Psychonomic Bulletin & Review
1997, 4 (2), 145-166

A model for recognition memory: REM—retrieving effectively from memory

RICHARD M. SHIFFRIN and MARK STEYVERS
Indiana University, Bloomington, Indiana

A new model of recognition memory is reported. This model is placed within, and introduces, a more elaborate theory that is being developed to predict the phenomena of explicit and implicit, and episodic and generic, memory. The recognition model is applied to basic findings, including phenomena that pose problems for extant models: the list-strength effect (e.g., Ratcliff, Clark, & Shiffrin, 1990), the mirror effect (e.g., Glanzer & Adams, 1990), and the normal-ROC slope effect (e.g., Ratcliff, McKoon, & Tindall, 1994). The model assumes storage of separate episodic images for different words, each image consisting of a vector of feature values. Each image is an incomplete and error prone copy of the studied vector. For the simplest case, it is possible to calculate the probability that a test item is "old," and it is assumed that a default "old" response is given if this probability is greater than .5. It is demonstrated that this model and its more complete and realistic versions produce excellent qualitative predictions.

The authors have been working with Jeroen Raaijmakers and Lael Schooler[1] to develop a new theory of memory that borrows certain elements from the SAM model (e.g., Gillund & Shiffrin, 1984; Raaijmakers & Shiffrin, 1980, 1981; Shiffrin, Ratcliff, & Clark, 1990), the MINERVA model (Hintzman, 1988), the model of Humphreys, Bain, and Pike (1989), and a number of other extant models. Its initial development borrows a conception put forward in John Anderson's Rational model (Anderson, 1990) and its application to explicit recognition is similar to a model developed in parallel and independently by McClelland and Chappell (1994). The new theory is aimed to predict phenomena of explicit and implicit, and episodic and generic (semantic), memory. It is termed REM (standing for retrieving effectively from memory).

The main goal of this article, however, is much more limited: to present in detail the portion of the theory that is needed to generate predictions for some basic phenomena of explicit, episodic, recognition memory. The restriction to explicit recognition may seem severe, but there exists a large and reliable database in this case. In addition, the basic mathematical form that underpins the general theory is justified by derivations for this case. Finally, the model for explicit recognition is both simple and powerful, able to predict qualitatively a number of basic phenomena that have proved difficult to handle within extant models. These phenomena include (1) list strength—the fact that strengthening some list items does not harm and may help recognition of other list items (e.g., Ratcliff, Clark, & Shiffrin, 1990; Shiffrin et al., 1990); (2) the mirror effect—a factor that improves recognition and simultaneously raises the hit rate and lowers the false alarm rate (e.g., Glanzer & Adams, 1990); and (3) the NRS effect (normal ROC slope, also called the *z*-ROC slope by some)—the fact that the ratio of the spread of the distractor distribution to the spread of the target distribution is less than one and does not change markedly with variations in length, strength, and word frequency (e.g., Ratcliff, McKoon, & Tindall, 1994). None of the extant models of recognition memory have proved fully adequate to deal with even this limited set of phenomena. For example, the SAM model gives perhaps the simplest and most complete account of the major phenomena of explicit recognition and recall, but even its most recent variant, designed to handle the list strength findings, provides no convincing account of mirror effects and has a variety of potential problems when applied to NRS effects.

For ease of exposition, we will begin by applying a simplified form of the REM model to episodic recognition memory. Although this is a stripped down version of REM, its assumptions allow precise derivations, and its structure is sufficient to predict the basic phenomena of recognition memory. We will defer discussion of basic recognition phenomena until we introduce their prediction by the model. After that we will introduce more complex and more realistic versions of REM, applying each of them to the same set of recognition phenomena. The relation of REM to extant models will begin our concluding discussion, followed by a brief sketch of the way the recognition model fits into the more general theory, and the directions of some extensions.

A computer program written in C for simulating REM 1, 2, and 3 is available upon request. The author is indebted to the University of Amsterdam, the graduate research institute EPOS (sponsored by the University of Amsterdam, the University of Leiden, and the Free University of Amsterdam), and Indiana University, which helped support the first author during a sabbatical at the University of Amsterdam in 1994–1995 when this research was initiated. The research was also supported by NIMH Grant 12717. Reprint requests should be sent to R. M. Shiffrin, Psychology Department, Indiana University, Bloomington, IN 47405 (e-mail: shiffrin@indiana.edu).

