations*.

As detailed in the Appendix, it proved possible to isolate specific element vectors through regression analyses, which indicated how the presentation of specific list elements (e.g., Item *M* at Position 4) affected the activation of each hidden unit. Examination of the resulting element vectors revealed three important points concerning the way that individual list elements are represented within the hidden layer. First, list elements are represented *independently*: The way in which a list element is represented does not depend on the other elements in the list. This makes sense, given the combinatorial structure of the target lists. Second, within the model's element representations, item and position are coded *conjunctively*. That is, the way that a given item is represented varies, depending on its position within the list. This is in fact a computational necessity, given the superpositional code used by the model. If item and position were represented independently, ambiguities would arise concerning the linkage between specific items and specific positions. Conjunctive coding thus addresses the need to bind item information with position information.

[4] Network specification files and stimulus generation scripts sufficient to recreate the simulations presented here are available for download from http://www.ccn.upenn.edu/~mmb/

The third, and perhaps most important, point concerning the model's element representations involves their *similarity relations*. When pairs of element representations are compared, they are found to resemble one another to an extent determined by both the items and the positions involved. Specifically, element representations tend to resemble one another to the degree that they involve similar items, and to the degree that they involve nearby positions. The point is illustrated in Figure 4. This shows average correlations for pairs of element vectors representing items at the same list position, or else separated by one, two, or three positions. The plot contains three data series, one based on pairs of vectors representing confusable items (as defined under the General Method section), one on pairs representing nonconfusable items, and one on pairs representing the identical item. Note that the correlation between element representations depends on the similarity between the items they represent; at each relative position, vectors representing the same item are more similar than vectors representing confusable items, and these are more similar than vectors representing nonconfusable items. However, as the figure also illustrates, the resemblance between element vectors also depends on the distance between the positions of the elements represented. Elements occupying the same position are represented more similarly than elements at adjacent positions, and with increasing distance, correlations continue to fall.

This similarity structure in the model's representations of item and position turns out to be critical for explaining the model's performance in the simulations to be reported below, and we refer to it frequently in subsequent analyses. An obvious question raised by this similarity structure is why it arises. The answer here has to do with the way that the model's internal representations support the selection of appropriate outputs, and how they evolve over the course of a trial, as discussed next.

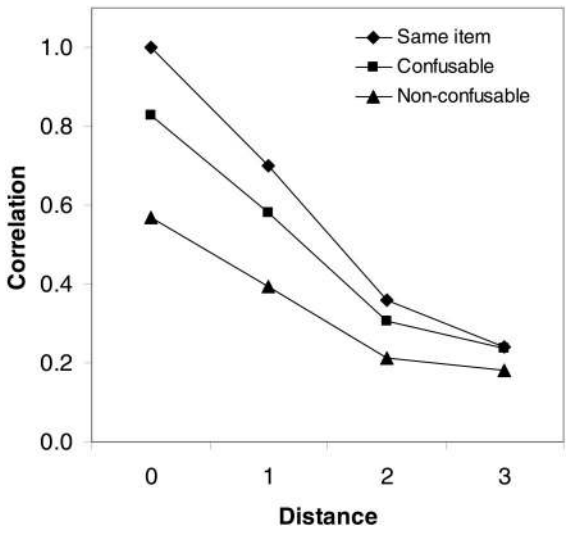

Figure 4. Mean correlations between element vectors for items at the same list position (0), or at distances of 1, 2, or 3 positions. Correlation for the same item at the same position, necessarily 1, is included for completeness. Confusable, nonconfusable = acoustically confusable and nonconfusable items.

Selection of Successive Outputs

Given the model's connectivity, the activation pattern in the model's hidden layer determines what units become active in the output layer. The hidden layer thus has two roles. Not only must it encode all of the elements in the target list. It must also somehow indicate which *single* item representation should be activated in the output group. These two demands add up to an interesting computational challenge. At any given time, the hidden layer must represent one element in such a way that it influences the output layer, while representing other elements in such a way that they do not. One might say that on each time step, one element must be made visible to the output layer, while other elements are kept invisible. Such "output gating" is, in fact, a generic challenge faced by any neural system relying on an activation-based memory mechanism (see, e.g., J. W. Brown, Bullock, & Grossberg, 2004; Hochreiter & Schmidhuber, 1997).

To understand the mechanism underlying output gating in the present model, consider that the influence of the hidden units on each output unit is determined by a specific, fixed set of connection weights. As explained under the General Method section, the input to each output unit is defined as the dot product of the hidden-layer activation vector with this weight vector. Graphically, this constitutes a projection of the hidden layer activation vector onto the weight vector for the output unit (see Jordan, 1986). The strength of this projection, and thus the strength of the output unit's activation, depends on the degree to which the hidden layer activation vector is aligned or correlated with the weight vector. A pattern of activation that aligns well with the weight vector will drive the output unit strongly; it will be "visible" to the output layer, in the above sense. Patterns less well correlated will have less influence on the output unit. And patterns orthogonal to the weight vector will not influence the output unit at all. Such patterns will, in effect, be "invisible" to the output unit.

These properties of the model are what allow it to output one item on each time step, while still managing to keep other list elements in memory. On steps in which an element is being encoded or recalled, the element is represented so as to be comparatively visible to the output layer. That is, it is represented by a vector of activation that is relatively well aligned with the weight vector connecting the hidden layer to the relevant output unit. On other time steps, the element is rerepresented so as to be comparatively invisible to the output layer. The element still figures robustly in the model's overall representation of the target sequence, but it is represented in such a way that it does not strongly influence activation in the output layer.

A demonstration of these points is provided in Figure 5. Each step along the *x*-axis in this plot relates to a single step in processing a four-element list: four steps of encoding and four steps of recall. At each step, the data points relate to element vectors for particular list positions (identified as described in the Appendix). The plot shows, for each step, the degree to which element vectors for each list position are "visible" to the output layer. This visibility is quantified as the cosine of the angle between each element vector and the weight vector for the relevant output unit (averaged across element vectors pertaining to a single list position and a single step of the task). The larger this cosine, the better the alignment between element and weight vectors, and the larger the projection of the element vector onto the output layer.

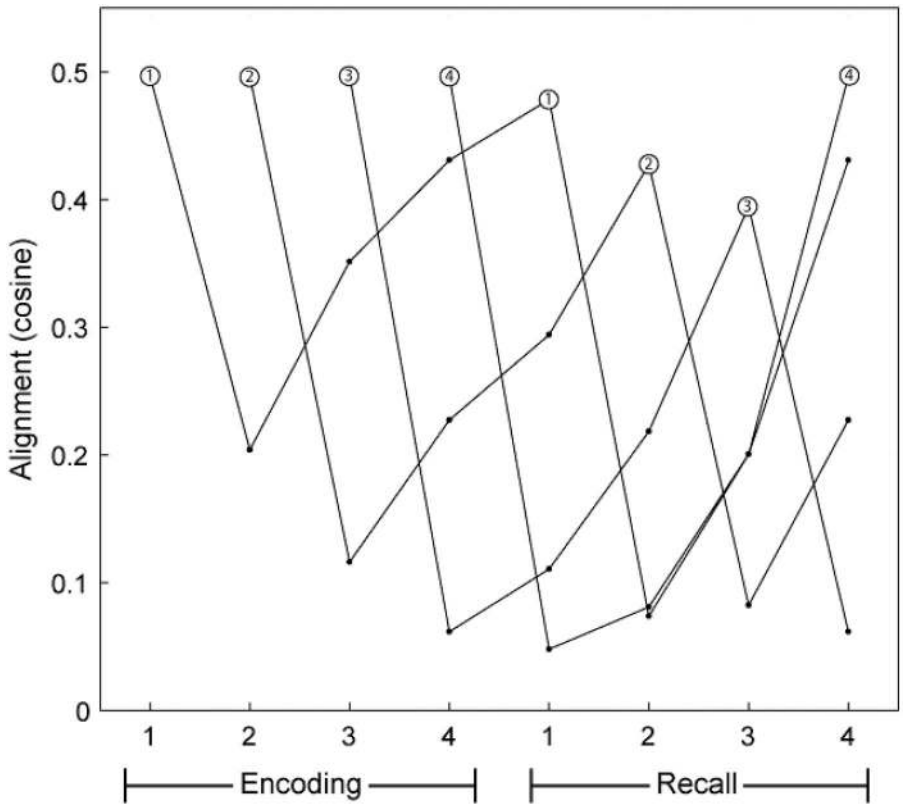

Figure 5. Mean cosine of angle between element vectors and relevant hidden-to-output weight vectors. Each time series is based on vectors representing elements at a particular list position, as indicated by the numeric labels. Labels are included only on time steps in which the represented element is to be output.

The data in Figure 5 are most easily parsed by focusing on a single step of processing. Consider, for example, the first step of recall. The four points plotted here indicate the visibility of elements at list positions one, two, three and four, reading from top to bottom. Note that elements at Position 1—the position being recalled on this step—are associated with the largest cosine, that is, the highest visibility. At recall step two, the pattern has changed. Here, list Position 2 has the largest cosine. Once again it is the element being recalled that is most visible to the output layer. Element 1, no longer immediately relevant to the system's output, is represented on this step so as to be quite invisible to the output layer, with an activation vector that is almost orthogonal to the relevant output weights.

The connected data series in the figure provide an indication of how the representation of individual list elements evolves over the course of a trial, from encoding through recall. On the step in which an element is encoded, it is represented with an activation vector that renders it relatively visible to the model's output layer. On the next time step, the representation of the element is strongly transformed, so as to render it essentially invisible to the output layer. Then, over the succeeding steps, the element's representation gradually shifts, bringing it more and more into line with the relevant output weights. By the time the element is to be recalled, it is again relatively well aligned with those weights, and thus once again relatively visible to the output layer. The overall process can be visualized as an incremental rotation of the vectors that represent individual list elements. In a manner of speaking, these vectors are rotated out of view just following encoding, and then gradually rotated back into view as the time to recall them approaches.[5,6]

[5] The relatively dramatic representational shift occurring just after encoding (and after recall) is driven by feedback from the output layer, as conveyed via the output-to-hidden projection. This can be inferred from the finding that the weight vector connecting the output layer to any given hidden unit tends to correlate negatively with the weight vector connecting that hidden unit back to the output layer. Given this role, the feedback projection from the output layer can be understood as paralleling the "competitive filter" in the competitive queuing model of Houghton (1990), as well as other mechanisms for post-output suppression. Note that the output-to-hidden projection was included in the model to implement the assumption that internal representations of serial order are influenced by feedback from output systems. Further simulations indicated that that the network can be successfully trained without this projection, resulting in patterns of performance similar to those observed when the projection is present. Under these circumstances, the hidden-to-hidden weight matrix appears to assume the function served by the output-to-hidden weights. In both versions of the network, the hidden-to-hidden weights are responsible for driving the smaller, stepwise transformations that occur over the remaining steps of recall.

[6] To say that element representations are rotated through representational space implies that their magnitudes remain constant. Further analysis indicated that this is, in fact, true for the model. With the one exception of the encoding step, in which element vectors tended to be relatively large, the magnitude of element vectors remained essentially constant over sub-

Factors Underlying Errors

To this point, we have spoken of element vectors as if they are fixed and invariant. However, as discussed in the General Method section, the internal representations in the present model are in fact subject to some degree of variability. This means that each item–position conjunction, rather than being represented by a fixed pattern, instead maps to a probability distribution over the space of possible activation patterns. Inevitably, the distributions for different item–position conjunctions will overlap. This, in turn, makes any given pattern of activation inherently ambiguous. Any pattern arising within the model's hidden layer could potentially have been induced by any of a set of item–position conjunctions.

The way that the model responds to such ambiguity reflects a fundamental property of neural networks. Under a standard set of conditions—all of them met by the present model—neural networks approximate maximum a posteriori or Bayesian classifiers (see General Method section; also see Bishop, 1995; McClelland, 1998; Rumelhart et al., 1996). Put simply, when faced with a novel or ambiguous input pattern, such networks produce the response that is most likely to be correct, given the pattern's similarities to the set of patterns encountered during training. In keeping with this principle, the present model, when faced with an ambiguous internal representation, interprets it as reflecting the input that is most likely to have generated it. Although this response policy often results in correct outputs, it is by definition probabilistic, and therefore sometimes results in errors.

An important corollary of these considerations, which figures prominently in succeeding analyses, is that errors tend to involve confusions between items and positions that are represented similarly. As we have just noted, the variability that is present in the model means that any element, that is, any conjunction of item and position, maps to a probability distribution over representational space. When two elements are represented similarly, their distributions will tend to be relatively highly overlapping, making it comparatively easy for one element to be mistaken for the other.

Simulation 1: Basic Behavioral Phenomena

Having established some general points concerning functioning of the trained model, we turn now to the details of our first simulation study. The target empirical phenomena for this simulation were a set of fundamental behavioral observations in the domain of ISR, concerning effects of list length, primacy and recency, transposition gradients, repetitions, and relative errors.

Benchmark Phenomena

Effect of List Length

One highly consistent finding across studies of ISR concerns the relation between list length and overall recall accuracy. As illustrated in Figure 6, based on data from Crannell and Parrish (1957), the proportion of lists recalled perfectly falls as list length increases, generally following a sigmoidal pattern.

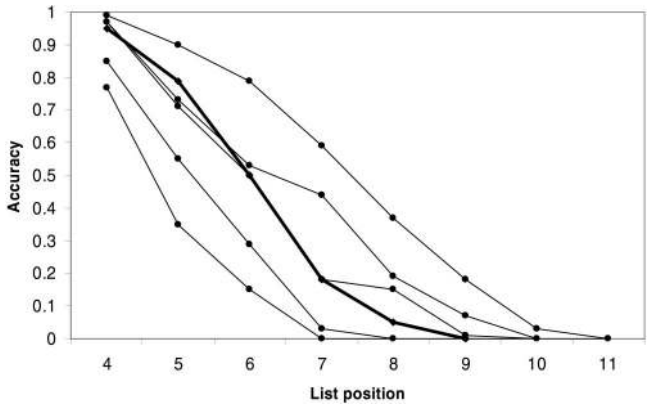

Figure 6. Relationship between sequence length and recall accuracy. Empirical data (light traces) are from Crannell and Parrish (1957). From left to right, the data series relate to recall for words (unrestricted stimulus set), words (restricted set), letters (unrestricted), letters (restricted), and digits. The heavy trace shows performance for the model in Simulation 1. From "A Comparison of Immediate Memory Span for Digits, Letters, and Words," by C. W. Crannell and J. M. Parrish, 1957, *Journal of Psychology, 44,* p. 323. Published by Heldref Publications. Copyright 1957 by the Helen Dwight Reid Educational Foundation. Adapted with permission.

Primacy, Recency, and Transposition Gradients

Some further key aspects of human serial recall become evident when accuracy is measured at the level of individual list positions. As shown in Figure 7 (left), based on Henson et al. (1998), this typically reveals a recall advantage for items toward the beginning of the list (the primacy effect), and a weaker advantage for the last one or two items in the list (the recency effect; see, e.g., Jahnke, 1963; Jahnke, 1965). Moreover, when items are recalled incorrectly, it is less often the case that they have been omitted from the list entirely than that they have been recalled in the wrong position. That is, recall tends to be better for item information than for order information (Bjork & Healy, 1974). This is reflected in Figure 7, which shows the proportion of trials on which items from each input position (i.e., position within the presented list) are recalled at each output position. As the figure makes clear, when items are relocated, there is a tendency to recall them at positions near their original position. This tendency, which Henson (1996) has called the "locality constraint," can also be visualized as a transposition gradient, as shown in Figure 8 (left), based on data from McCormack, Brown, and Vousden (2000). Figure 8 shows, further, that transposition gradient for children was less steep than that for adults. That is, children had a tendency to relocate items further from their original positions than adults.

Repetitions

Another benchmark empirical phenomenon concerns errors of item repetition. When there are few or no repeats in the presented lists, repetition errors tend to be infrequent and, more informatively, the positions of the repeated item tend to be widely separated. This regularity, which Henson (1996) has dubbed the "repetition constraint," has suggested to some researchers that there exists a special mechanism for transiently inhibiting item representations, once the associated response has been produced (Vousden & Brown, 1998).

sequent steps of processing. Element vectors for items occupying different list positions did not consistently differ in magnitude. For recent computational work demonstrating how a process of vector rotation can support encoding of serial order, see White et al. (2004).

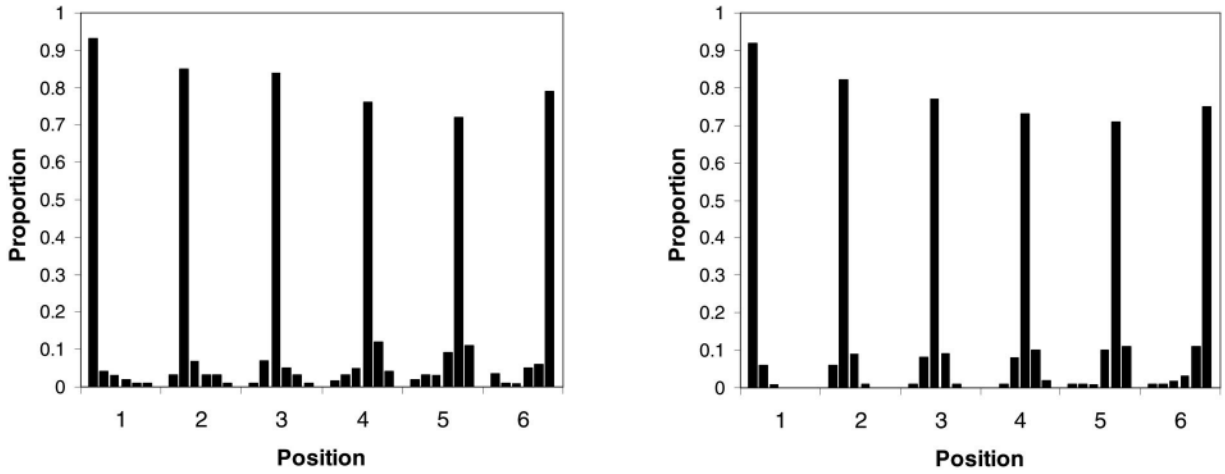

Figure 7. Left: Empirical data from Henson (1998). Each bar indicates the proportion of items from a specific input position that appeared at a specific position at recall. The *x*-axis indicates position within the target list. Within each cluster of six bars, the first bar pertains to recall position one, the second to recall position two, and so forth. Right: Performance of the model in Simulation 1. Left panel from "Short-Term Memory for Serial Order: The Start–End Model," by R. N. A. Henson, 1998, *Cognitive Psychology, 36,* p. 89. Copyright 1998 by Elsevier. Adapted with permission.