REM.1 The Basic Model

Representation

Memory consists of separate images; each is represented as a vector of feature values, V. The absence of knowledge about a feature is represented by the value zero, and knowledge of a feature is represented by positive integers, these values differing in their environmental base rates. The distribution of environmental base rates has been chosen for simplicity to be geometric,[2] based on a parameter, g:

$$P[V=j] = (1-g)^{j-1}g, \quad j = 1, \ldots, \infty. \quad (1)$$

The lexical/semantic representation of a word consists of w non-zero feature values (in our simulation, w was set to 20, a number small enough to allow the simulations to operate at reasonable speeds); different features within a word, and different words, have feature values that are generated independently according to Equation 1. Although word frequency effects will be introduced and discussed later, rigor requires us to list here the following assumption: High-frequency words have more common feature values than low-frequency words do, and hence they are generated with a higher value of g, g_{H}, than are low-frequency words, g_{L}.

Storage

An episodic image is stored as a result of studying a word in a list; the stored episodic vector is an incomplete and error prone copy of the studied word vector. Each unit of time that a word is studied, there is a probability u^* that something will be stored for each feature, given that nothing has yet been stored for that feature (once a value is stored, it is not changed thereafter).[3] If something is stored for a feature, its value is copied correctly from the studied vector with probability c; with probability $1-c$, the stored value is chosen randomly according to Equation 1 (allowing the possibility of accidentally choosing the correct value for storage).[4] In our simulations, we have allowed 10 units of storage time for slow (or strong, or multiple) presentations, and 7 units of storage time for fast (or weak, or single) presentations. Note that repetitions of a word within a list are treated as a single slow presentation, resulting in storage of a single episodic image.[5]

Retrieval

The probe vector consists of a word vector that either has been studied (a target) or has not been studied (a distractor). The probe (a vector of 20 feature values) is matched in parallel to the episodic images of the n words on the list. The resultant data, D, is a set of D_j, $j = 1, \ldots, n$. D_j is obtained by aligning the feature values of the probe and image j, and noting values of those positions whose values match and those positions whose values mismatch (ignoring feature positions where the image contains no value). An episodic image that has been stored during an earlier presentation of the word currently presented is termed an s-image (s for *same*). An episodic image that has been stored during presentation of any word other than the word currently presented is termed a d-image (d for *different*).

The critical part of the matching process between a probe and image I_j consists of calculation of a likelihood ratio, λ_j: the probability that D_j would have been observed if image I_j was an s-image, divided by the probability that D_j would have been observed if image I_j was a d-image. This calculation takes into account the environmental base rates of the feature values making up D_j (based on the long-run environmental base rates, but not knowledge of the experimental manipulations, such as whether the list contained high- or low-frequency words). λ_j plays the role in REM that the "product of retrieval strengths" or "activations" plays in the SAM model (e.g., Gillund & Shiffrin, 1984; Raaijmakers & Shiffrin, 1980). In addition, λ_j takes the role of activations or match values in global matching models such as MINERVA (e.g., Hintzman, 1988), the MATRIX model (e.g., Humphreys, Pike, Bain, & Tehan, 1989), TODAM (e.g., Murdock, 1982), and CHARM (e.g., Eich, 1982). A related likelihood calculation is utilized in a similar model developed in parallel to the present one—the recognition model originally reported in the 1994 Psychonomic Society meetings by McClelland and Chappell (1994); we will say a bit more about this model in the discussion. Justification for use of the λ_j in REM is based on the following Bayesian derivation.

Bayesian Decision

Based on the set of D_j, the system uses Bayes rule to calculate the probability that the test word is "old" (i.e., represented in the set of activated images), as opposed to "new." It is assumed that the calculated probability (or odds) corresponds to the feeling of "familiarity" of an item and is used to produce a recognition decision.[6] As a default, in the absence of payoffs, instructions, or other factors providing a reason to do otherwise, an "old" decision is given if this probability is greater than .5 (i.e., the odds are greater than 1.0; see below, under Calculations). This calculation is a normative one. The decision that is made is the best one possible, given the data available and the assumptions regarding storage error. It is for this reason that the word *effective* is used in the REM description. This general idea has been used in signal-detection–based theories of perception for many years, and in memory recently by John Anderson (e.g., 1990).

We do not necessarily argue that the system calculates probabilities in principle, only that the system has evolved to retrieve efficiently. Furthermore, we would find it hard to believe that a real system evolved to handle the demands of an artificial empirical recognition paradigm. Nonetheless, a real system may well have evolved to find matches to the memory probe, and our normative model may therefore provide important insights into the functional form of the calculations that such a system uses to carry out retrieval. It is just this point that gives the present approach an advantage over most previous process models of memory. The assumptions of previous models have been chosen for practical or functional reasons. The assumptions of SAM, for example (in which each cue has a strength to a given image, and image activation is the product of these strengths), were chosen to have certain desired properties, such as interactive cue combination. However, the mathematical form chosen for SAM was fairly arbitrary, and many other forms could have had similar desirable properties. The probability approach taken here provides a principled reason for using a particular functional form.

Calculations

It is convenient to calculate the odds in favor of an old over a new test item (the odds equal the probability that the test item is old divided by the probability that the test item is new). The sequence of derivations is straightforward, and the derivations for Equations 2, 3, and 4A and 4B are given in Appendix A. Here, we simply give the final results. Let Φ be the odds. It can be shown that:

$$\Phi = \frac{1}{n}\sum_{j=1}^{n}\lambda_j . \qquad (2)$$

Let V_k denote the value of the kth feature in the test word. Let V_{kj} denote the value of the kth feature in the jth image. For a given image, let M represent the set of feature indices (i.e., their positions in the vector) for which the nonzero feature values match the probe; let Q be the set of feature indices for which the nonzero feature values mismatch the probe. Let $P_{km}(i)$ = the probability that feature value i would have been stored in position k given that this image is an s-image, the probe feature in position k has value i, and some value is stored. Let $P_{kq}(i)$ = the probability that feature value i would have been stored in position k given that this image is an s-image, the probe feature in position k has some value other than i, and some value is stored. Let $P_{kd}(i)$ = the probability that feature value i would have been stored in position k given that this image is a d-image and some value is stored. Then, for any given image (the subscript denoting the image is suppressed for simplicity),

$$\lambda = \prod_{k\in M}\left[\frac{P_{km}(i)}{P_{kd}(i)}\right]\prod_{k\in Q}\left[\frac{P_{kq}(i)}{P_{kd}(i)}\right]. \qquad (3)$$

The λs can also be written in terms of the parameters c (the probability of copying a stored feature correctly) and g (determining the geometric base rates for feature values):

$$\lambda_j = \prod_{k\in M_j}\left[\frac{c+(1-c)g(1-g)^{V_{kj}-1}}{g(1-g)^{V_{kj}-1}}\right]\prod_{k\in Q_j}\left[1-c\right]. \qquad (4A)$$

It is convenient to rewrite Equation 4A in the following form. Let n_{jq} = number of all nonzero mismatching features in the jth image, regardless of value, and n_{ijm} = the number of nonzero features in the jth image that match the probe value and have value i. Then

$$\lambda_j = (1-c)^{n_{jq}}\prod_{i=1}^{\infty}\left[\frac{c+(1-c)g(1-g)^{i-1}}{g(1-g)^{i-1}}\right]^{n_{ijm}}. \qquad (4B)$$