Relative Errors and Fill-In

A final benchmark finding is the fact that relative errors do not occur at levels above chance. As discussed in the introduction, Henson (1996; Henson et al., 1996) evaluated the frequency of these errors as a proportion of adjacent transpositions. His claim was that any chaining model would, of necessity, predict a ratio greater than 20% for six-item sequences. In a large-scale empirical study, the proportion of relative errors was found to be lower than this (at least, the observed proportion did not differ significantly from 20%). A closely related observation involves what Page and Norris (1998) described as *fill-in*. This refers to the finding that when an item is displaced because of a transposition error, that item tends to be recalled in the next position. Thus, if recall of the sequence *ABCDE* were to begin *AC*, the next item recalled would tend to be *B*. Henson (1996) examined responses following initial errors in which an item was recalled one step too early and found that fill-in errors accounted for 53% of such responses. In contrast, only 21% of responses involved following the first incorrect item with the item that followed it in the target list (*ACD. . .*). Henson (1996) argued that this finding was incompatible with chaining models, as well as with some context-based models, including that of Burgess and Hitch (1992).

Method

The model contained 27 units in both input and output layers. The first 26 of these were used each to represent an individual English letter. The

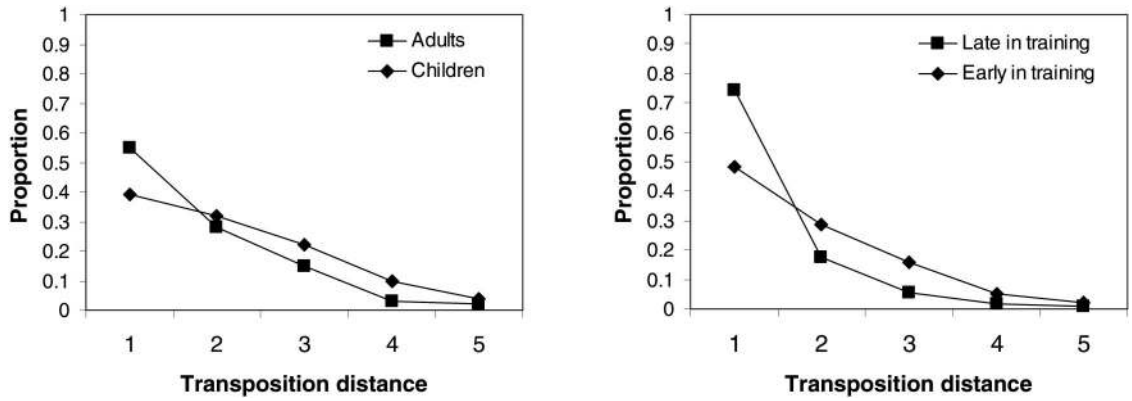

Figure 8. Left: Empirical data from McCormack, Brown, and Vousden (2000). "Transposition distance" refers to the distance, in either direction, between an item's position in the target list and its position at recall. The *y*-axis indicates the proportion of all target items recalled at the specified transposition distance. Right: Performance of the model in Simulation 2. Left panel from "Children's Serial Recall Errors: Implications for Theories of Short-Term Memory Development," by T. McCormack, G. D. A. Brown, and J. I. Vousden, 2000, *Journal of Experimental Child Psychology, 76,* p. 237. Copyright 2000 by Elsevier. Adapted with permission.

remaining unit in the input layer coded for the recall cue, and the final unit in the output layer was used to represent the end-of-list response. The model was trained on lists ranging in length from one to nine. Each list was composed of a randomly selected set of letters, the only constraint being that repeats were forbidden. Training continued until overall accuracy levels reached levels reported in the benchmark empirical studies. Benchmark data pertaining to primacy and recency, transposition gradients, repetitions, and relative errors all derived from Experiment 1 of Henson (1996; further reported in Henson et al., 1996). Overall accuracy for lists of six nonconfusable letters in this data set was 58%, and this provided the stopping criterion for our simulations. This level of accuracy was reached following 114,444 training cycles. It should be noted that this duration of training meant that the network was exposed to less than 0.0001% of all possible lists, and less than 0.07% of lists of six elements.

Results

Effect of List Length

As shown in Figure 6 (heavy trace), the proportion of trials recalled perfectly by the model fell with list length, with an overall shape comporting well with the data reported by Crannell and Parrish (1957) for arbitrary letter lists (*RMSE* = 0.053).[7] It should be noted that this is not a simple frequency effect, because lists of all lengths were presented equally often.

Primacy, Recency, and Transposition Gradients

Figure 7 (right) plots the model's accuracy on six-item lists, evaluated separately for each position. The curve clearly reflects a primacy effect, with accuracy being highest for items at the head of the list, and a smaller recency effect, benefiting the last item in the list (*RMSE* = 0.038, with respect to corresponding data from Henson, 1996). The same figure also indicates the proportion of trials on which items from each input position appeared at each output position. When items were not recalled correctly, it was most often the case that they were recalled in an incorrect position, rather than being omitted entirely. Thus, item memory was superior to order memory, as observed empirically. The model's performance also fit well with Henson's (1996) locality constraint. As is evident in the figure, when items were recalled in the wrong position, there was a tendency for them to appear near to their input position. *RMSE* for the entire data set plotted in Figure 7, with respect to the empirical data from Henson (1996), is 0.025. The pattern is illustrated in another way in Figure 8 (right), in the form of a transposition gradient comparable to the one observed empirically by McCormack et al. (2000). As also illustrated here, the model generated a broader transposition gradient (*RMSE* = 0.086) when tested at an earlier point in training (42,222 cycles), when its overall performance was comparable to that of the children in the McCormack et al. study.

Repetitions

As in the empirical data, few repetition errors were observed in the model's performance. Henson (1996) reported a rate of 1.6 repetition errors per transposition error. The comparable value, based on the model's performance, was 0.7.[8] In the model, as in the empirical data, repetitions tended to span several intervening items. In the empirical study of Henson (1996; Henson et al., 1996), the average distance between repeated list items was 3.4; in the model, 3.56.

Relative Errors and Fill-In

In the model's performance, the ratio of relative errors to adjacent transpositions, as defined earlier, was 12%. Across training runs, at comparable levels of training, this value was never observed to approach 20%. Fill-in was evaluated following the procedure followed by Henson (1996), pooling as in that study across sequences of 7, 8, and 9 elements. As in the empirical data reported Henson (1996), in cases in which the model recalled at position i an item belonging at position $i + 1$, the next response was more likely to be the item belonging at position i (a fill-in error) than the item belonging at position $i + 2$ (63% vs. 17% of relevant responses).

Discussion

In this first simulation, the model was found to generate behavior fitting with several key aspects of human behavior in the domain of ISR. These included the basic relationship between list length and accuracy, the pattern of primacy and recency commonly observed in positional recall curves, the tendency for transposition errors to cover short distances, and the tendency for repetition errors to span relatively large distances. A particularly important finding was that the model produced far fewer relative errors than would be expected of any system relying on chaining to perform ISR.

Each of these aspects of the model's behavior can be explained in terms of the basic principles laid out in the Initial Analyses section. Consider first the relationship between list length and recall accuracy. As established in Initial Analyses, the model's internal representations code for multiple list elements by superposition. As also established earlier, the model's internal representations are subject to a degree of random variability, a factor that creates the conditions for errors. These two aspects of the model are jointly responsible the list length effect. As list length increases, the number of list elements that must be concurrently represented in the hidden layer rises (a fact, e.g., evident in an increase in overall hidden layer activation with increasing list length). In the presence of random variability, this in turn makes it more difficult to analyze the model's internal representation into its element-specific components. As the number of list items increases, so does the

[7] Because the empirical data are based on averages across subjects, the underlying curves for individual subjects can be assumed to have been at least slightly more sharply inflected. Happily, this seems to be true of the curve yielded by the model.

[8] The fact that the model displayed fewer repetition errors than observed empirically (at least in the benchmark study) is explained by the fact that the model was trained on material in which repetitions never occurred. The frequency of repetitions during training influences the model's tendency to make repetition errors during recall. Thus, a higher repetition rate could have been expected, had repetitions occurred occasionally in the training set.

ambiguity of the model's internal representation, and the probability of an error rises accordingly.[9]

The model's reproduction of standard transposition gradients stems from the fact that errors in the model are most likely to involve confusions between elements that are represented similarly. As illustrated in Figure 4, positions close to one another tend to be represented more similarly than positions further apart. As a result, when an item is recalled in the wrong position, it is more likely to be recalled near its input position than distant from it. By contrast, repetition errors are rare and involve greater movement because, once an element is recalled, its representation is rotated "out of view" of the output layer (see Figure 5). This reduces the probability that items will be incorrectly repeated at short delays.

Related principles underlie the primacy and recency effects. Note that elements at both the beginning and the end of a list have fewer positional near neighbors than items toward the middle. This makes it relatively unlikely that the positions of elements near the list boundaries will be mistaken for other, similarly represented positions. In this sense, both primacy and recency are, in part, edge effects. However, there is also another reason for these effects, which has to do with the number of list elements held in memory at any given time. Consider that when the first element in a list is encoded, there are no other elements yet represented in the hidden layer. When the second item is encoded, there is only one other element represented there. With each successive element encoded, the number of elements already in memory continues to increase. This provides a partial explanation for the primacy effect. As noted a moment ago, the more items held in memory at any given time, the more difficult the overall representation becomes to decode. This means that items at early list positions have an advantage, because during the overall period from encoding to recall they share the hidden layer with relatively few other elements. A related principle contributes to the recency effect. Further analysis along the lines reported in the Initial Analyses section indicates that the representations of elements already recalled are quite distinct from those for items not yet recalled. Thus, as recall nears the end of the list, the representations of elements remaining to be recalled are less and less likely to become confused with other elements being represented in the hidden layer. This protective effect is small, compared with the relative isolation of early list items at encoding, explaining the asymmetry between primacy and recency.

The infrequency of relative errors stems from the fact that individual elements are represented independently within the hidden layer. As observed earlier, the way that any given list element is represented is not affected by the identity of the other elements in the target sequence. As a result, if the representation of one list element becomes disrupted, causing the relevant item to be recalled at the wrong list position, it does not follow that the element's successor will be similarly displaced. Thus, relative errors do not occur at high rates, as would be expected from a system in which element representations were interdependent.

Simulation 2: Effects of Interitem Similarity

A number of important empirical phenomena in ISR concern situations involving confusable items, such as letters with phonologically similar names. The present simulation extended the approach taken in Simulation 1 by introducing item representations that varied in their degree of overlap, making it possible to apply the model to several benchmark phenomena involving interitem similarity.

Benchmark Phenomena

Recall Accuracy and Transposition Gradients

As originally shown by Conrad and Hull (1964), lists of confusable items are recalled less accurately than identical length lists of nonconfusable items. Henson et al. (1996) have referred to this behavioral finding as the "similarity constraint." Representative empirical data, from an experiment by Baddeley (1968), are diagramed in Figure 9 (left; upper- and lowermost data series). It has also been shown that transpositions in lists of confusable items tend to span a slightly larger distance, on average, than transpositions in nonconfusable lists. This pattern is evident in the transposition gradients illustrated in Figure 10 (left), drawn from Henson (1996).

Sawtooth Pattern for Alternating Lists

An extremely important benchmark finding is the pattern of behavior on lists in which confusable and nonconfusable items alternate (Baddeley, 1968). The critical finding here is that nonconfusable items in alternating lists are recalled just as accurately as items in the same positions in pure-nonconfusable lists (see Figure 9). As explained earlier, this finding essentially rules out chaining-based mechanisms for ISR.

Method

The approach was identical to that taken in Simulation 1, with the exception that list items were encoded in a fashion that allowed for two levels of interitem similarity. Specifically, each item was represented in terms of two features. The input layer (and output layer) contained two groups of units, one representing values of Feature 1 (36 units), the other values of Feature 2 (6 units). Each item was represented by activating one unit in each of the feature groups so that each Feature 1 unit was unique to a particular item, and each Feature 2 unit was shared by six items. Thus, every item overlapped with 5 other items (on Feature 2) and did not overlap with the remaining 30 items. Overlapping items were used to represent confusable items; nonoverlapping items were used to represent nonconfusable ones.

The network was trained on randomly constructed lists (without repeats), ranging in length from one to six. Items were selected from the overall set of 36 without regard to interitem similarity; thus, training lists included

[9] A separate but related question about the model concerns its capacity or span. What determines the absolute accuracy of the model's performance at specific list lengths? Although we have not systematically explored this issue, some general comments can be made. Because errors in the model result from representational degradation or variability, any factor that affects the frequency and severity of such degradation will impact the model's span. Factors that increase the variability of representations (e.g., activation noise, learning rate, corpus size) will tend to reduce span, and factors that lead to increased separation between the representations of different item–position conjunctions (e.g., number of hidden units, training time) will tend to increase span. It is interesting to speculate that the particular solution to the ISR task that the model finds, as detailed in the Initial Analyses section, may place bounds on the degree to which different element representations can be separated within representational space. However, this cannot be established on the basis of analyses conducted to date.

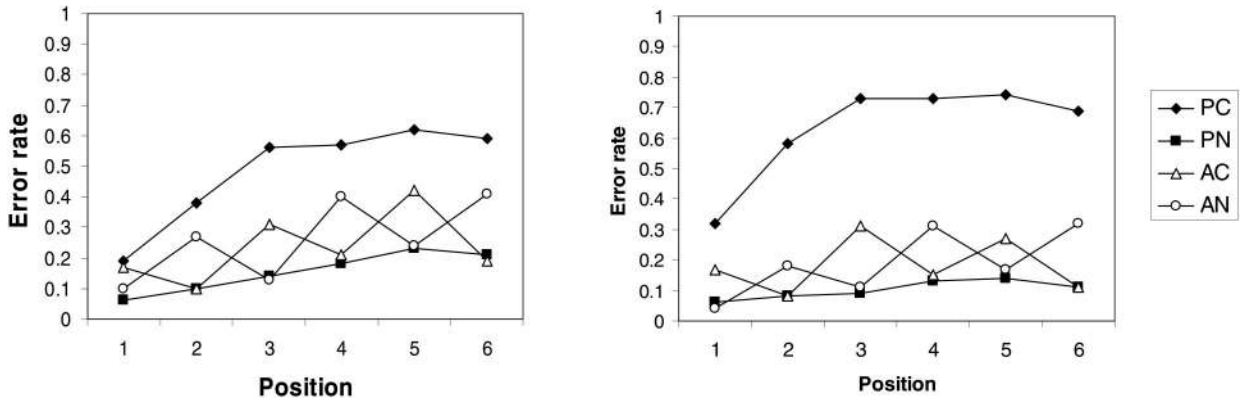

Figure 9. Left: Empirical data from Baddeley (1968), comparing performance lists of confusable items (pure-confusable lists; PC), pure-nonconfusable lists (PN), and alternating lists, beginning with either confusable (AC) or nonconfusable (AN) items. Right: Performance of the model in Simulation 2. Left panel from "How Does Acoustic Similarity Influence Short-Term Memory?", by A. D. Baddeley, 1968, *Quarterly Journal of Experimental Psychology, 20,* p. 260. Copyright 2005 by the Experimental Psychology Society (www.psypress.co.uk/journals.asp). Adapted with permission.

arbitrary mixtures of confusable and nonconfusable items. In simulating the transposition gradient data from Henson (1996), training continued until accuracies for pure-confusable and pure-nonconfusable lists of six elements surpassed 0.39, their average in the empirical data (316,666 cycles, covering less than 0.025% of all possible six-item lists). In simulating Baddeley (1968), training continued until the proportion of items recalled in the correct position, for pure-confusable and pure-nonconfusable lists of six elements, averaged 0.34 (202,000 cycles, covering less than 0.020% of possible lists).

Testing was with lists of six items, using pure-confusable, pure-nonconfusable, or alternating lists as appropriate to the corresponding experimental condition. During test runs, the model's output was determined by using an item-based winner-take-all approach. Here, the pattern of activity over all output units, including both feature sets, was compared against binary representations of each of the 36 items, and the binary pattern most closely matching the activation pattern (largest dot product) was selected as the model's output.

Results

Recall Accuracy and Transposition Gradients

In line with empirical observations, the model performed better on nonconfusable lists than confusable lists (Figure 9, right) and made longer-distance transposition errors for confusable than nonconfusable lists (Figure 10, right; $RMSE = 0.076$). In both cases, the effect of confusability was stronger in the model than in the empirical data. However, this discrepancy is a direct consequence of the relative similarity among confusable and nonconfusable items. For simplicity, we used item representations that involved only two levels of overlap, 0% and 50%; intermediate values would have yielded a smaller similarity effect. The same comment applies to the other effects tested.