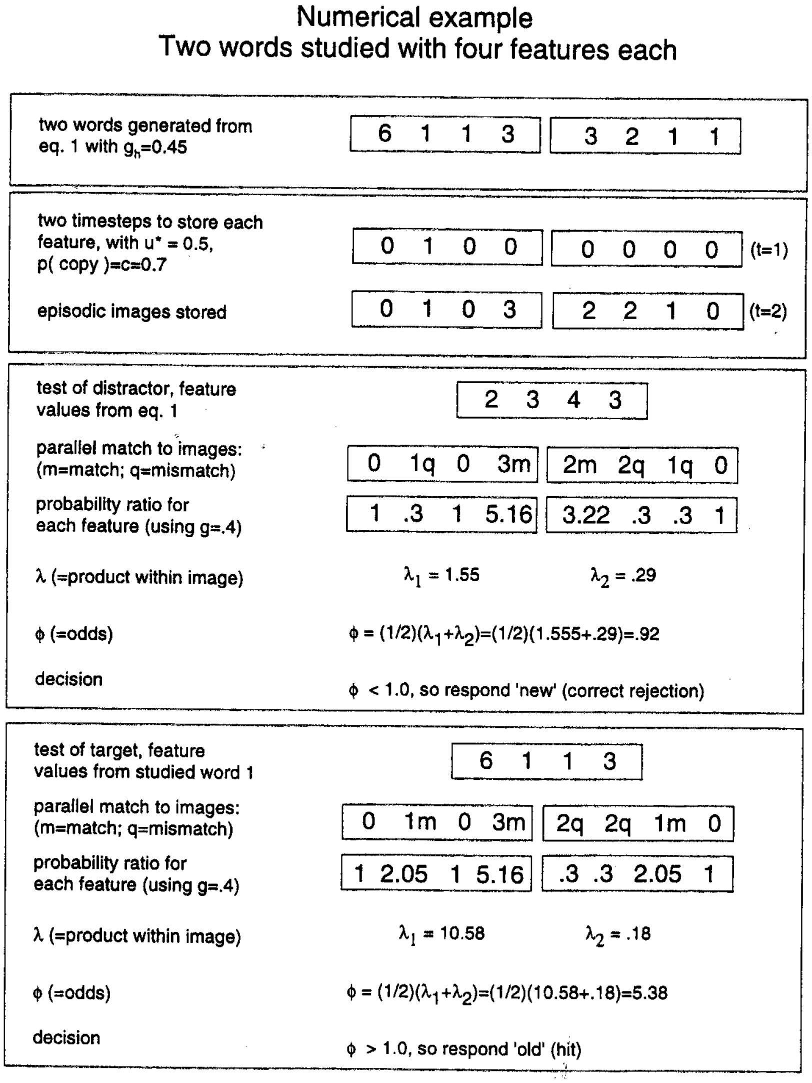
It should be noted that in Equations 4A and 4B, only the values of the matching features turn out to be relevant; for mismatching features, neither the value in the image or the probe is utilized.

These results are remarkably simple, consisting (in Equation 2) of a sum of the likelihood ratios for each image, divided by the number of images. It might be thought that such a simple result would lend itself to explicit derivations, but this turns out not to be the case, because the exponents on the terms in Equation 4 are random variables with complex distributions. Nonetheless, Equations 2 and 3 or 4 lend themselves readily to simulation methods.

A Numerical Example

It may clarify matters to work through a simple example in exactly the sequence of events utilized in the simulation. Assume a word is represented by four content features, and that there are two such words presented in a list. The two words are assumed to be of high frequency, and hence have feature values generated with g_H = .45. In the example, we assume that there is enough presentation time so that each feature of each word gets 2 storage attempts, and the probability of storing a value for a feature on each attempt is u^* = .5. The probability of copying correctly a feature that is being stored is c = .7. Later, one distractor, and one target, are presented successively for *old–new* recognition judgments. The calculations of Equation 4 are based on the long-run base rate value of g = .4.

Figure 1 illustrates the sequence of events. The first row shows the vectors representing the two words; each of these feature values was generated independently from Equation 1, with g_H = .45. The next two rows show the results of the two attempts at storage for each word. In all, two features were stored for Word 1 (both correct) and three for Word 2 (two correct and one incorrect—a new choice from Equation 1 resulted in the incorrect value of 2 being stored for Feature 1). The first test word is a distractor. Its features are generated with Equation 1, with g = .45, since we assume that all study and test words are high frequency. The resultant vector is matched in parallel to the two episodic images; for Image 1, there are one matching feature (with value 3) and one mismatching feature (remember that the values of mismatching features do not enter into the calculations); for Image 2, there are one matching feature (with value 2) and two mismatching features. Note that these factors are calculated with the value of g known to the system: .4. Each feature contributes a factor to the likelihood ratio (as indicated in Equation 3, or Equation 4A), and these factors are given in the next row; note that any mismatch contributes a



Numerical example
Two words studied with four features each

two words generated from eq. 1 with g_h=0.45	6 1 1 3	3 2 1 1	
two timesteps to store each feature, with u* = 0.5, p(copy)=c=0.7	0 1 0 0	0 0 0 0	(t=1)
episodic images stored	0 1 0 3	2 2 1 0	(t=2)
test of distractor, feature values from eq. 1	2 3 4 3		
parallel match to images: (m=match; q=mismatch)	0 1q 0 3m	2m 2q 1q 0	
probability ratio for each feature (using g=.4)	1 .3 1 5.16	3.22 .3 .3 1	
λ (=product within image)	λ_1 = 1.55	λ_2 = .29	
φ (=odds)	$\phi = (1/2)(\lambda_1+\lambda_2)=(1/2)(1.555+.29)=.92$		
decision	φ < 1.0, so respond 'new' (correct rejection)		
test of target, feature values from studied word 1	6 1 1 3		
parallel match to images: (m=match; q=mismatch)	0 1m 0 3m	2q 2q 1m 0	
probability ratio for each feature (using g=.4)	1 2.05 1 5.16	.3 .3 2.05 1	
λ (=product within image)	λ_1 = 10.58	λ_2 = .18	
φ (=odds)	$\phi = (1/2)(\lambda_1+\lambda_2)=(1/2)(10.58+.18)=5.38$		
decision	φ > 1.0, so respond 'old' (hit)		

Figure 1. A numerical example illustrating the storage of two words, and the operation of recognition for a distractor and a target, for the model designated REM.1. See text for a description of the entries.

value of $1-c = .3$, whereas matches contribute a value that increases with the value of the feature (since high values are unlikely and hence are highly diagnostic; they tend not to match by chance). Although the derivations of Equations 3 and 4A are given in Appendix A, it may be useful to show where the values of 5.16 and .3, say, arise in our example. The matching value of 3 must have occurred by chance for a d-image and could have occurred by chance (in the case of a storage error) for an s-image; in both cases the probability of a chance match is $g(1-g)^{(3-1)} = .4(.6)^2 = .144$. For an s-image, the probability of a matching value of 3 must include the possibility of correct storage of the feature value: $c+(1-c)(.144) = .7 + .3(.144) = .7432$. The ratio is $.7432/.144 = 5.16$. In the case of a mismatch (say, with values of 1 and 3, as in our example), we know that the stored value could not have been a copy of the test value, and must have been a random choice according to Equation 1, regardless of whether this was an s-image or a d-image. The probability of random storage for a d-image is 1.0, and for an s-image is $1-c = .3$, regardless of the values involved, giving a ratio of .3.

The overall likelihood ratio for an image is calculated from the product of these feature ratios (corresponding to the use of Equation 4A or 4B), and these λs are given in the next row. The λs are then inserted into Equation 2, producing odds given in the next row that are less than 1.0, so a "new" response is given. This response is correct, since a distractor had been tested. Note that this correct rejection is made even though the likelihood ratio for the first image alone is greater than 1.0, because the division by n (2 in this case) appropriately and correctly takes into account the increased probability of an image matching by chance as n increases. This whole process repeats for the test of the target (Studied Word 1). In this case, the two matching features for Image 1 provide quite strong evidence, producing a high likelihood ratio (10.58). Even after averaging in the low likelihood ratio for the second image (0.18), the odds are above 1.0, so an "old" response is given. This response is again correct (a "hit").

In the actual simulations, longer study and test lists are used, and the process of presenting a list and of testing targets and distractors is repeated a large number of times (usually enough to produce 20,000 data points per condition). With this number of pseudodata points, the distributions of the number of matching features and their values, and the number of mismatching features, are quite well specified, and the resultant predictions derived from analyzing the pseudodata are quite accurate.

A Note on Performance Measures

It is often assumed that recognition is based on a measure like summed activation, or familiarity. The familiarity distributions for targets and distractors are usually assumed to be normally distributed (with means μ_t and μ_d, and standard deviations σ_t and σ_d), as illustrated in Figure 2a. Whatever the shape of the distributions, the subject is assumed to select a criterion C along the familiarity axis, and respond "old" if the observation on a trial is greater than the criterion, and "new" otherwise. The hit probability is the probability of saying "old" when a target is presented; in Figure 2a, this is the shaded area above the criterion under the target distribution. The probability of a false alarm is the probability of saying "old" when a distractor is presented; in Figure 2a, this is the shaded area above the criterion under the distractor distribution.