Sawtooth Pattern for Alternating Lists

Performance of the model on lists alternating between confusable and nonconfusable items yielded the typical sawtooth accuracy curve (Figure 9, right). As in Baddeley (1968) and Henson (1998), performance on nonconfusable items in alternating lists matched that for items in the same positions in pure-nonconfusable lists (RMSE for overall data set shown in the figure was 0.099).[10] The finding remains unchanged if error rates for each position are computed only on trials in which no error has yet occurred, a step recommended by Henson et al. (1996) for technical reasons.

Discussion

In the present simulation, the model reproduced several empirically observed effects of interitem similarity on serial recall. With regard to the effect of confusability on accuracy, as has been noted, the model's errors are most likely to involve confusions between item–position conjunctions that are represented similarly. Because confusable items are associated with relatively similar internal representations (see Figure 4), they are more likely to be confused for one another, explaining the higher error rate associated with list of confusable items.

The model's reproduction of the pattern reported by Baddeley (1968), for alternating lists, can be understood in the same terms as the low incidence of relative errors observed in Simulation 1. Because list elements are represented independently, there is no tendency for the transposition of one element to induce a transposition of its successor, as would occur in a system based on chaining. Thus, when exchanges occur between confusable items in alternating lists, this has no particular impact on the recall of the intervening nonconfusable items.

Taken together, the results of Simulations 1 and 2 demonstrate the ability of the present model to account for a set of empirical

[10] In a similar task context, Farrell and Lewandowsky (2003) found, contrary to Baddeley (1968), that recall for nonconfusable items in alternating lists was slightly superior to items at the same positions in pure-nonconfusable lists. The same pattern arises in the model if nonconfusable items are represented as being more similar to one another than they are to the confusable items. Assuming this pattern of interitem similarity appears consistent with the interpretation Farrell and Lewandowsky offered for their empirical findings.

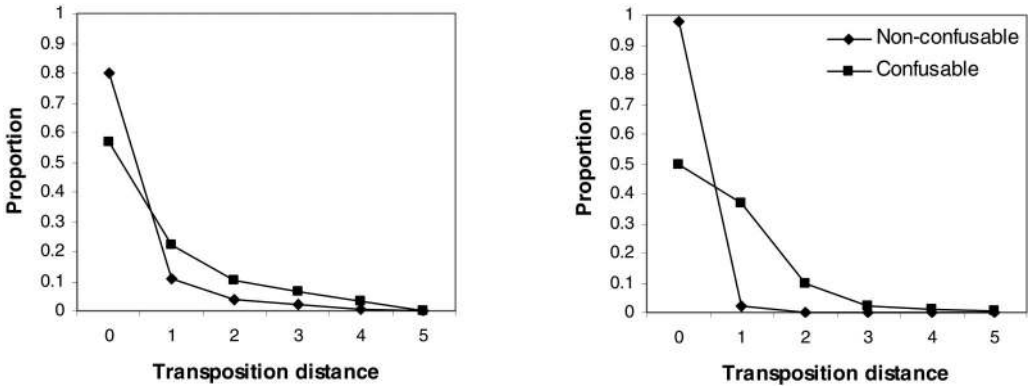

Figure 10. Left: Empirical data from Henson (1996). "Transposition distance" refers to the distance, in either direction, between an item's position in the target list and its position at recall (0 corresponds to correct recall). The *y*-axis indicates the proportion of all target items recalled at the specified transposition distance. Confusable, nonconfusable = acoustically confusable and nonconfusable items. Right: Performance of the model in Simulation 2. Left panel from *Short-Term Memory for Serial Order,* by R. N. A. Henson, 1996, unpublished doctoral dissertation, MRC Applied Psychology Unit, University of Cambridge, England, p. 33. Copyright 1996 by R. N. A. Henson. Adapted with permission.

phenomena that are widely accepted as benchmarks in the domain of ISR. We now turn to a set of behavioral phenomena that have received considerably less attention from theorists. Specifically, we focus on observations concerning the role of domain-specific background knowledge in ISR. Such effects present an important, although rarely acknowledged, challenge to models of ISR. As discussed earlier, recent models have justifiably avoided any reliance on item-to-item associations. However, when one turns to effects of background knowledge, item-to-item transitions (e.g., transitions from one letter to the next or one phoneme to the next) suddenly appear quite important to recall performance. Thus, the empirical data impose seemingly incompatible constraints on models of serial recall, requiring an insensitivity to item-to-item transitions in one context and a definite sensitivity to such transitions in another. The previous simulations demonstrated that the present model complies with the first of these constraints, by showing that the model does not operate through chaining. In Simulations 3 and 4, we present results demonstrating that the model complies with the second constraint as well, displaying a sensitivity to domain-specific regularities of sequential structure.

Simulation 3: The Bigram Frequency Effect

As introduced earlier, numerous studies have demonstrated that the mechanisms underlying serial recall are sensitive to regularities of sequential structure, when these are present in the material's source domain. In particular, several studies have demonstrated that recall is better for lists that fit well with familiar sequencing constraints than for lists that violate those constraints. Perhaps the clearest, and certainly the most replicated, finding concerning serial recall in a structured domain is the bigram frequency effect. Here, as introduced earlier, recall is better for letter strings containing bigrams that appear with relatively high frequency in English than for strings containing low-frequency bigrams (Baddeley, 1964; Kantowitz et al., 1972; Mayzner & Schoenberg, 1964). The present simulation tested whether the model shows the same sensitivity to domain structure.

Benchmark Phenomena

Although the bigram frequency effect has been reported by others (Mayzner & Schoenberg, 1964), we adopted, as benchmarks for modeling, data reported by Baddeley (1964) and by Kantowitz et al. (1972). The first of these studies involved presentation of lists reflecting the bigram frequency structure of English and also lists constrained only by the individual letter frequencies of the language. As shown in Figure 11, recall was superior for the former group of stimuli. Kantowitz et al. (1972) presented nine-item lists, each a permutation of a fixed set of nine consonants. Lists were divided into two groups, one with higher summed bigram frequencies than the other. As shown in Figure 12 (left), recall was again better for high bigram frequency lists. The data conveyed an additional detail, namely that the bigram frequency effect impacted performance on the list-initial item less than later items.

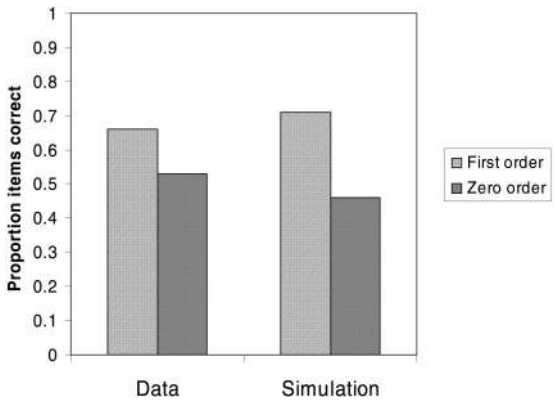

Figure 11. The bigram frequency effect. Data = empirical data from Baddeley (1964); Simulation = performance of the model in Simulation 3; first order = letter sequences generated on the basis of the letter-transition probabilities of English; zero order = letter sequences reflecting individual letter frequencies, but arbitrarily sequenced.

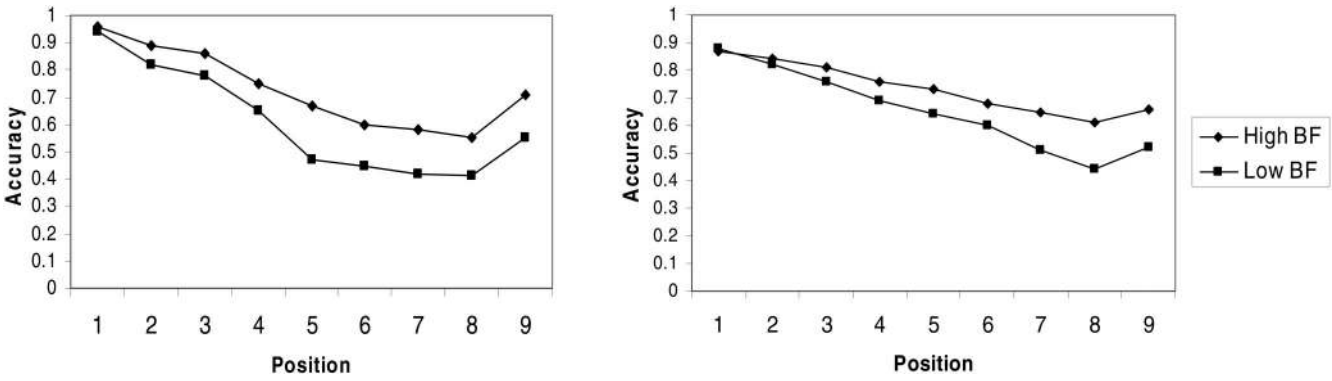

Figure 12. Left: Empirical data from Kantowitz et al. (1972), for high and low bigram-frequency (BF) lists. Right: Performance of the model in Simulation 3. Left panel from "Encoding and Immediate Serial Recall of Consonant Strings," by B. H. Kantowitz, P. A. Ornstein, and M. Schwartz, 1972, *Journal of Experimental Psychology, 93,* p. 106. Copyright 1972 by the American Psychological Association. Adapted with permission.

Method

The model architecture, representations, and general training and testing procedures were identical to those used in Simulation 1. The only change was that the lists presented, rather than being randomly constructed, were generated according to the first-order (letter-to-letter) transition probabilities of English. These were calculated on the basis of a corpus of text drawn from the *Wall Street Journal* (see Marcus, Santorini, & Marcinkiewicz, 1993). The first letter in each training list was selected on the basis of individual letter frequency.

Simulation 3A

Baddeley (1964) compared performance on eight-item lists of letters, reflecting the bigram frequency structure of English with that on lists constrained only by the individual letter frequencies of the language. In Simulation 3A, the network was trained on lists that were one to eight items in length and performance was evaluated in terms of the proportion of items recalled in the correct position. Training was halted at 187,500 cycles, when accuracies for zero and first-order lists of eight items surpassed 0.59, as in the empirical study. The model was then tested on two stimulus sets, one composed of lists generated in the same manner as those used during training (first-order lists), the other based only on individual letter frequencies (zero-order lists).

Simulation 3B

Kantowitz et al. (1972) presented nine-item lists, each a permutation of a fixed set of nine consonants. Lists were divided into two groups, one with higher summed bigram frequencies than the other. In Simulation 3B, the network was trained with lists one to nine items long, based on the same first-order transition probabilities as before, but including only the letters *C, D, F*, *H*, *L*, *N*, *R*, *S*, and *T*, as in the empirical study. Training was halted at 24,000 cycles, when the positional accuracy—proportion of items recalled in the correct position—reached 0.67 when averaged between high and low bigram frequency lists, as in the empirical study. Testing was conducted using lists constructed in the same manner as those used during training. For analysis, lists were divided into two groups on the basis of a median split on summed bigram frequency. Performance was compared between these high and low bigram frequency groups.

Results and Discussion

Simulation 3A reproduced the central finding of Baddeley (1964), in that recall was better for first-order lists than zero-order lists (Figure 11, right). Simulation 3B yielded behavior similar to that reported in Kantowitz et al. (1972), in that responses were more accurate for high than low bigram frequency lists (*RMSE* = 0.075; Figure 12, right). Moreover, as in the empirical study, the effect was stronger late in the list than at early list positions. These results confirm that the model's recall performance, like that of human subjects, depends on the degree to which the structure of target lists fits with the constraints governing previously encountered sequences. Although the model's basic mechanism for serial recall is not based on chaining, the model nonetheless shows a sensitivity to item-to-item transition probabilities.

Why does the model display such sensitivity? Like many other aspects of the model's performance, this one is connected to the presence of variability in the model's internal representations. As pointed out earlier, this variability means that the occurrence of any item–position conjunction in the target list maps not to a single pattern of hidden-unit activation, but instead to a probability distribution over possible activation patterns. This, in turn, makes any particular pattern of hidden-unit activation intrinsically ambiguous. Faced with such ambiguity, the model behaves like a Bayesian classifier, responding with the item that is most likely to have generated the pattern, on the basis of the pattern's similarities to those encountered during training (see the General Methods and Initial Analyses sections). In discussing this probabilistic decoding process previously, we have focused on representations of individual list items. However, note that precisely the same account extends to representations of multiple elements.

For illustration, assume the model is presented with a target list containing the low-frequency bigram *KC*. Once these two elements are encoded, they will continue to be represented together, through superposition, within the model's developing internal representation.[11] Note that because the model represents adjacent positions

[11] We conducted an analysis of the hidden representations arising in the bigram frequency model, using the same procedure described in the Appendix. Somewhat to our surprise, we found that here, once again, the encoding could be understood as a superposition of essentially independent element vectors. This highlights the point that the model can show a sensitivity to interitem relationships without such relationships affecting its sequence encodings. Having said this, we nonetheless suspect that if the

similarly (see the Initial Analyses section), the representation of *KC* will resemble that for the higher frequency bigram *CK*. To put it more precisely, the probability distributions for these bigrams, within representational space, will be relatively highly overlapping. This means, in turn, that many representations of *KC* could also plausibly have been induced by *CK*. Once again, when faced with such ambiguity, the model produces the response that is most likely to be correct, given the resemblances between its current internal representation and those encountered during training. A critical aspect of this computation, which is again generic to neural networks of the kind we are studying, is a sensitivity to frequency. The response the model selects is influenced by the prior probability of candidate responses. Specifically, the selection process is biased toward responses that occurred relatively frequently as targets during training. Thus, to return to our example, when faced with a pattern of activation that is equally likely to represent *CK* and *KC*, the model will respond with *CK*, because this bigram was encountered more frequently during training than *KC*.

Although this account is informal (see Botvinick, 2005, for a more explicit, mathematical exposition), it explains why the model shows higher accuracy on high-bigram frequency sequences. In decoding variable, and therefore ambiguous, internal representations, the model's outputs are biased toward sequences with a high prior probability. This property of the model supports correct responding on high-bigram frequency lists and undermines its recall of low-frequency bigram lists.

Simulation 4: Serial Recall for Sequences Structured by an Artificial Grammar

The bigram frequency effect addressed in Simulation 3 is one of several findings indicating greater recall accuracy for sequences that fit with domain-specific constraints on ordering. Two additional aspects of recall for structured material were demonstrated in a recent experiment by Botvinick (2005; Botvinick & Bylsma, 2005). First, as detailed below, it was shown that recall accuracy depends not only on the overall probability of the stimulus sequence but also on the neighborhood relations of that sequence (Botvinick, 2005). Second, in addition to influencing recall accuracy, background knowledge was also found to influence the content of incorrect responses. Botvinick and Bylsma (2005) found that errors in the artificial grammar ISR task displayed a tendency toward regularization. That is, responses on incorrect trials were weighted toward high-probability sequences and, in particular, sequences higher in probability than the just-presented stimulus.

Benchmark Phenomena

In the experiment, subjects were trained to perform ISR on sequences derived from an artificial grammar. The experiment began with a multisession exposure phase, during which subjects performed ISR, followed by a testing phase in which a filled delay was added to the task, to keep recall accuracy well below ceiling. During both training and testing, each presented sequence contained the same six pseudowords (*dah, fie, poe, kay, tee,* and *noo*, presented auditorily) and any of the 720 permutations of these items could occur. However, the grammar used to generate the sequences caused some sequences to be more probable than others. Under the grammar, the six pseudowords were arbitrarily divided into two groups of three, referred to as Groups *A* and *B,* and transition probabilities were imposed such that items from Group *A* were relatively likely to be followed by items from Group *B*, and vice versa. That is, the grammar created a tendency for sequences to alternate between the two groups. As a consequence, sequences with the structure *ABABAB* (or *BABABA*) were most frequent (30% of trials), whereas sequences with the structure *AAABBB* (or *BBBAAA*) were least frequent (1.9% of trials). (In what follows, list structures should be understood as referring to their inverses as well; e.g., *AAABBB* should be understood as referring to both *AAABBB* and *BBBAAA*.) The probabilities of each of the 20 possible list structures is shown in Table 1. The probability of occurrence for each list type can be understood as reflecting its goodness-of-fit with the alternation constraint implicit in the grammar; the more violations of this soft constraint a list contains, the less probable the list is to occur. In view of this, Botvinick (2005) referred to the probability of a list type as its "goodness of fit" (with domain-specific sequencing constraints), or simply its "goodness." Analysis of recall performance during the testing phase yielded three principal findings, as detailed below.

List Goodness

Analogous to the bigram frequency effect, Botvinick (2005) found that recall accuracy was superior for lists highly consistent with domain-specific sequencing constraints than for less consistent ones (Figure 13, left).

Neighborhood Relations

Botvinick (2005) predicted that recall would be worse for sequences with high-goodness near neighbors than for stimulus sequences with equal goodness having no such neighbors. The prediction was tested by comparing recall for the stimuli with the structure *AABABB* with stimuli of types *AABBAB, ABBAAB,* and *ABAABB.* All four of these stimulus groups have the same good-

Table 1
List Structures and Probabilities

List structure	Probability
ABABAB,BABABA	.00420
ABABBA,BABAAB	.00209
ABAABA,BABBAB	.00209
AABBAB,BBAABA	.00106[a]
ABBAAB,BAABBA	.00106[a]
ABAABB,BABBAA	.00106[a]
AABABB,BBABAA	.00106[b]
ABBBAA,BAAABB	.00052
AABBBA,BBAAAB	.00052
AAABBB,BBBAAA	.00026

Note. List structures are from Botvinick (2005) and Botvinick and Bylsma (2005). Each structural category contains 72 specific stimulus lists. Probabilities shown are for specific lists.
[a] Lists with high-goodness near neighbors. [b] Lists without high-goodness near neighbors.

model were trained on more highly predictable sequences, its internal representations would be likely to involve less independent (that is, more conjunctive) item representations. The structure of the model's internal representations in structured domains is an important area for further work.