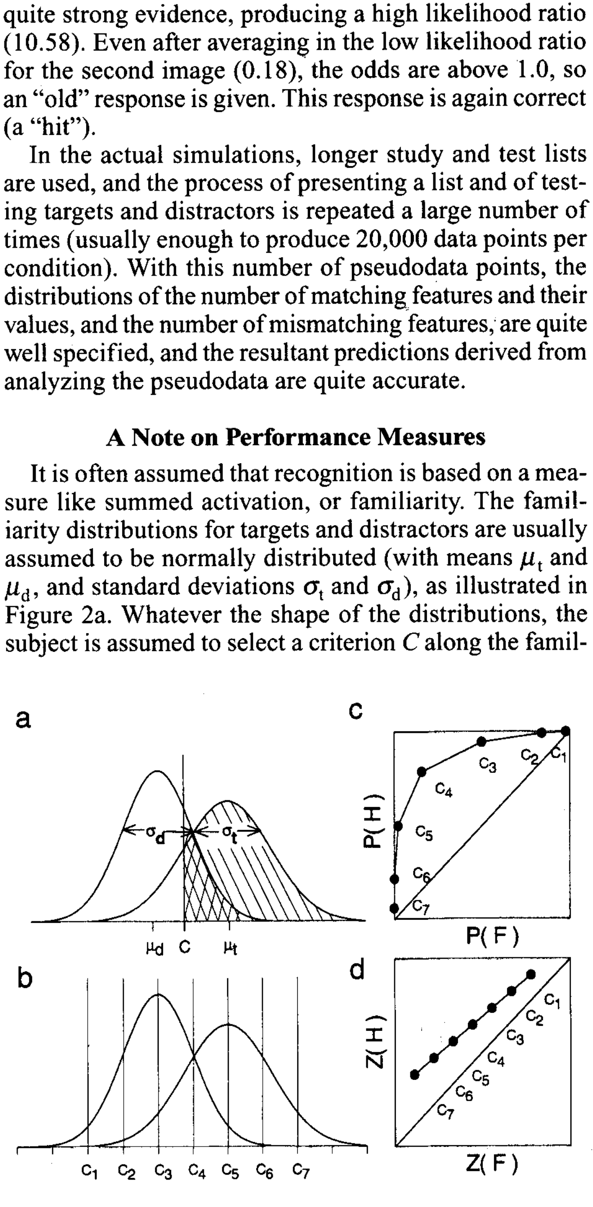


Figure 2. Analysis of recognition memory according to signal-detection theory. Panel a: normally distributed target and distractor distributions of familiarity, with means μ_t and μ_d, and standard deviations σ_t and σ_d, and response criterion, C. P(H) is the area under the target distribution to the right of C; P(F) is the area under the distractor distribution to the right of C. Panel b: several response criteria, C_1 to C_7. Panel c: receiver-operating characteristic function (ROC) with points corresponding to the criteria in panel b. Panel d: ROC in panel c replotted on gaussian axes (z-ROC); the slope of the z-ROC is the standard deviation ratio for distractors to targets, equal to σ_d/σ_t. When empirical slope estimates are obtained, or when a similar analysis is carried out for distributions that are not normal, the best fitting slope is termed the NRC (for normal ROC slope).

The usual performance measure is d': the distance between the means of the target and distractor distributions divided by the standard deviation of the distractor distribution. If the standard deviations for the target and distractor distributions are equal, and if the distributions are normal, one can derive d' from the observed hit and false alarm probabilities by using normal probability tables. Even when the distributions are not assumed or predicted to be normal and the standard deviations are not assumed or predicted to be equal, it is common practice to compute a d' value in the standard way as a convenient measure of performance. (As we shall see, REM does not predict normality or equality, but predicts that d' calculated in the standard way is higher when the vector is longer, the number of storage attempts is larger, c is larger, or g is smaller).

The subject's choice of a criterion can be manipulated experimentally, by use of differential payoffs, or instructions to use confidence ratings. Figure 2b shows several such criteria, spaced along the familiarity axis. Each criterion gives rise to a hit rate and a false alarm rate. These can be plotted as an ROC curve in a graph with hit probability along one axis and false alarm probability along the other axis, as in Figure 2c. It is useful to plot such curves on a graph with normally transformed axes, as in Figure 2d. If the underlying distributions are normal, this curve will be linear, with a slope giving the ratio of the distractor standard deviation to the target standard deviation. Even when the distributions are far from normal, the normal ROC curve can be close to linear (as is the case for REM), and in such cases a linear regression is fit to the curve and a slope obtained. This slope will not usually equal the actual ratio of standard deviations when the distributions are not normal, so we will term the slope the NRS (for normal ROC slope; the term z-ROC slope has been used; but we decided to use new terminology, lest there be any tendency to infer that the slope may represent a ratio of standard deviations).

Applications

Despite the simplicity of this version of the REM model, it can provide enlightening predictions for basic phenomena of recognition memory. This section gives typical data along with predictions of the model. The simulations use the following parameter values: w = 20 word features, u^* = .04 per storage attempt, t = 10 storage attempts for strong words, and t = 7 storage attempts for weak words, g_H = .45 for high-frequency words, g_L = .325 for low-frequency words, c = .7, and g = .4. These values were not chosen to match data quantitatively via some sort of parameter search. We checked a few sets of values and chose these because they give d' values that align very roughly with the broad and reliable trends typical of recognition data. Our aim in this article is to illustrate the degree to which the basic structure of the model predicts such data patterns.

The figures that show data and predictions of the current model and its later variants and extensions are arranged as follows: A given figure shows the prediction for some paradigmatic variation, such as list length. The rows correspond to different phenomena that vary when list length, say, is varied: row 1 gives performance measured by d'. Row 2 gives the "hit" probability [p(old) for targets, termed P(H)], and the false alarm probability [p(old) for distractors, termed P(F)]. Row 3 gives the normal ROC slope, NRS. Within a row, the first panel gives representative data (taken from the literature), and the subsequent panels give the predictions derived from various versions of the model. Only the second panel in each row (i.e., column 2) is relevant for the present version, REM.1.

List Length

1. Performance (i.e., d') is quite reliably lower for longer lists (Figure 3, row 1, panel 1 shows typical findings, in this case taken from Murnane & Shiffrin, 1991b, Experiment 1; the abscissa refers to the numbers of five-word sentences that were studied: 10, and 30). These findings occur even when care is taken to eliminate any contamination of the results by retrieval from short-term memory, inhibition increasing during the test period, or differential study–test lags (e.g., Gronlund & Elam, 1994; Murnane & Shiffrin, 1991b). All extant models predict such a result, for various and differing reasons.

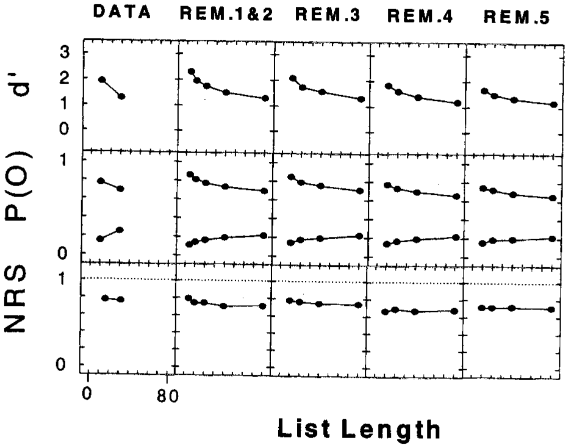


Figure 3. List length data and predictions. List lengths for models = 4, 10, 20, 40, 80 (single words, whether or not words are presented in pairs). Row 1, d'; row 2, P(H) (upper points) and P(F) (lower points); row 3, SDR. Column 1, rows 1 and 2, data from Murnane & Shiffrin (1991b, Experiment 1; n = 10 and 30 five-word sentences); row 3, data from Ratcliff et al. (1994, Experiment 3; n = 8 and 32 word pairs). Columns 2–5: predictions from various REM models (see text).

REM predicts list length effects basically because each additional list word introduces an additional chance of matching the test word well by accident, regardless of whether the test word is a target or distractor (a more technical answer will be given shortly). The REM predictions for list lengths (number of words) of 4, 10, 20, 40, and 80, for strong high-frequency words (i.e., t = 10, and g_H = .45), are given in Figure 3, row 1, panel 2.

2. A mirror effect is usually seen: When performance is lower, in this case for longer lists, the hit rate drops and the false alarm rate rises (Figure 3, row 2, panel 1 shows typical results, from Murnane & Shiffrin, 1991b, Experiment 1). Extant models are capable of predicting such a result because they are free to adjust the criterion for different lists. They usually provide no reason why the subject should adjust the criterion in the required way (cf. Hintzman, Caulton, & Curran, 1994).