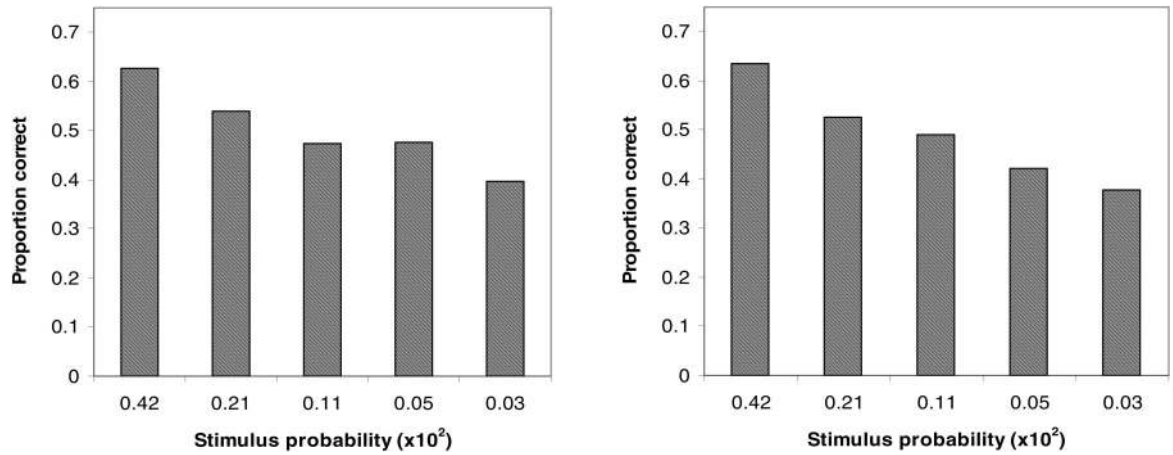

Figure 13. Left: Empirical data from the experiment of Botvinick and Bylsma (2005). Right: Performance of the model in Simulation 4. Left panel from "Regularization in Short-Term Memory for Serial Order," M. Botvinick and L. M. Bylsma, 2005, *Journal of Experimental Psychology: Learning, Memory, and Cognition, 31,* p. 354. Copyright 2005 by the American Psychological Association. Adapted with permission.

ness or probability of occurrence (see Table 1). However, the first category (*AABABB*) differs from the others in its neighborhood relations. Specifically, each of the other stimulus types can be transformed into a high-goodness *ABABAB* sequence by simply transposing two adjacent items (Items 2 and 3 for *AABBAB,* Items 3 and 4 for *ABBAAB*, and Items 4 and 5 for *ABAABB*). This is not the case for stimulus structure *AABABB.* Sequences of this type, unlike the others, do not have a very high-goodness near neighbor. Given this, the specific prediction was that recall would be better for *AABABB* sequences than for the other three categories, considered as a group. This was, in fact, the pattern observed (mean accuracy = 51% vs. 46%).

Regularization Errors

Botvinick and Bylsma (2005) compared error patterns against those of a control group, who performed ISR on the same pseudowords, but without the artificial grammar (i.e., all 720 permutations of the items were equiprobable). As compared with a baseline inferred from this control group, subjects exposed to the grammar produced error responses that were higher in goodness. In addition, they produced a higher proportion of regularizing responses, that is, responses higher in goodness than the to-be-recalled stimulus sequence (see Figure 14).

Method

The model was identical to that used in the previous simulations, except that the input and output layers contained 12-item units. Six of these were used to represent each of the pseudowords used in the experimental task (3 *A* items and 3 *B* items). During both training and testing, lists of 1–6 items were generated according to the grammar used in the empirical experiment, resulting in the list-type frequencies presented in Table 1 for 6-item lists. Training was terminated when accuracy for six-item lists reached 50% (83,333 cycles). Testing was on 6-item lists generated in the same manner. In keeping with Botvinick (2005), only responses that contained all 6 items were considered; exceptions to this requirement occurred on less than 3.4% of trials. Accuracy (proportion of lists recalled correctly) was evaluated for lists at each level of goodness or probability of occurrence (as listed in Table 1). To demonstrate the effect of neighborhood relations, we also compared the accuracy between lists of type *AABABB* and lists of type *AABBAB*, *ABBAAB*, and *ABAABB*.

The model was also trained to perform the control task from the experiment using the remaining six input and output units. As with the first set of units, each unit represented an item (pseudoword). However, sequences presented over this second set of units were arbitrarily sequenced. Training trials alternated between lists presented over the first set of six units (generated on the basis of the grammar) and lists presented over the second set (unstructured lists). Training the model on both versions of the task concurrently resulted in a larger overall training set, as compared with training the model twice, using a single set of input and output units. This was advantageous, because it increased the pressure to discover a general solution to the serial recall task, rather than an idiosyncratic one tailored to a small set of potential sequences. A more literal, but less efficient, strategy would have been to train the model on each version of the task separately but also to include, in both training runs, additional sequences involving items beyond the six pseudowords from the benchmark experiment. Computationally speaking, there is little difference between this approach and the one we took, and we assume that comparable results would have been obtained.

For the present simulation, the noise level (variance) added to hidden units at the end of each processing cycle was 0.05 (for reasons of implementational convenience, noise was injected only at test, not during training). Analyses were based on a sample of 100,000 trials, assuring an adequate sample for low-frequency sequences.

Results

List Goodness

Recall accuracy varied monotonically with list goodness. As in human performance, recall in the model was better for sequences with higher goodness (Figure 13, right; *RMSE* = 0.027).

Neighborhood Relations

The model reproduced the effect of neighborhood relations observed by Botvinick (2005). Specifically, recall was superior for stimuli with the structure *AABABB* than for those with the structures *AABBAB, ABBAAB,* and *ABAABB* (0.54% vs. 0.44%).

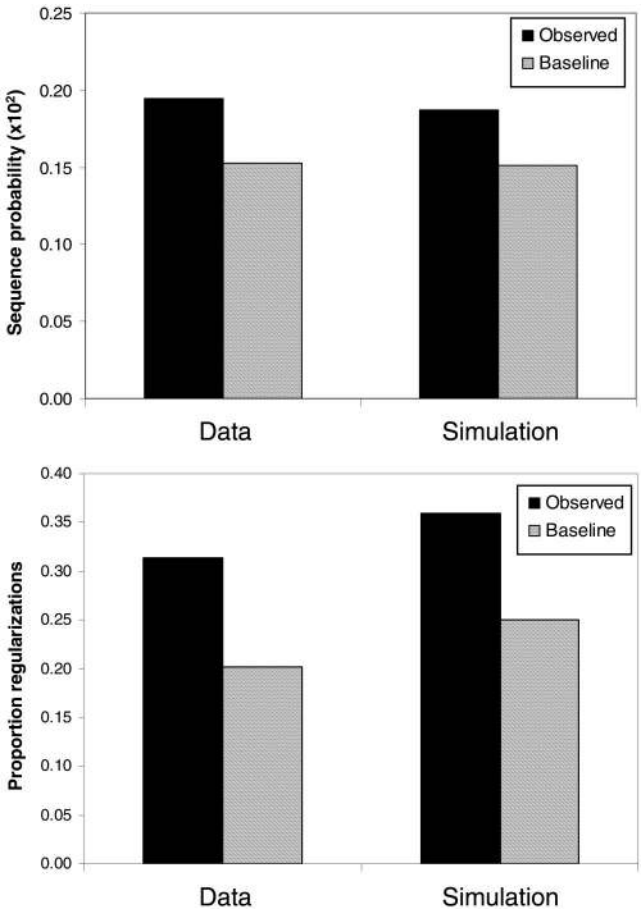

Figure 14. Empirical data from Botvinick and Bylsma (2005) and performance of the model in Simulation 4. Following the terminology used by Botvinick and Bylsma, "observed" refers to performance on sequences generated on the basis of the artificial grammar, following previous experience with the grammar, and "baseline" refers to the pattern of performance that would be expected if recall were not influenced by previous experience with the grammar. From "Regularization in Short-Term Memory for Serial Order," M. Botvinick and L. M. Bylsma, 2005, *Journal of Experimental Psychology: Learning, Memory, and Cognition, 31,* p. 355. Copyright 2005 by the American Psychological Association. Adapted with permission.

Regularization Errors

The regularization effect reported by Botvinick and Bylsma (2005) was also present in the performance of the model. As shown in Figure 14, the errors produced by the model were higher in goodness and included a higher proportion of regularizations, than would be expected on the basis of the model's performance on the control task.

Discussion

This simulation, like Simulation 3, considered the performance of the model in a structured domain, showing that the recall performance of the model, like that of human subjects, is influenced by prior experience with regularities of serial order. High-goodness sequences are better recalled because, faced with variable and therefore ambiguous internal representations, the model is biased toward responses associated with a high prior probability, that is, high-goodness sequences. The regularization effect is also a direct consequence of this principle, because it means that low-goodness sequences will often be reorganized at recall, yielding higher goodness sequences. The neighborhood effect arises from the fact that the model's errors are most likely to involve confusions between sequences that are represented similarly. This point interacts with the model's bias toward high-goodness outputs. All things being equal, this bias leads the model often to transform target lists into higher goodness sequences. However, this will only happen if the target list and the higher goodness list tend to be represented similarly. If the target list has no high-goodness near neighbors in representational space, regularization will be less likely to occur.[12]

The results of the simulation, taken together with the preceding findings, underline the model's ability to deal with two seemingly contradictory constraints. The empirical data provide clear evidence that serial recall is not based on item-to-item associations (chaining). Yet, recall performance is nonetheless impacted by background knowledge concerning item-to-item transition probabilities. Although our initial analyses and the results of Simulations 1 and 2 showed that the present model does not function through chaining, Simulations 3 and 4 illustrate that it is nonetheless sensitive to the fit of item-to-item transitions with domain-specific constraints.

General Discussion

In the current article, we have presented simulation results from a recurrent neural network model of ISR. Simulations 1 and 2 were aimed at establishing the basic viability of the model, by demonstrating its ability to account for a set of key benchmark empirical data. Simulation 1 showed that the model reproduces the empirically observed effect of list length on recall accuracy, the shape of the serial position curve and transposition gradients, and effects relating to item repetition and relative errors. Simulation 2 demonstrated that the model's performance, like that of human subjects, suffers when list items are confusable and that this reduction in accuracy is associated with a spread in the transposition gradient. The same simulation showed that the model reproduces the pattern of performance reported by Baddeley (1968) on lists alternating between confusable and nonconfusable items.

This last result was particularly significant, because it implies that the model does not function through chaining. The point is reinforced by the demonstration, in Simulation 1, that recurrent networks need not yield relative errors at the rates predicted for chaining models (Henson et al., 1996). The results detailed in the Initial Analyses section show directly that the model does not rely on interitem associations. It seems likely that these findings will come as a surprise to many working in the field, given that recent commentaries have routinely grouped recurrent networks with chaining models, prematurely rejecting them as viable models of serial recall (see G. Brown et al., 2000; Burgess & Hitch, 1999; Henson et al., 1996; Houghton, 1990; Houghton & Hartley, 1995)

Although the present model clearly does not simply chain, it nonetheless responds to consistent sequential relationships among

[12] For a more formal development of this same account, see Botvinick (2005).

the items to which it is exposed. This was demonstrated in Simulation 3, in which the model was shown to reproduce the empirically observed effect of bigram frequency on recall performance. The model's ability to capture this effect, without positing special mechanisms beyond those responsible for serial recall itself, sets it apart from all other existing models of ISR of which we are aware. The results of Simulation 3 were extended in Simulation 4, in which the model was applied to recent data pertaining to recall of sequences generated by an artificial grammar.

In the ensuing discussion, we compare the present account with other models of short-term memory for serial order, review some predictions of the account, and enumerate issues requiring further research.

Comparison With Other Models

At the outset of this article, we characterized our model as an implementation of one general approach to serial recall. The basic elements of this approach are (a) use of an activation-based, rather than weight-based, memory mechanism and (b) a central functional role for recurrent connectivity. Virtually all of the other models in the literature that adopt an activation- and recurrence-based approach are concerned with addressing neurobiological data rather than detailed behavioral data. It is interesting to consider the relationship between such models and our own, and we do so in a later section (Neuroscience-Based Models). However, given that our model focuses on addressing behavioral data, the more pressing comparison is between the present model and other psychologically oriented models of serial recall. As noted in the beginning of the article, the majority of such models adopts quite a different approach to the domain, which depends on transient Hebbian links between independent representations of item and position or context. In what follows, we begin with a discussion of such context-based models, then turn to other psychological models (in particular, the primacy model of Page & Norris, 1998), and finally consider some neurobiologically inspired models.

Context-Based Models

A basic difference between the model we have presented and context-based accounts (G. Brown et al., 2000; Burgess & Hitch, 1999; Henson, 1996, 1998; Hartley & Houghton, 1996; Houghton, 1990; Houghton & Hartley, 1995) involves the distinction between activation-based and weight-based forms of short-term memory. A defining aspect of the context-based framework is its dependence on transient, trial-specific links between item and context representations. In the model we have presented, the system's connection weights do not change during a single trial. Recall performance depends instead on sustained activation of relevant representations. The distinction involved here between weight-based and activation-based memory mechanisms has become fundamental in computational neuroscience. Both kinds of mechanisms have been implicated, at the neural level, in supporting memory for specific events (O'Reilly & Munakata, 2000). However, when it comes to ISR, available neuroscientific evidence points to a more central role for activation-based memory. The prefrontal cortex, a brain area widely believed to participate in short-term memory for sequences (Barone & Joseph, 1989), appears to implement an activation-based memory mechanism (Fuster, 1997; E. K. Miller & Cohen, 2001). Indeed, as noted earlier (and discussed at further length below) most serial recall models striving to account for neurophysiologic data have been based on this type of mechanism.

Of course, although changes to the connection weights in our model are not responsible for encoding trial-specific sequence information, the weights are nonetheless critical to the network's function. Rather than being used to transiently encode information about a specific sequence, as in context-based models, the weights play a central role in instantiating and updating the patterns of activation that do encode such information. One might say that the system's weights, rather than being shaped by the sequence presented on any specific trial, are shaped so as to support sequence encoding and production more generally. As demonstrated in Simulations 3 and 4, the weights are shaped, in addition, by any regularities of sequential structure that may be present in frequently processed material. The inherent sensitivity of the model to such regularities, as demonstrated in those simulations, is among its most distinctive aspects. Indeed, this may be the most critical point of contrast between the account we have presented and the context-based account now prevalent in the psychological literature. As has been acknowledged by a number of modelers pursuing the context-based paradigm (e.g., Henson, 1998), phenomena relating to domain structure present a challenge for this framework, given its strict avoidance of item-to-item associations. It should be acknowledged, though, that tentative efforts have been made to shore up this aspect of the context-based approach, and we discuss such efforts in a subsequent section.

A second basic way in which the present account differs from context-based accounts pertains to the kinds of representations each approach posits. Context-based models assume separate and independent representations of item and position (i.e., context). By the account we have presented, item and position are instead represented conjunctively, within a single distributed representation. The distinction between conjunctive versus independent representations of item and position has some important functional implications. In particular, the separation of item and position codes in context-based models has frequently been linked to the claim that serial recall involves a two-stage process: a first stage involving retrieval of position information, at which positional confusions may occur, and a second stage at which item identification occurs and at which confusions between similar (e.g., phonologically related) items may occur (Farrell & Lewandowsky, 2002; Henson, 1998; Page & Norris, 1998). The model we have presented demonstrates that one need not assume such a two-tiered process to account for human recall behavior. Within the model, similarity-based confusions between items originate at the same level of processing—indeed, within the same distributed representations—that support order memory.

The idea that item and order information are represented together may appear to conflict with certain empirical findings that have been interpreted as reflecting dissociations between item and order memory. For example, Saint-Aubin and Poirier (1999; Poirier & Saint-Aubin, 1995) showed that ISR performed on lists of words all relating to the same semantic category tend to improve item recall but not order recall. How could such a dissociation be captured by the present model? Saint-Aubin and Poirier's own analysis provides a straightforward answer. They concluded that the effect reflects a strategy whereby only words belonging to the relevant category are considered viable candidates for recall. Such

a strategy can be simulated in the present model if each input and output unit is considered to represent a word, and only category-relevant output units are permitted to become active. The result (confirmed through simulations in which this approach was taken) is to improve item recall, without affecting order recall.

Another apparent dissociation between item- and order-related performance comes from neuropsychological work. Shallice and Butterworth (1977; patient JB) and Vallar and Baddeley (1984; patient PV) both reported patients with impaired serial order memory (digit span) but who could reliably repeat single words. Burgess and Hitch (1999) addressed this finding in their model by lesioning the link from input phoneme representations to the model's core short-term memory mechanism, leaving intact a direct pathway from input to output phoneme representations. The same effect can be produced in the present model by lesioning a proportion of the model's hidden units. A small amount of damage impairs performance mainly at longer list lengths. With increasing damage, performance begins to deteriorate at shorter list lengths, and at some point, only single items can be reliably recalled. Of course, this account (like Burgess & Hitch, 1999), provides no explanation for why the relevant patients retained knowledge of word meanings, or for why at least one of them (JB) showed essentially normal spontaneous speech. Nor does it explain why these patients show relatively preserved serial recall for visually presented items. Accounting for this larger pattern of findings would clearly require a theory considerably wider in scope than the one we have put forth here.

The Primacy Model

Page and Norris (1998) proposed a model of ISR that differs fundamentally from context-based models. Here, order is encoded only on the basis of an activation gradient across item representations. The encoding process involves activating item representations in such a way that the first item in the sequence is most highly activated, with a progressive decrease in activation across the remaining items in the sequence. The recall process exploits this activation gradient, by selecting for recall the most active item. Once selected, item representations are inhibited, allowing recall to move on to the next item (for related work, see Grossberg, 1977, 1978).

This account, which its authors have named the "primacy model," shares a number of features with the account we have presented. Perhaps most important is that, unlike context-based accounts, both models are activation-based, and encode item and position information using the same units. However, whereas the primacy model encodes position information purely through the activation level of item representations, the model we have presented represents position in a richer way, exploiting a continuous, multidimensional representational space within which item and position can be represented along different dimensions.

The distinction here between unidimensional (localist) and multidimensional (distributed) representation has important consequences when it comes to addressing effects of item similarity. Like the context-based theories discussed earlier, Page and Norris (1998) proposed a two-stage account, according to which positional errors occur at a first stage of selection, which is insensitive to item similarity, with similarity-based item errors arising at a second stage. As Page and Norris themselves admit, their implementation of this two-tiered process required a number of awkward assumptions, including multiplication of the item activations at the second level by those in the first, and suppression of selected items at both levels, even if these do correspond. In defense of these apparently ad hoc aspects of their model, Page and Norris wrote

> The choice of a two-stage process to account for the effects of phonological similarity might strike some as inelegant. However, it is a choice that has been dictated by the complexities of the data. On the assumption that a one-stage model would be more parsimonious, we devoted a great deal of time to an attempt to develop a simpler account of the data. However . . . none of the one-stage models we tested were able to give a proper simulation of the data. [Our] analysis, demonstrating the necessity for two stages of suppression to model the alternating-list data, highlights the central problem faced by one-stage models. We believe these data force the use of a two-stage model. (p. 774)

The model we have presented clearly demonstrates that, in fact, a single-stage model can account for effects of item similarity, including the alternating list data to which Page and Norris (1998) refer.[13]

Another important difference between the primacy model and the model we have proposed is that the former, like context-based models, is not inherently sensitive to regularities in sequential structure. As a consequence, it is not clear how the model would account for such findings as the bigram frequency effect or the artificial grammar data discussed in the present Simulation 4. As with context-based models, it is possible that the primacy model could be augmented to make contact with such data, perhaps through the inclusion of superordinate "chunk" nodes, as in the work of C. L. Lee and Estes (1981). However, until such steps are taken, effects of domain structure must be seen as an outstanding challenge to the Page and Norris (1998) account.

Other Psychological Models

ISR, as a field of study, has produced a remarkable number of theoretical models, so many in fact that an exhaustive review simply cannot be undertaken here. Instead, we concentrate on accounts that, like our own and those provided by context-based models and the primacy model, posit a concrete mechanism for encoding and recalling serial order information.

One of the earliest efforts to specify the mechanisms supporting serial recall was by Estes (1972; see also C. L. Lee & Estes, 1977, 1981). Here, item representations are activated at encoding and undergo a cycle of suppression and reactivation during list rehearsal. This reactivation process is subject to "perturbations," giving rise to transpositions between adjacent items. This pioneering account faces difficulty with some subsequent empirical data, which call into question the central role Estes (1972) accords to

[13] It may be tempting to view the model we have put forth as two-tiered, in that the model contains, in addition to the stage of internal representation, a stage at which responses are selected in a winner-take-all fashion. It is important to emphasize, however, that these two stages do not correspond to those Page and Norris (1998) consider obligatory: A first, position-based stage that is insensitive to item similarity and a second stage that is sensitive to item similarity. In our model, the representational factors that give rise to item confusions inhere in the same level of the system in which position information is encoded.

rehearsal (Baddeley, 1986), and which challenge the theory's account of transpositions (e.g., Baddeley, 1968). Furthermore, the theory provides no explanation for item similarity effects. Nonetheless, many of the account's most promising aspects have found continued application in the primacy model of Page and Norris (1998), who portray their model as a "direct descendant" of Estes (1972).

Another influential account of serial order memory is the TODAM model of Lewandowsky and Murdock (1989; Murdock, 1997). This model shares at least one critical feature with the one we have put forth, which is that it uses a single, distributed memory representation, into which information about all list items is integrated. Despite this parallel, there are other aspects of TODAM that differentiate it strongly from the model we have presented. First, TODAM relies to a significant degree on inter-item chaining, an approach that, as we have discussed, appears to conflict with certain empirical findings. Second, TODAM displays the primacy effect only as a result of an ad hoc parameter, included essentially to produce this effect. Third, TODAM has never been used to model transposition gradients, currently considered a basic benchmark for models of ISR. Together, these and other problems have raised doubts about the viability of TODAM (Mewhort, Popham, & James, 1994; Nairne & Neath, 1994). Indeed, one of the theory's creators has recently proposed a model of serial recall that is quite different in character, to which we turn next.

Farrell and Lewandowsky (2002) put forth a model of serial recall (dubbed the "SOB model") that takes the form of an attractor network. Items, represented in the network by distributed patterns of activation, are encoded through weight changes that give the network a tendency to settle into the appropriate pattern through auto-association. Critically, the magnitude of these weight changes is modulated by a factor (Hopfield energy) that decreases with each successive list item. Recall is performed by imposing a random pattern of activation and allowing the network to settle into a stable state, which typically turns out to be the pattern corresponding to the first item presented. Following each step of recall, weight changes are made to counter those that occurred when the just-recalled item was originally encoded, and the network is again allowed to settle. The SOB model is intriguing for the novelty of its encoding and recall mechanisms, and it will be interesting to see whether it can be extended to the full set of benchmark phenomena pertaining to ISR. At the present stage of development, however, it bears several limitations. One is that it relies on strictly orthogonal item representations, making it difficult to deal with issues of item similarity. Second, it relies on a suppression process that essentially expunges already-recalled items from the system's memory, making it uncertain how the model would account for the phenomenon of rehearsal. Third, it relies in several places on novel, domain-specific, and sometimes elaborate assumptions (e.g., the modulation of encoding strength by energy), whose plausibility will depend on experimental validation. Finally, because it uses a weight-based memory mechanism, shaped anew on each trial of learning, it is not clear how the SOB model could be used to address effects of domain structure, like those to which we have called attention.

Anderson and Matessa (1997) proposed an account of serial recall based on the ACT-R production system architecture. Here, list elements are encoded by linking representations of item and position to a common "node" in memory. Recall involves activation of a production (labeled *get-next*). This consults a pointer indicating the current position of recall, to locate the node associated with the same list position. Once this node is identified, the item associated with it is accessed and recalled, and the position of the pointer is incremented. It is difficult to compare this ACT-R account with the model we have presented, or with the context-based neural network models now prevalent in the literature, because several critical aspects of the account (such as the functioning of the position pointer and the basis for similarity between position representations) are directly stipulated, without an explicit account of the underlying mechanisms or representational structure. However, the Anderson and Matessa model appears, at a fundamental level, to implement a context-based account, as we have defined this, because of its reliance on trial-specific links between item and position representations. As such, the points made in the earlier discussion of context-based models can arguably be extended to the ACT-R theory, in particular such models' lack of intrinsic sensitivity to domain structure. This being said, the Anderson and Matessa model, unlike the context-based models we have cited, can also be viewed as portraying item and position information as being stored together, as part of a single, structured representation. If viewed from this perspective, the theory has a bit more in common with the account we have put forward.

Related comments apply to the feature-based models of serial recall proposed by Nairne (1990) and by Neath (2000). Here, list elements are represented as feature vectors encoding both item identity and list position. The encoding of an entire list is described by Nairne (1990, p. 253) as a "vector of vectors," a description that resonates with our characterization of the sequence encodings arising in the present model.[14] Order errors at recall stem from perturbations in the representation of position information within the element-specific feature vectors, an account that again bears some relation to our own. Indeed, at the level of representational structure, the model we have presented appears rather close to the feature model of Nairne and of Neath. Similar comments apply to the recently proposed SIMPLE model, in which list elements are represented as points within a multidimensional similarity space, with dimensions relating to item and position or time (G. D. A. Brown, Neath, & Chater, 2005; Lewandowsky, Brown, Wright, & Nimmo, in press). A central assumption of this model is that recall errors result from interference between element representations being held concurrently in memory, an idea that also plays an important role in the model we have presented.

Notwithstanding these similarities, the present account can be distinguished from feature and SIMPLE models in several ways. First, the representations involved in the present account emerged from more basic assumptions about system architecture and task structure, rather than being directly stipulated. More important, the present account implements a concrete mechanism for generating and acting on the relevant sequence representations. A detailed consideration of this mechanism, as it bears on the process of recall, reveals differences from the algorithmic account of recall

[14] A subtle but important distinction is that the superpositional code involved in our model requires conjunctive representation of item and position. Because no such conjunctive coding is used in the feature model, the phrase "vector of vectors" must refer to some method of combining element-specific vector representations other than vector summation (superposition).

presented by Nairne (1990) and Neath (2000), a point we unpack in the following section.

Models Addressing the Role of Background Knowledge

We have emphasized the ability of the present model to account for effects of background knowledge, arguing that such phenomena present a challenge for other models of serial recall. In view of the latter claim, it is important to consider previous theories that have engaged the issue of background knowledge.

One place where this issue has been discussed is in connection with the idea of trace redintegration. Numerous theories have suggested that long-term memory representations are brought to bear in the process of recall, to help disambiguate degraded short-term traces (Hulme, Maughan, & Brown, 1991; Hulme et al., 1997; Lewandowsky, 1999; Nairne, 1990; Neath, 2000; Schweickert, 1993). Although the idea of trace redintegration does address the relation between short-term memory and background knowledge, the vast majority of relevant work has focused on the issue of item recall (considering, e.g., effects of item frequency and lexicality). In contrast, very little work has focused on the effect of background knowledge on recall for order, the issue with which we have been concerned in the present work. Gathercole et al. (1999) have extended the idea of redintegration to the domain of order, accounting for phonotactic effects on short-term sequence memory by suggesting that long-term knowledge concerning constraints on serial order is brought to bear in disambiguating degraded short-term sequence representations. A similar idea has been pursued by Botvinick (2005), which frames the idea of redintegration within a more general, Bayesian account of serial recall. At an abstract level, both this account and the one provided by Gathercole et al. (1999) fit well with the one we have presented in the current article. Indeed, we have characterized the present model's performance as involving the decoding of noisy sequence representations, with the help of long-term knowledge. However, the present model goes beyond these related accounts, in that it implements an explicit mechanism that accomplishes the posited decoding process.

We are aware of only one other published account that has been similarly explicit concerning the mechanisms through which long-term knowledge may influence recall for serial order, specifically, a study by Hartley and Houghton (1996) modeling short-term memory for nonwords (see also Glasspool & Houghton, 1997; Gupta & MacWhinney, 1997). The primary empirical phenomenon addressed in this work was the tendency of error responses to preserve the consonant–vowel structure of targets. To model this, Hartley and Houghton (1996) used a context-based model but also allowed this to interact with a "syllable template" mechanism. The effect of this mechanism was to provide top-down support, during recall, to the most appropriate subset of phonemes, given the outputs already produced (e.g., encouraging the selection of a vowel after production of a consonant). In essence, the approach was to supplement the familiar context-based mechanism with a chaining mechanism that biased response selection.

Although this model has not been applied to such phenomena as the bigram frequency effect, it is possible that it might prove sufficient for this. After all, Baddeley (1964) originally characterized the bigram frequency effect as depending on the predictability of each letter given its predecessors. By the same token, the approach might prove capable of addressing the behavior discussed in connection with our Simulation 4. If this is the case, then further data would be needed to adjudicate between the theory we have presented and the approach suggested by Hartley and Houghton (1996). It should be emphasized, however, that the two accounts are, in principle, empirically distinguishable. In what follows, we discuss one prediction that differentiates the present account from that of Hartley and Houghton, which we refer to as the *retrograde compatibility effect*.

Consider two sequences, presented as targets for serial recall. Let us assume that both sequences begin the same way, or as we put it, that the two share the same "base" sequence. However, the two sequences end in different ways, that is, they have different "codas." Assume, furthermore, that the coda for one list is highly consistent with the preceding base sequence; that is, the base and coda together yield an overall sequence that is relatively high in probability, given sequencing constraints familiar to the subject from previous experience. The coda for the other list is less consistent with the base, yielding a lower probability sequence. Given these two hypothetical stimuli, we now ask whether recall for the base sequence will be affected by its consistency with the coda. According to the theory of Houghton and Hartley (1996), the answer to this question should be no, because background knowledge is brought to bear only through forward chaining during recall. Therefore, recall for the base sequence should be equally accurate, regardless of the coda. In contrast, the model we have put forth predicts that the base sequence should be recalled more accurately when it is followed by a highly consistent coda than when it is followed by a less consistent coda.

To demonstrate this, we return briefly to the task discussed in Simulation 4. Recall that, here, sequences presented for recall were generated on the basis of an artificial grammar, which gave rise to a tendency to alternate between items assigned to two groups *A* and *B* (see Table 1). For present purposes, we focus on target lists beginning with the structure *ABAB*. Given the sequencing constraints implicit in the grammar, such sequences can end with either of two coda sequences, structured *AB* or *BA*. Using the terminology just established, the coda *AB* is more consistent with the base *ABAB*, because it yields an overall sequence (*ABABAB*) that is higher in goodness than the one yielded by the coda *BA* (*ABABBA*). According to the model of Hartley and Houghton (1996), this difference between the two codas should have no effect on recall accuracy for the four-item base sequence. However, this is not true for the model we have presented; when applied to the artificial grammar task, as described in Simulation 4, the model showed better recall for the order of the first four items of *ABABAB* lists than for the first four items of *ABABBA* lists (73% vs. 63% accuracy).[15]

[15] A detailed consideration of the Hartley and Houghton (1996) account indicates that it might, in fact, predict a greater number of exchanges between positions four and five in sequences of type *ABABBA*. This would, in turn, lead the theory to predict poorer recall for positions one through four. One way of controlling for this factor is to compare accuracy for the first four items, considering only cases on which stimulus items one through four appear, in some permutation, in the first four positions of the response sequence. In the case of this subset of trials, the Hartley and Houghton theory does appear to predict strictly equivalent recall accuracy for the initial four item subsequence. For the present theory, this approach provides a very conservative test of the retrograde compatibility effect, because it counts as correct trials in which the coda was not recalled

As it turns out, this aspect of the model's behavior matches the performance of human subjects performing the artificial grammar task. A new analysis of data from the experiment conducted by Botvinick (2005) indicates that, as in the model, recall was better for the first four items in *ABABAB* sequences than *ABABBA* sequences (accuracy 64% vs. 57%, paired t test, $p < .001$, two-tailed).[16] This finding provides an initial piece of empirical support for the retroactive compatibility effect predicted by the present model and presents a challenge to the competing theory presented by Hartley and Houghton (1996).

Another opportunity to distinguish between the Hartley and Houghton (1996) account and the one we have presented stems from the fact that, in the former model, the mechanisms that are sensitive to domain structure are structurally dissociable from the mechanisms responsible for short-term serial order memory. The Hartley and Houghton model thus predicts that it should be possible for focal brain damage to eliminate effects of background knowledge, while leaving serial recall performance otherwise intact. For example, it should be possible to identify patients who show normal memory span, when tested on lists of consonants, but who show no bigram frequency effect. According to the account we have put forth, such a dissociation should not occur, because the processing system's sensitivity to domain structure inheres in the same mechanisms that support basic serial recall.

Neuroscience-Based Models

In addition to the expressly psychological models reviewed so far, a number of models of serial recall have come out of the neuroscience literature (Beiser & Houk, 1998; Dominey, 1995, 1997; Dominey et al., 1995; Jaeger, 2001; O'Reilly & Soto, 2001). The central goal of these modeling efforts has been to engage neuroanatomic and neurophysiologic data, showing how certain neuroanatomical structures might support the basic function of serial recall, or how such structures might give rise to the patterns of neural activity observed in experiments using serial recall tasks. With regard to accounting for behavior, the goals of such models have been modest. For example, Beiser and Houk (1998), although providing a detailed account of single-unit recording data, dealt only with encoding, providing no account of how sequences are recalled. Dominey (1995) went a step further, providing a basic account of recall but not of generalization to never-before-seen sequences. O'Reilly and Soto (2001) addressed this aspect of behavior but, like the other models cited, did not address any of the behavioral patterns that have been of central interest to psychologists, such as primacy and recency, transposition gradients, similarity effects, or effects of domain structure.

Despite this difference in focus, there is a fundamental connection between the neuroscientific models just cited and the theory we have presented here, in that both use an activation-based memory mechanism supported by recurrent connectivity. In view of this common basis, the present model should not be viewed as directly competing with the neuroscience-based models we have cited. Rather, we view the two approaches as complementary. Although the neuroscientific models show how a recurrence- and activation-based account can be derived from neuroanatomic and neurophysiologic data, our work shows how such a framework can be used to explain detailed aspects of behavior.

Having said this, it is possible to point to differences between the account we have presented and those that have come out of neuroscience. One important contrast relates to the question of how relevant patterns of connectivity are established, and in particular, what role is played by learning.

The role of learning. Again, like the theory we have put forth, most neuroscience-based models use massively recurrent connectivity. However, in models such as those of Beiser and Houk (1998) and Dominey (1995, 1997), the specific pattern of connection weights involved in recurrent projections is fixed from the outset. Indeed, one interesting aspect of these accounts is that they assume patterns of connectivity that are in certain respects random. Subsequent theoretical work supports the idea that networks with essentially arbitrary recurrent connectivity can encode sequences of inputs (Jaeger, 2001; Maass et al., 2002; White et al., 2004). Such observations raise the question of whether the neural circuitry supporting memory for serial order might emerge independent of learning. Along these lines, Beiser and Houk (1998) suggested that the brain "has an innate ability to encode serial events into integrated concepts represented in spatial patterns of neural activity" (p. 3182).

It is interesting to note that learning, in our model, gave rise to patterns of connectivity corresponding, in fundamental ways, to those posited as innate by Beiser and Houk (1998). The latter model involved two recurrent projections: (a) excitatory feedback between prefrontal cortex and thalamus and (b) a circuit running from prefrontal cortex through basal ganglia and thalamus, and back to prefrontal cortex, which includes both excitatory and inhibitory connections (see Figure 1). The former circuit was made up of closed, parallel loops; that is, units in thalamus sent excitatory inputs only to those prefrontal units from which they receive excitatory inputs. The second circuit was more integrative, allowing activation in each prefrontal unit to impact many others. An inspection of the pattern of connection weights within our own model revealed a striking parallel to this dually recurrent pattern of connectivity. In particular, following training, there was a marked difference in distribution between (a) weights running from each unit to itself (self-connections) and (b) weights running between two different units (heteroconnections). As illustrated by the histogram shown in Figure 15, self-connections were almost all excitatory, whereas heteroconnections were smaller, with a magnitude distribution including both excitatory and inhibitory values, and centered near zero. The parallel between these two populations of connections, and the two recurrent pathways posited by Beiser and Houk is readily apparent.

Although Beiser and Houk (1998) suggested that innate patterns of connectivity may underlie the capacity to encode sequences, they added the following caveat:

> This is not to say that adaptive mechanisms do not play a role; for example, they would be quite useful for tuning the competitive pattern

correctly (final two items transposed). Nevertheless, the model presented in Simulation 4 continues to show a small but reliable difference in performance between sequence types *ABABAB* vs. *ABABBA*, even when only the specified subset of trials is considered (90% vs. 88%). This difference was consistently observed across training runs.

[16] The same comparison was conducted focusing on the subset of trials specified in the preceding note, with similar results (accuracy 79% vs. 74%, paired t test, $p < 0.05$, two tailed).

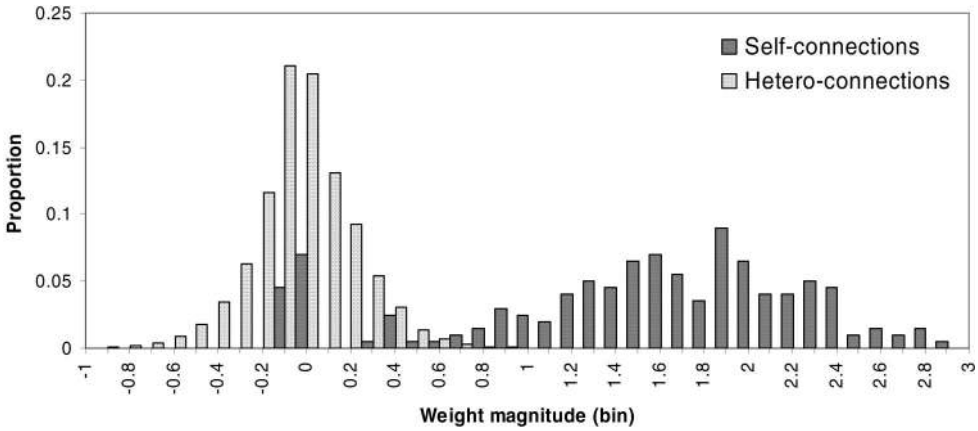

Figure 15. Distribution of weight magnitudes for self-connections (connections from a unit to itself) and heteroconnections (connections from one unit to another).

> classification stage so as to improve the encoding performance of the network. This might provide a more efficient code or it might emphasize serial events that are of particular relevance to the organism. (p. 3181)

Indeed, some experience-based tuning of connectivity seems to be required to give rise to effects of background knowledge, such as the bigram frequency effect. Even in arbitrary domains, some learning appears to be needed to implement the process of recall; at least, to our knowledge, learning has played some role in every neuroscience-based account of serial recall that has addressed the recall process (Dominey, 1995, 1997; Jaeger, 2001; O'Reilly & Soto, 2001; White et al., 2004).

Given the important role that learning plays in our theory in shaping the mechanisms underlying serial recall, it must be asked to what extent the specific kind of learning involved in our simulations might correspond to learning in the brain. Our use of back-propagation learning, in particular, raises questions concerning biological plausibility. It is important to note that previous research (e.g., Zipser & Anderson, 1988) has demonstrated that back-propagation can give rise to patterns of activity closely resembling those observed in actual neural systems. However, it must be acknowledged that certain aspects of back-propagation, in particular the transfer of error signals backward across synapses, have not yet been linked to known biological mechanisms.

In this regard, it is interesting to note that some neuroscience-based models of serial recall have used learning algorithms more closely tied to biological mechanisms. In particular, Dominey (1995, 1997) used a Hebbian reinforcement learning algorithm arguably grounded in neuroscientific data. One limitation of this algorithm, which discouraged its use in our simulations, is that it does not support learning of internal representations, within which inputs are recoded in a task-specific fashion. A wealth of computational work, in both psychology and neuroscience, suggests that the ability to recode information internally may be critical to performance in numerous domains (examples in the realm of sequencing include Botvinick & Plaut, 2004; Cleeremans, 1993; Elman, 1990). Thus, until it can be shown that the learning approach adopted by Dominey (1995, 1997) can be used to account for detailed patterns of behavior, such as those addressed by our own simulations, its viability in the domain of serial recall must remain open to question.

There do exist biologically plausible learning algorithms that support the learning of internal representations (O'Reilly & Munakata, 2000; Xiaohui & Seung, 2003). However, currently available algorithms are limited in their capacity to learn sequences. This limitation has been stressed by O'Reilly (2003), who proposed that it may be overcome, in the brain, by a special gating mechanism. Indeed, O'Reilly and Soto (2001) directly addressed how this gating mechanism might support ISR. At a general level, the account they put forth is not inconsistent with the one we have provided. However, at the current stage of development, the O'Reilly and Soto account relies on storage of item information in independent position-specific slots. It is difficult to see how such an implementation could account for such aspects of serial recall as the locality constraint.

As the foregoing comments imply, there is, as of yet, no account of learning that is both biologically validated and computationally adequate to the domain of serial recall. In using back-propagation, we express a commitment to an account of learning that involves gradual, error-based adjustments in connection weights and that gives rise to a task-relevant recoding of inputs in the form of distributed internal representations. Whether such an account can be grounded in neurobiology is, of course, an important question currently under investigation by numerous researchers in neuroscience.

Hippocampal models. In addition to the models cited so far, there is another group of neuroscience-based models of serial recall, which invoke rather different sequencing mechanisms. These models relate specifically to the function of medial temporal lobe structures including the hippocampus (e.g., Levy, 1996; Lisman, 1999). Arguably, such models are not of immediate relevance to the behavioral phenomena we have focused on. This is because neuropsychological evidence indicates that serial recall performance (at or below memory span) is essentially unaffected by lesions to such temporal lobe structures (Baddeley & Warrington, 1970; Drachman & Arbit, 1966; Warrington, 1982). Thus, serial recall appears to depend on other neural mechanisms, such as the cortico-basal-ganglionic loops addressed by such models as those of Beiser and Houk (1998) and Dominey (1995).

This being said, it is of course not our claim that hippocampal mechanisms are entirely irrelevant to understanding sequence memory in general. Neurophysiologic evidence clearly indicates

that the hippocampus does encode sequences (Fortin, Agster, & Eichenbaum, 2002; A. K. Lee & Wilson, 2002; Lisman, 1999), and medial temporal lobe lesions have been associated with marked deficits in sequence memory above span (Drachman & Arbit, 1966). Even at or below span, a contribution of medial temporal lobe memory mechanisms to serial recall cannot be ruled out; indeed, we consider below at least one phenomenon (protrusion errors) that may reflect an impact of medial temporal lobe structures in ISR performance, even at relatively short list lengths. However, whatever the contribution of the hippocampus and associated structures, because these mechanisms are not necessary to most aspects of span-level ISR, the primary mechanisms underlying this function must be sought elsewhere.

Predictions

An important aspect of the account we have put forth is that it gives rise to distinctive and testable predictions. Several such predictions have already been introduced, including the retrograde consistency effect and the indissociability of domain-specific effects from basic serial memory function. As detailed above, the first of these predictions has already been preliminarily tested and confirmed. The latter stands as a falsifiable claim of the account.

Further predictions can be derived from our analysis of the model's sequence representations, as laid out in the Initial Analyses section. Specifically, these translate directly into predictions concerning neural activity during performance of the ISR task. At the most general level, the model predicts that the encoding process should result in a distributed pattern of neural activation that contains information about all items in the target sequence and their respective positions. As noted earlier, this is consistent with some neuroscience-based models of sequence encoding (Beiser & Houk, 1998; Dominey, 1995; O'Reilly & Soto, 2001). At a more specific level, the current model predicts that individual neurons participating in the representation of sequences will code not only for a specific item or position, but also, in a graded fashion, for items or positions that are similar. This prediction has direct corollaries at the level of overall sequence representations. If one considers the patterns of activation arising at the end of encoding, these are predicted to show the following properties: (a) patterns for lists containing the same items in different orders are predicted to resemble one another to the extent that the orderings are similar and (b) sequence representations for lists composed of the same elements, in different orders, will be more similar if the items involved are confusable than if they are not. These predictions, which are illustrated in Figure 16 on the basis of model output, might be testable using techniques such as functional magnetic resonance imaging, which has recently been applied to evaluate similarity relations between distributed neural representations (see, e.g., Piazza, Izard, Pinel, Le Bihan, & Dehaene, 2004).

A final prediction concerns the way that item and position information combine at the neural level. The model predicts that at any given point in time, the way that target items in the list are represented will depend on their list position. Furthermore, the representation of any item–position conjunction should evolve over successive steps of encoding and recall. Thus, for example, the presence of item *i* in position *p* will be represented by a

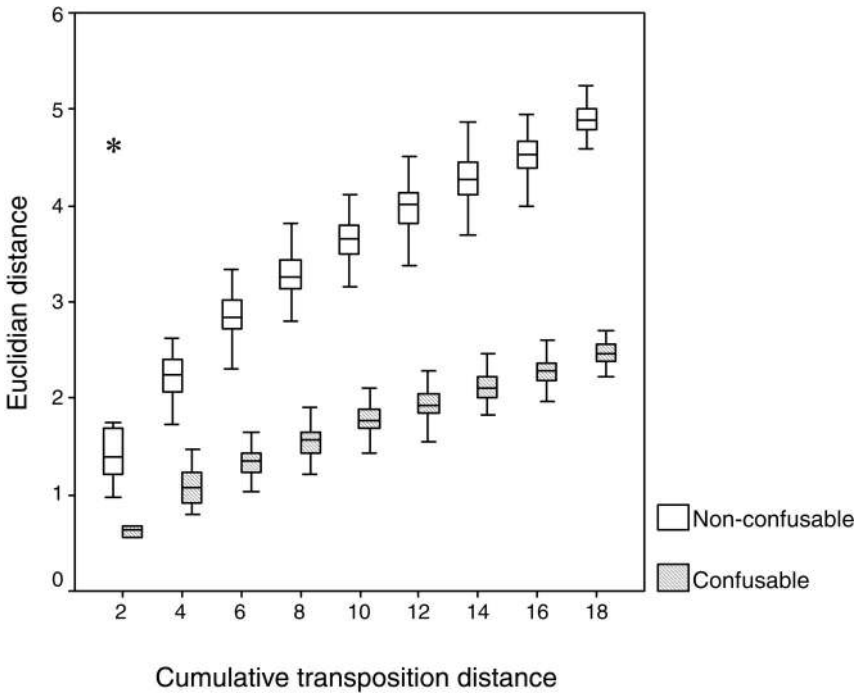

Figure 16. Distance in hidden-unit space between encodings for lists with a range of similarities. "Cumulative transposition distance" refers to the total number of shifts between adjacent positions that would be required to transform one of the lists being compared into the other. Boxes indicate interquartile range and median. Whiskers indicate range. The asterisk indicates the median for pairs of lists containing nonoverlapping sets of items. Confusable, nonconfusable = acoustically confusable and nonconfusable items.

different (although possibly overlapping) set of neurons at encoding step $p + 2$ than on step $p + 1$. This prediction of the model may appear implausible, given that most neuroscientific studies of working memory have stressed the role of tonic, sustained activation in preserving information over time. However, although less often discussed, empirical studies have also routinely observed activations that change gradually over time (see Batuev, 1994; Fuster, 1997). Indeed, a recent study by Brody, Hernandez, Zainos, and Romo (2002), in which the evolution of working memory codes was analyzed in detail, prefrontal neurons appeared to implement something very close to the rotational coding scheme we described within the Initial Analyses section. Moreover, in another recent single-unit recording study (Inoue & Mikami, 2006), the neural representation of items in a sequence was observed to change over successive steps of encoding. How general such evolving working memory codes are, and the extent to which they are involved in serial order memory, are questions for future empirical research.

The model's predictions concerning representational structure, as detailed both in the preceding analysis and elsewhere in the article, could in principle be evaluated through neurophysiologic experiments with monkeys, along the lines of those performed by Barone and Joseph (1989) and Ninokura et al. (2003, 2004). However, there is an important qualification to make, in this regard. We have, along with others, interpreted such existing studies as providing support for an activation-based memory mechanism. However, in our view, it is more difficult to evaluate the information such studies have provided concerning specific patterns of neural activation. The problem is that existing neurophysiologic studies have involved massive experience with a very small set of sequences (a total of six, in both Barone & Joseph, 1989, and Ninokura et al., 2003, 2004). It seems almost inevitable that such conditions would give rise to highly sequence-specific representations (see Tanji & Shima, 1994), possibly quite different in character from those that would be used to encode less familiar sequences of items. To be sure, this is what would occur in our model were it to be trained on such a limited corpus. For this reason, without modification, existing paradigms in animal neurophysiology do not appear to guarantee a suitable basis for evaluating the model's predictions concerning representational structure. A substantive test of these predictions will thus depend on future methodological innovations.

Addressing Further Behavioral Phenomena

Aspects of ISR Not Modeled

Although we have aimed to address a broad set of benchmark phenomena pertaining to ISR, there are, of course, a number of findings against which the theory has not yet been tested. These include effects of stimulus timing (Neath & Crowder, 1996), stimulus modality (Crowder, 1972), response delay (Bjork & Healy, 1974), suffix effects (Baddeley & Hull, 1979), item frequency (Hulme et al., 1997), articulatory suppression (Murray, 1967), irrelevant speech (Neath, 2000), and recall order (including reverse and free recall; Klein, Addis, & Kahana, 2005; Li & Lewandowsky, 1995), just to name a few. Whether the theory we have presented will prove sufficient to deal with such additional phenomena stands as a question for future work.[17]

This being said, there are two empirical phenomena that call for further comment, given the importance that has been accorded to them in recent theoretical work on serial recall. These are (a) the effect of grouping and (b) intrusions from preceding lists.

Grouping

A variety of studies have shown that subjects performing ISR often spontaneously adopt a strategy of partitioning items into smaller groups. Experiments focusing on this phenomenon have shown that such grouping can benefit recall accuracy (Wickelgren, 1967) but also gives rise to distinctive error patterns. A particularly important finding is that grouping alters the usual transposition curve, such that the frequency of transpositions between items occupying corresponding positions within their respective groups rises to levels close to those for transpositions between adjacent items (Henson, 1998; C. L. Lee & Estes, 1981; Ryan, 1969a, 1969b; Wickelgren, 1967; Figure 17, left). This pattern has been a focus of recent theoretical discussions, partially as a result of the claim by Henson (1999) that it rules out models that do not use an explicit positional code. Thus, it is important to address how intergroup transpositions might be addressed by the model we have proposed.

Figure 17 (right) shows a transposition curve for nine-item lists, generated using the model described in Simulation 2, but using nine-item sequences with a special structure. Specifically, Items 1, 4, and 7 were selected so as to be mutually confusable but nonconfusable with other items in the list, and similarly for Items 2, 5, and 8, and for Items 3, 6, and 9. The resulting pattern of errors resembles that observed empirically for lists grouped by threes. What this shows is that the model can account for transpositions between items in the same within-group position, if an assumption is made concerning the encoding of grouped items. Specifically, the assumption is that items occupying the same within-group position are represented similarly at encoding. Page and Norris (1998; see also Wickelgren, 1967) have suggested that items in groups are marked with a coarse indication of their position (distinguishing only among beginning, middle, and end). If this labeling is assumed to affect the way that items are encoded, then it could provide a basis for the grouping effect, as reflected in Figure 17. However, there are other reasons that items in corresponding within-group positions might be encoded similarly. One that strikes us as particularly plausible is that items occupying identical positions within different groups tend to receive a similar intonation or emphasis. This could provide another basis for the assumption that such items are represented similarly. In favor of this idea, Reeves, Schmauder, and Morris (2000) showed that stress patterns in serial recall stimuli can induce specific grouping strategies (see also Frankish, 1995). Even more to the point, Palmer and Pfordresher (2003) showed that, in music performance,

[17] In some cases, the present model could apparently be applied directly to further phenomena: for instance, item frequency or suffix effects. In others, a more elaborate implementation would clearly be needed. For example, Lewandowsky and colleagues (Lewandowsky & Brown, 2005; Nimmo & Lewandowsky, 2005) have presented data arguing for an event-based processing mechanism, as opposed to a time-based mechanism. Such a distinction is difficult to address using the present model, given its very coarse discretization of time. However, the present model could in principle be implemented using a more continuous representation of time, allowing exploration of the relevant issues.

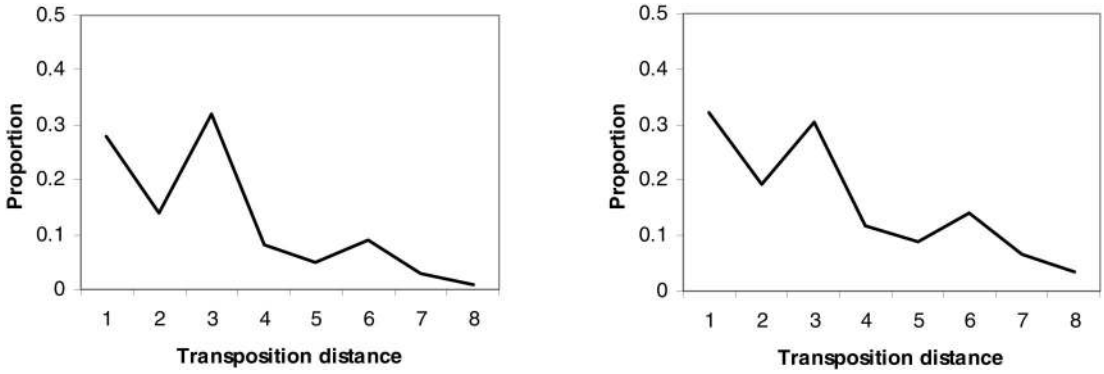

Figure 17. Left: Empirical data from Henson (1998), for lists grouped by threes. "Transposition distance" refers to the distance, in either direction, between an item's position in the target list and its position at recall. Right: Performance of the model discussed in Simulation 2, trained on nine-item lists, and tested on sequences with a special similarity structure, as described in the text. Left panel from "Short-Term Memory for Serial Order: The Start–End Model," by R. N. A. Henson, 1998, *Cognitive Psychology, 36,* p. 96. Copyright 1998 by Elsevier. Adapted with permission.

errors in note selection tend to involve confusions between notes with similar stress.

Although the pattern of errors reflected in Figure 17 has received the most attention in recent theoretical debates, there are many other interesting consequences of grouping that we do not address here (see Frick, 1989; Hitch, Burgess, Towse, & Culpin, 1996; Ryan, 1969a). Such phenomena represent an area for future work with the model.

Protrusions

Subjects performing ISR have been shown to make errors, at above chance levels, that involve an intrusion into the current list of an item appearing on the previous trial (Conrad, 1959, 1960; Henson et al., 1996). It is interesting to note that when such intrusions (or, as Henson et al., 1996, call them, protrusions) occur, the intruding item tends to occupy the same list position that it held in its original context. Context-based models (e.g., Henson, 1998) account for this finding by assuming that the Hebbian associations between items and context states persist between trials. Because the context representation follows the same temporal trajectory on each trial, such persisting links will lead to the reactivation of items occurring at the same list position on the preceding trial.

Our model does not provide an account of protrusions, at least not as currently implemented. It is interesting to consider that, in other work, recurrent neural networks with continuous time dynamics and adjustable weights have been used to model priming phenomena (Cree, McRae, & McNorgan, 1999; Plaut & Booth, 2000). It is possible that the principles involved in such modeling work might be integrated with the framework we have established here, resulting in an account of the protrusion effect.

However, other considerations suggest that it may not be necessary, or even appropriate, to apply the present model to protrusion errors. Page and Norris (1998) have argued, in agreement with Estes (1991), that protrusions may reflect the contribution of memory mechanisms separate from those primarily responsible for supporting short-term sequence memory. They pointed out, first, that because protrusions span successive trials, they reflect an influence of memory traces covering intervals considerably longer than short-term sequence memory mechanisms are typically considered to span. Indeed, as Page and Norris pointed out, there is evidence suggesting that the short-term availability of phonemic codes (Baddeley, 1986) may actually protect against proactive interference in ISR. This, Page and Norris suggest, is at least consistent with the idea that protrusion errors derive from some mechanism other than the one primarily responsible for carrying sequence information over the short term. A firm foundation for this argument could be provided if it were shown that some form of neurological injury resulted in an elimination or reduction of protrusion errors, in the face of otherwise normal short-term serial recall. Amnesia based on medial temporal lobe injury provides an obvious context in which to test this prediction. Although, as noted earlier, amnesic patients typically show normal short-term memory span, it has not, to our knowledge, been asked whether they display the same frequency of protrusion errors as normal controls. If this were found to be the case, then it might be reasonable to consider a hybrid model of short-term serial order memory, containing both an activation-based component, along the lines we have proposed, and a weight-based component, understood as involving the medial temporal lobe. Such a hybrid approach has in fact already been taken, with considerable success, in recent research on free recall (Davelaar, Goshen-Gottstein, Ashkenazi, Haarmann, & Usher, 2005). Depending on where the empirical data point, such work could potentially serve as a template for potential further developments in modeling serial recall.

Conclusion

Short-term memory for serial order has been of central interest to psychologists for many years. However, we believe the study of this form of memory currently stands at a very interesting and promising juncture. First, thanks to decades of behavioral research on ISR, we now possess an exquisitely detailed and theoretically constraining picture of what human performance is like in this domain. Second, thanks to increasingly sophisticated efforts to develop computational models of serial recall, and an intensification of such efforts in the last decade, a coherent set of plausible and compelling theoretical alternatives has come into focus. Third,

highly informative data have gradually begun to emerge from neuroscientific studies of serial recall. Finally, there has been the growing recognition of a deep connection between short-term sequence memory and the processing of language (Baddeley, 2003), a development that both amplifies the importance of serial recall as an object of psychological inquiry and brings to the fore the neglected question of how short-term memory for serial order may interact with domain-specific background knowledge.

The theory of recall that we have presented is situated where these four developments intersect. The account picks up some fundamental themes arising from neuroscientific research, implementing these in a form that brings them into contact with detailed behavioral data. Analysis of the resulting model revealed a novel mechanism for short-term serial order memory, one that contrasts in fundamental ways with currently prevalent psychological accounts, and which makes numerous predictions both about recall behavior and about the neural representation of sequence information.

Of course, for all of its successes, the work we have presented leaves numerous open questions. In recognition of this, we have endeavored to point out aspects of the model that are in need of further development. Even in its current form, however, the model establishes a distinctive theoretical perspective, one that appears worthy of serious consideration in future research on serial recall.

References

Anderson, J. R., & Matessa, M. (1997). A production system theory of serial memory. *Psychological Review, 104,* 728–748.

Baddeley, A. D. (1964). Immediate memory and the "perception" of letter sequences. *Quarterly Journal of Experimental Psychology, 16,* 364–367.

Baddeley, A. D. (1968). How does acoustic similarity influence short-term memory? *Quarterly Journal of Experimental Psychology, 20,* 249–264.

Baddeley, A. D. (1986). *Working memory.* New York: Clarendon Press.

Baddeley, A. D. (2003). Working memory and language: A review. *Journal of Communication Disorders, 36,* 189–208.

Baddeley, A. D., Conrad, R., & Hull, A. J. (1965). Predictability and immediate memory for consonant sequences. *Quarterly Journal of Experimental Psychology, 17,* 175–177.

Baddeley, A. D., & Hull, A. (1979). Prefix and suffix effects: Do they have a common basis? *Journal of Verbal Learning and Verbal Behavior, 18,* 129–140.

Baddeley, A. D., & Warrington, E. K. (1970). Amnesia and the distinction between long- and short-term memory. *Journal of Verbal Learning and Verbal Behavior, 9,* 176–189.

Barone, P., & Joseph, J. P. (1989). Prefrontal cortex and spatial sequencing in macaque monkey. *Experimental Brain Research, 78,* 447–464.

Batuev, A. S. (1994). Two neuronal systems involved in short-term spatial memory in monkeys. *Acta Neurobiologiae Experimentalis, 54,* 334–344.

Beiser, D. G., & Houk, J. C. (1998). Model of cortical-basal ganglionic processing: Encoding the serial order of sensory events. *Journal of Neurophysiology, 79,* 3168–3188.

Bishop, C. M. (1995). *Neural networks for pattern recognition.* New York: Oxford University Press.

Bjork, E. L., & Healy, A. F. (1974). Short-term order and item retention. *Journal of Verbal Learning and Verbal Behavior, 13,* 80–97.

Blankenship, A. B. (1938). Memory span: A review of the literature. *Psychological Bulletin, 35,* 1–25.

Botvinick, M. (2005). Effects of domain-specific knowledge on memory for serial order. *Cognition, 97,* 135–151.

Botvinick, M., & Bylsma, L. M. (2005). Regularization in short-term memory for serial order. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 31,* 351–358.

Botvinick, M., & Huffstettler, S. (2006). *Sequence learning in short-term memory: Computational and empirical investigations of the Hebb effect.* Manuscript submitted for publication.

Botvinick, M., & Plaut, D. C. (2004). Doing without schema hierarchies: A recurrent connectionist approach to normal and impaired routine sequential action. *Psychological Review, 111,* 395–429.

Brody, C. D., Hernandez, A., Zainos, A., & Romo, R. (2002). Timing and neural encoding of somatosensory parametric working memory in macaque prefrontal cortex. *Cerebral Cortex, 13,* 1196–1207.

Brown, G., Preece, T., & Hulme, C. (2000). Oscillator-based memory for serial order. *Psychological Review, 107,* 127–181.

Brown, G. D. A., Neath, I., & Chater, N. (2005). *SIMPLE: A local distinctiveness model of scale-invariant memory and perceptual identification.* Manuscript submitted for publication.

Brown, J. W., Bullock, D., & Grossberg, S. (2004). How laminar frontal cortex and basal ganglia circuits interact to control planned and reactive saccades. *Neural Networks, 17,* 471–510.

Burgess, N., & Hitch, G. J. (1992). Toward a network model of the articulatory loop. *Journal of Memory and Language, 21,* 429–460.

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

Cer, D. M., & O'Reilly (in press). Neural mechanisms of binding in the hippocampus and neocortex: Insights from computational models. In H. D. Zimmer, A. Mecklingber, & U. Lindenberger (Eds.), *Binding in memory.* Oxford, England: Oxford University Press.

Cleeremans, A. (1993). *Mechanisms of implicit learning: Connectionist models of sequence processing.* Cambridge, MA: MIT Press.

Compte, A., Brunel, N., Goldman-Rakic, P. S., & Wang, X. -J. (2000). Synaptic mechanisms and network dynamics underlying spatial working memory in a cortical network model. *Cerebral Cortex, 10,* 910–923.

Conrad, R. (1959). Errors of immediate memory. *British Journal of Psychology, 50,* 349–359.

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

Conrad, R. (1965). Order error in immediate recall of sequences. *Journal of Verbal Learning and Verbal Behavior, 4,* 161–169.

Conrad, R., & Hull, A. J. (1964). Information, acoustic confusion, and memory span. *British Journal of Psychology, 55,* 429–432.

Crannell, C. W., & Parrish, J. M. (1957). A comparison of immediate memory span for digits, letters, and words. *Journal of Psychology, 44,* 319–327.

Cree, G. S., McRae, K., & McNorgan, C. (1999). An attractor model of lexical conceptual processing: Simulating semantic priming. *Cognitive Science, 23,* 371–414.

Crowder, R. G. (1972). Visual and auditory memory. In J. F. Kavanagh & I. G. Mattingly (Eds.), *Language by ear and by eye* (pp. 251–276). Cambridge, MA: MIT Press.

Cumming, N., Page, M., & Norris, D. (2003). Testing a positional model of the Hebb effect. *Memory, 11,* 43–63.

Davelaar, E. J., Goshen-Gottstein, Y. Ashkenazi, A., Haarmann, H. J., & Usher, M. (2005). The demise of short-term memory revisited: Empirical and computational investigations of recency effects. *Psychological Review, 112,* 3–42.

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

Dominey, P. F. (1997). An anatomically structured sensory-motor sequence learning system displays some general linguistic capacities. *Brain and Language, 59,* 50–75.

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

Drachman, D. A., & Arbit, J. (1966). Memory and the hippocampal complex: II. Is memory a multiple process? *Archives of Neurology, 15,* 52–61.

Drewnowski, A. (1980). Attributes and priorities in short-term recall: A new model of memory span. *Journal of Experimental Psychology, 109,* 208–250.

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

Estes, W. K. (1972). An associative basis for coding and organization in memory. In A. W. Melton & E. Martin (Eds.), *Coding processes in human memory* (pp. 161–190). Washington, DC: Winston.

Estes, W. K. (1991). On types of item coding and sources of recall in short-term memory. In W. E. Hockley & S. Lewandowsky (Eds.), *Relating theory and data: In honor of Bennet B. Murdock* (pp. 155–173). Hillsdale, N. J.: Erlbaum.

Farrell, S., & Lewandowsky, S. (2002). An endogenous distributed model of ordering in serial recall. *Psychonomic Bulletin & Review, 9,* 59–79.

Farrell, S., & Lewandowsky, S. (2003). Dissimilar items benefit from phonological similarity in serial recall. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 29,* 838–849.

Fortin, N. J., Agster, K. L., & Eichenbaum, H. B. (2002). Critical role for the hippocampus in memory for sequences of events. *Nature Neuroscience, 5,* 458–462.

Frankish, C. (1995). Intonation and auditory grouping in immediate serial recall. *Applied Cognitive Psychology, 9,* 5–22.

Frick, R. W. (1989). Explanations of grouping in immediate ordered recall. *Memory & Cognition, 17,* 551–562.

Funahashi, S., Inoue, M., & Kubota, K. (1997). Delay-period activity in the primate prefrontal cortex encoding multiple spatial positions and their order of presentation. *Behavioural Brain Research, 84,* 203–223.

Fuster, J. M. (1997). *The prefrontal cortex: Anatomy, physiology, and neuropsychology of the frontal lobe.* Philadelphia: Lippincott-Raven.

Gathercole, S. E. (1995). Is nonword repetition a test of phonological memory or long-term knowledge? It all depends on the nonwords. *Memory & Cognition, 23,* 83–94.

Gathercole, S. E., Frankish, C. R., Pickering, S. J., & Peaker, S. (1999). Phonotactic influences on short-term memory. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 25,* 84–95.

Gathercole, S. E., Willis, C., Emslie, H., & Baddeley, A. D. (1991). The influence of syllables and wordlikeness on children's repetition of nonwords. *Applied Psycholinguistics, 12,* 349–367.

Glasspool, D. W., & Houghton, G. (1997). Dynamic representation of structural constraints in models of serial behaviour. In J. Bullinaria, D. Glasspool, & G. Houghton (Eds.), *Proceedings of the 4th Neural Computation and Psychology Workshop: Connectionist representations* (pp. 269–282). London: Springer-Verlag.

Goldman-Rakic, P. S. (1995). Cellular basis of working memory. *Neuron, 14,* 477–485.

Grant, J., Karmiloff-Smith, A., Gathercole, S. E., Paterson, S., Howlin, P., Davies, M., et al. (1997). Phonological short-term memory and its relationship to language in Williams syndrome. *Cognitive Neuropsychiatry, 2,* 81–99.

Grossberg, S. (1977). A theory of human memory: Self-organization and performance of sensory-motor codes, maps, and plans. In R. Roden & F. Snell (Eds.), *Progress in theoretical biology* (pp. 496–639). New York: Academic Press.

Grossberg, S. (1978). Behavioral contrast in short-term memory: Serial binary memory models or parallel continuous memory models? *Journal of Mathematical Psychology, 17,* 199–219.

Grossberg, S. (1986). The adaptive self-organization of serial order in behavior: Speech, language, and motor control. In E. C. Schwab & H. C. Nusbaum (Eds.), *Pattern recognition by humans and machines. Vol. 1: Speech perception* (pp. 187–294). New York: Academic Press.

Gupta, P., & MacWhinney, B. (1997). Vocabulary acquisition and verbal short-term memory: Computational and neural bases. *Brain and Language, 59,* 267–333.

Hartley, T., & Houghton, G. (1996). A linguistically constrained model of short-term memory for nonwords. *Journal of Memory and Language, 35,* 1–31.

Hebb, D. O. (1949). *The organization of behavior: A neuropsychological theory.* New York: Wiley

Henson, R. N. A. (1996). *Short-term memory for serial order.* Unpublished doctoral dissertation. MRC Applied Psychology Unit, University of Cambridge, England.

Henson, R. N. A. (1998). Short-term memory for serial order: The start–end model. *Cognitive Psychology, 36,* 73–137.

Henson, R. N. A. (1999). Coding position in short-term memory. *International Journal of Psychology, 34,* 403–409.

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

Hitch, G. J., Burgess, N., Towse, J. N., & Culpin, V. (1996). Temporal grouping effects in immediate recall: A working memory analysis. *Quarterly Journal of Experimental Psychology: Human Experimental Psychology, 49*A, 116–139.

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

Houghton, G. (1990). The problem of serial order: A neural network model of sequence learning and recall. In R. Dale, C. Nellish, & M. Zock (Eds.), *Current research in natural language generation* (pp. 287–318). San Diego, CA: Academic Press.

Houghton, G., & Hartley, T. (1995). Parallel models of serial behaviour: Lashley revisited. *Psyche, 2*(25). Retrieved February 21, 2006, from http://psyche.cs.monash.edu.au/v2/psyche-2–25-houghton.html

Hulme, C., Maughan, S., & Brown, G. D. A. (1991). Memory for familiar and unfamiliar words: Evidence for a long-term memory contribution to short-term memory span. *Journal of Memory and Language, 30,* 685–701.

Hulme, C., Roodenrys, S., Schweickert, R., Brown, G. D. A., Martin, S., & Stuart, G. (1997). Word-frequency effects on short-term memory tasks: Evidence for a redintegration process in immediate serial recall. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 23,* 1217–1232.

Inoue, M., & Mikami, A. (2006). Prefrontal activity during serial probe reproduction task: Encoding, mnemonic and retrieval processes. *Journal of Neurophysiology, 95,* 1008–1041.

Jaeger, H. (2001). *The "echo state" approach to analyzing and training recurrent neural networks* (Tech. Rep. No. 148). Germany: German National Research Center for Information Technology.

Jahnke, J. C. (1963). Serial position effects in immediate serial recall. *Journal of Verbal Learning and Verbal Behavior, 2,* 284–287.

Jahnke, J. C. (1965). Primacy and recency effects in serial-position curves of immediate recall. *Journal of Experimental Psychology, 70,* 130–132.

Jordan, M. I. (1986). An introduction to linear algebra in parallel distributed processing. In D. E. Rumelhart, J. L. McClelland, & the PDP Research Group (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition. Vol. 1: Foundations* (pp. 365–422). Cambridge, MA: MIT Press.

Kantowitz, B. H., Ornstein, P. A., & Schwartz, M. (1972). Encoding and immediate serial recall of consonant strings. *Journal of Experimental Psychology, 93,* 105–110.

Klein, K. A., Addis, K. M., & Kahana, M. J. (2005). A comparative analysis of serial and free recall. *Memory & Cognition, 33,* 833–839.

Lashley, K. S. (1951). The problem of serial order in behavior. In L. A. Jeffress (Ed.), *Cerebral mechanisms in behavior: The Hixon symposium* (pp. 112–136). New York: Wiley.

Lee, A. K., & Wilson, M. A. (2002). Memory of sequential experience in the hippocampus during slow wave sleep. *Neuron, 36,* 1183–1194.

Lee, C. L., & Estes, W. K. (1977). Order and position in primary memory for letter strings. *Journal of Verbal Learning and Verbal Behavior, 16,* 395–418.

Lee, C. L., & Estes, W. K. (1981). Item and order information in short-term memory: Evidence for multilevel perturbation processes. *Journal of Experimental Psychology, 7,* 149–169.

Levy, W. B. (1996). A sequence predicting CA3 is a flexible associator that learns and uses context to solve hippocampal-like tasks. *Hippocampus, 6,* 579–590.

Lewandowsky, S. (1999). Redintegration and response suppression in serial recall: A dynamic network model. *International Journal of Psychology, 34,* 434–446.

Lewandowsky, S., & Brown, G. D. A. (2005). Serial recall and presentation schedule: A micro-analysis of local distinctiveness. *Memory, 13,* 283–292.

Lewandowsky, S., Brown, G. D. A., Wright, T., & Nimmo, L. M. (in press). Timeless memory: Evidence against temporal distinctiveness models of short-term memory for serial order. *Journal of Memory and Language*.

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

Li, S. C., & Lewandowsky, S. (1995). Forward and backward recall: Different retrieval processes. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 21,* 837–847.

Lisman, J. E. (1999). Relating hippocampal circuitry to function: Recall of memory sequences by reciprocal dentate–CA3 interactions. *Neuron, 22,* 233–242.

Maass, W., Natschlager, T., & Markram, H. (2002). Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural Computation, 14,* 2531–2560.

Marcus, M., Santorini, B., & Marcinkiewicz, M. (1993). Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics, 19,* 313–330.

Marshuetz, C. (2005). Order information in working memory: An integrative review of evidence from brain and behavior. *Psychological Bulletin, 131,* 323–339.

Mayzner, M. S., & Schoenberg, K. M. (1964). Single-letter and digram frequency effects in immediate serial recall. *Journal of Verbal Learning and Verbal Behavior, 3,* 397–400.

McClelland, J. L. (1998). Connectionist models and Bayesian inference. In M. Oaksford & N. Chater (Eds.), *Rational models of cognition* (pp. 21–53). Oxford, England: Oxford University Press.

McCormack, T., Brown, G. D. A., & Vousden, J. I. (2000). Children's serial recall errors: Implications for theories of short-term memory development. *Journal of Experimental Child Psychology, 76,* 222–252.

Mewhort, D. J. K., Popham, D., & James, G. (1994). On serial recall: A critique of chaining in the theory of distributed associative memory. *Psychological Review, 101,* 534–538.

Miller, E. K., & Cohen, J. D. (2001). An integrative theory of prefrontal cortex function. *Annual Review of Neuroscience, 24,* 167–202.

Miller, G. A., & Selfridge, J. A. (1951). Verbal context and the recall of meaningful material. *American Journal of Psychology, 63,* 176–185.

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

Murdock, B. B. (1997). Context and mediators in a theory of distributed associative memory. *Psychological Review, 104,* 839–862.

Murray, D. J. (1967). The role of speech responses in short-term memory. *Canadian Journal of Psychology, 21,* 263–276.

Nairne, J. S. (1990). A feature model of immediate memory. *Memory & Cognition, 18,* 251–269.

Nairne, J. S., & Neath, I. (1994). Critique of the retrieval/deblurring assumptions of the theory of distributed associative memory. *Psychological Review, 101,* 528–533.

Neath, I. (1998). *Human memory: An introduction to research, data, and theory.* Pacific Grove, CA: Brooks/Cole.

Neath, I. (2000). Modeling the effects of irrelevant speech on memory. *Psychonomic Bulletin & Review, 7,* 403–423.

Neath, I., & Crowder, R. G. (1996). Distinctiveness and very short-term serial position effects. *Memory, 4,* 225–242.

Nimmo, L. M., & Lewandowsky, S. (2005). From brief gaps to very long pauses: Temporal isolation does not benefit serial recall. *Psychonomic Bulletin & Review, 12,* 999–1004.

Ninokura, Y., Mushiake, H., & Tanji, J. (2003). Representation of the temporal order of visual objects in the primate lateral prefrontal cortex. *Journal of Neurophysiology, 89,* 2868–2873.

Ninokura, Y., Mushiake, H., & Tanji, J. (2004). Integration of temporal order and object information in the monkey lateral prefrontal cortex. *Journal of Neurophysiology, 91,* 555–560.

Nipher, F. E. (1876). On the distribution of numbers written from memory. *Transactions of the Academy of St. Louis, 3,* 79–80.

O'Reilly, R. C. (2003). *Making working memory work: A computational model of learning in the prefrontal cortex and basal ganglia* (Tech. Rep. No. 03–03). Institute of Cognitive Science, University of Colorado, Boulder.

O'Reilly, R. C., Braver, T. S., & Cohen, J. D. (1999). A biologically-based computational model of working memory. In A. Miyake & P. Shah (Eds.), *Models of working memory: Mechanisms of active maintenance and executive control* (pp. 375–411). New York: Cambridge University Press.

O'Reilly, R. C., & Munakata, Y. (2000). *Computational explorations in cognitive neuroscience.* Cambridge, MA: MIT Press.

O'Reilly, R. C., & Soto, R. (2001). A model of the phonological loop: Generalization and binding. In T. G. Dietterich, S. Becker, & Z. Ghahramani (Eds.), *Advances in neural information processing systems* (Vol. 14; pp. 83–90). Cambridge, MA: MIT Press.

Page, M., & Norris, D. (1998). The primacy model: A new model of immediate serial recall. *Psychological Review, 105,* 761–781.

Palmer, C., & Pfordresher, P. Q. (2003). Incremental planning in sequence production. *Psychological Review, 110,* 683–712.

Piazza, M., Izard, V., Pinel, P., Le Bihan, D., & Dehaene, S. (2004). Tuning curves for approximate numerosity in the human intraparietal sulcus. *Neuron, 44,* 547–555.

Plaut, D. C., & Booth, J. R. (2000). Individual and developmental differences in semantic priming: Empirical and computational support for a single-mechanism account of lexical processing. *Psychological Review, 107,* 786–823.

Poirier, M., & Saint-Aubin, J. (1995) Memory for related and unrelated words: Further evidence concerning the influence of semantic factors on immediate serial recall. *Quarterly Journal of Experimental Psychology: Human Experimental Psychology, 48*(A), 384–404.

Reeves, C., Schmauder, A. R., & Morris, R. K. (2000). Stress grouping improves performance on an immediate serial list recall task. *Journal of Experimental Psychology, 26,* 1638–1654.

Rohde, D. L. T. (1999). *Lens: The light, efficient, network simulator* (Tech. Rep. No. CMU-CS-99–164). Department of Computer Science, Carnegie Mellon University, Pittsburgh, PA.

Roodenrys, S., & Hinton, M. (2002). Sublexical or lexical effects on serial recall of nonwords? *Journal of Experimental Psychology: Learning, Memory, and Cognition, 28,* 29–33.

Rumelhart, D. E., Durbin, R., Golden, R., & Chauvin, Y. (1996). Backpropagation: The basic theory. In P. Smolensky, M. C. Mozer, & D. E. Rumelhart (Eds.), *Mathematical perspectives on neural networks* (pp. 533–566). Hillsdale, NJ: Erlbaum.

Rumelhart, D. E., & McClelland, J. L. (1986). *Parallel distributed processing: Explorations in the microstructure of cognition.* Cambridge, MA: MIT Press.

Ryan, J. (1969a). Grouping and short-term memory: Different means and patterns of grouping. *Quarterly Journal of Experimental Psychology, 21,* 137–147.

Ryan, J. (1969b). Temporal grouping, rehearsal, and short-term memory. *Quarterly Journal of Experimental Psychology, 21,* 148–155.

Saint-Aubin, J., & Poirier, M. (1999). Semantic similarity and immediate serial recall: Is there a detrimental effect on order information? *Quarterly Journal of Experimental Psychology: Human Experimental Psychology, 52*(A), 367–394

Schweickert, R. (1993). A multinomial processing tree model for degradation and redintegration in immediate recall. *Memory & Cognition, 21,* 167–175.

Shallice, T., & Butterworth, B. (1977). Short-term memory impairment and spontaneous speech. *Neuropsychologia, 15,* 729–735.

Tanji, J., & Shima, K. (1994, September 29). Role for supplementary motor area cells in planning several movements ahead. *Nature, 371,* 413–416.

Vallar, G., & Baddeley, A. D. (1984). Phonological short-term store, phonological processing and sentence comprehension: A neuropsychological case study. *Cognitive Neuropsychology, 1,* 121–141.

Van Bon, W. H. J., & Van der Pijl, J. M. L. (1997). Effects of word length and wordlikeness on pseudoword repetition by poor and normal readers. *Applied Psycholinguistics, 18,* 101–114.

Vousden, J. I., & Brown, G. (1998). To repeat or not to repeat: The time course of response suppression in sequential behavior. In D. W. Bullinaria, D. Glasspool, & G. Houghton (Eds.), *Proceedings of the 4th Neural Computation and Psychology Workshop: Connectionist Representations* (pp. 301–315). London: Springer-Verlag.

Warrington, E. K. (1982). The double dissociation of short- and long-term memory deficits. In L. S. Cermak (Ed.), *Human memory and amnesia* (pp. 61–76). Hillsdale, NJ: Erlbaum.

White, O., Lee, D., & Sompolinsky, H. (2004). Short-term memory in orthogonal neural networks. *Physical Review Letters, 9,* 148102.

Wickelgren, W. A. (1966). Associative intrusions in short-term recall. *Journal of Experimental Psychology, 72,* 853–858.

Wickelgren, W. A. (1967). Rehearsal grouping and hierarchical organization of serial position cues in short-term memory. *Quarterly Journal of Experimental Psychology, 19,* 97–102.

Williams, R. J., & Zipser, D. (1995). Gradient-based learning algorithms for recurrent neural networks and their computational complexity. In Y. Chauvin & D. E. Rumelhart (Eds.), *Backpropagation: Theory, architectures, and applications* (pp. 433–486). Hillsdale, NJ: Erlbaum.

Xiaohui, X., & Seung, S. (2003). Equivalence of backpropagation and contrastive Hebbian learning in a layered network. *Neural Computation, 15,* 441–454.

Zipser, D., & Anderson, R. A. (1988). A back-propagation programmed network that simulates response properties of a subset of posterior parietal neurons. *Nature, 331,* 679–684.

Zipser, D., Kehoe, B., Littlewort, G., & Fuster, J. (1993). A spiking network model of short-term active memory. *Journal of Neuroscience, 13,* 3406–3420.

Appendix

In the section Initial Analyses: How the Trained Model Works, we summarized the conclusions of a set of analyses that were aimed at elucidating the functioning of the model. The details of those analyses are presented here.

The superpositional nature of the model's sequence representations was established through a regression analysis, focusing on the hidden unit activations of the model presented in Simulation 2. The patterns analyzed were generated by running the model on a set of 5,040 four-item lists, which together contained all orderings of a limited set of items (five confusable items and five mutually nonconfusable items). For each target sequence, the pattern of hidden unit activation was recorded on the first step of recall, a point where the entire four-element list must be represented. The activation of each unit within these sequence representations was then linearized, applying the inverse of the sigmoidal unit activation function. A linear regression was then performed on the transformed activation of each unit, across all lists, with an indicator variable for each specific letter-position conjunction.

(Note: Performing linear regression on linearized unit activations was equivalent to performing logistic regression on raw unit activations. We describe the analysis in terms of linear regression because it is convenient, in what follows, to express our results as they apply to linearized unit activations. Although the regression results are described in terms of what they reveal about the model's internal representations, it should be noted that at a technical level, such descriptions apply directly to the linearized activation patterns that formed the basis of the regression. Clearly, our conclusions from the linearized activations capture the model's representational scheme only if the contribution of the nonlinearity involved in the hidden units' activation function can be legitimately ignored. Three observations support this assumption. First, the cosine analysis illustrated in Figure 5 indicates that the model's internal representations relate sensibly to the output weights, even if linearized. Second, following encoding, the vector representing a specific list element remained more or less constant in length or magnitude over the remaining steps of the trial [see Footnote 5]. This indicates that the representations of list elements were not compressed at any stage of processing, as would be expected if nonlinearities in the hidden layer were being heavily relied on. Finally, in separate simulations, we observed that the model can be successfully trained using linear hidden units. Indeed, under these conditions, the model continued to display primacy, recency, typical transposition gradients, and other behavior patterns shown by the version of the model we report in detail here.)

For every unit in the hidden layer, this regression accounted for over 98% of activation variance. Because the indicator variables included in the regression model correspond only to individual list elements (item–position conjunctions), this finding implies that the hidden representation can be understood as a linear superposition of activation vectors representing individual list elements. The result also indicates that element representations are unaffected by list context; they do not depend on which other elements in the same target list. If there were such a dependency, then our regression analysis would not have captured such a large proportion of the activation variance.

The Initial Analyses section presents data concerning the similarity relations among element vectors (see Figure 4). The element vectors used in those comparisons were identified on the basis of the regression analysis just described. For each item–position conjunction, the regression yielded a coefficient for each hidden unit, reflecting the way in which the presence of the conjunction in a target list impacted that unit's activation. When the coefficients for all of the hidden units, for any specific item–position conjunction, were concatenated, they formed a vector indicating the way that the conjunction was represented inside the overall sequence encoding, that is, an element vector. The comparisons discussed under Initial Analyses were based on a set of 40 element vectors (crossing 10 items with 4 positions) derived from one instance of the model described in Simulation 2.

Note that, in principle, the pattern of similarities in Figure 4 could arise from a coding scheme in which some hidden units code for item (independent of position) others code for position (independent of item), and still others code exclusively for a single item–position conjunction. However, further analysis indicated that the vast majority of hidden units showed coding properties intermediate among these extremes. This was shown by a multivariate analysis of variance (MANOVA), testing the degree to which position and item could predict the activation of each

hidden unit. Each dependent variable was the activation of a specific hidden unit, as this varied across all specific item–position conjunctions. The values here were drawn from the element vectors already described. That is, at each hidden unit, the data being fit were not literal activation values but instead were the portions of unit activation attributable to the occurrence of a specific item at a specific position. Factors were included for item and for serial position, with no interaction term.

This analysis yielded two informative results. First, the vast majority of hidden units showed main effects of either position or item, or both. This indicates that very few, if any, units coded in an exclusive manner for a single item–position conjunction. Second, the *r*-squared values for individual units, from this MANOVA, tended to be far from maximal (M = 0.72, range = 0.25–0.99). This indicates that although most units carried some information about item (independent of position) or position (independent of item), most units also coded conjunctively for item and position. Together, these findings indicate the use of a graded, coarse conjunctive code, as characterized by Cer and O'Reilly (in press).

Figure 5 in the Initial Analyses section displays data concerning the relationship between element vectors and output weights over successive steps of processing. The element vectors analyzed here were generated by repeating the earlier regression analysis at each step of encoding and of recall. This yielded a set of 320 element vectors, one for each of 10 items, at each of 4 positions, and at each of 8 time steps. Each of these vectors was compared with the vector of weights connecting the hidden layer to the (Feature 2) output unit representing the same item.

Received August 18, 2004
Revision received August 24, 2005
Accepted August 31, 2005 ■