REM predicts a mirror effect without adjusting the criterion for different lists. We assume that, in the absence of a good reason not to do so, the criterion is left at the default value derived from the Bayesian approach: odds of 1.0. Why then do changes in length produce a mirror effect? The short answer is that the Bayesian approach is designed to produce mirror effects, because the decision is based on the calculation of the odds that the test word is old, and half the test words are old. The long answer is moderately technical and will be deferred until the distributions of likelihood ratios are discussed in the next section. The REM predictions are given in Figure 3, row 2, panel 2.

3. The normal ROC curve is usually linear, with a slope (i.e., of the NRS) of less than 1 (usually about .8). There is some debate about the constancy of these slopes as conditions vary (see, e.g., Glanzer, Adams, Iverson, & Kim, 1993; Glanzer & Kim, 1997; Ratcliff et al., 1994; and Ratcliff, Sheu, & Gronlund, 1992). The issue is of some importance, because current models predict either slopes of 1.0, or slopes of less than 1.0 that change substantially with list length (see, e.g., Gronlund & Elam, 1994).[7] Although changes in NRS are at most small in magnitude (almost always .1 or less), whatever changes there are may depend on the amount of training and testing given a subject. For subjects receiving only moderate training or less, the conditions that produce higher accuracy usually produce a lower NRS (with the apparent exception of repetition; see Glanzer & Kim, 1997). This pattern appears to hold for length also: shorter lengths produce a higher d' and lower NRS (Gronlund & Elam, 1994). In any event, we illustrate NRS values as a function of length in Figure 3, row 3, panel 1 with data from Ratcliff et al. (1994, Experiment 3; n = 8 and 32 word pairs).

The REM predictions are shown in the second panel of row 3 of Figure 3: the NRS is lower than 1.0 and has values that are fairly constant at about the level seen in the data. It is critical to note that the REM predictions are derived in the same way that the empirical NRS is obtained: the odds distributions for targets and distractors are obtained via simulation, several criteria are assumed (the natural logarithms of the criteria are: −1.0, −0.8,

−0.7, −0.2, 0, 0.2, 0.7, 1.0, 1.25, 1.75, 2.5), a normal ROC is plotted, a regression line is fit, and the NRS is the slope of this line.

To understand the basis for the NRS and mirror effect predictions better, it helps to examine the shape of the distributions of the likelihood ratios and the odds. The distribution of λ for a d-image has a mean of 1.0 (trivial to prove, as demonstrated in Appendix A), but is extremely skewed toward large values, so that the mode, median, and most of the area is below 1.0. The distribution of λ for s-images has a mean greater than 1.0 (the mean of $1/\lambda$ is 1.0, shown in Appendix A), has an area to the right of 1.0 that is greater than .5, and it is even more skewed toward large values. These distributions are so highly skewed that their graph is not visually enlightening, and we therefore plot them on a log scale. Figure 4A gives results for a list length of 20, and the parameters of Figure 3 (strong, high-frequency words). The distributions on a log scale are still skewed toward the high values, especially for the s-images. (The irregular pattern is due to the use of 20 features, with the peaks corresponding to certain combinations of matching and mismatching feature values).

The distributions of the log odds are plotted in Figure 4b. For distractors, this is the log of the average of n independent samples from the d-image λ distribution; for targets, one of the d-image samples is replaced by an s-image sample. For distractors, the average (before taking a log) remains at 1.0, but the variance and skewing drop as n increases; asymptotically the distractor odds become normal, centered at 1.0. The change in skewing forces the portion of the odds distribution above 1.0 to start at a low value and approach .5 as n increases, accounting for the rise in false alarms with n. For targets, the contribution of the s-image sample is divided by n, so the target odds distribution starts with high skewing, a mean greater than 1.0, and an area to the right of 1.0 greater than .5, and very gradually moves down toward the distractor odds distribution in mean and shape as n increases, accounting for the drop in hits with n.

Figure 4b makes it visually clear that the spread of the target odds distribution is greater than the spread of the distractor odds distribution, consistent with the NRS findings. The predicted constancy with length shown in Figure 3 is unexpected at first glance, since a calculation of the theoretical ratio of standard deviations reveals a substantial rise with n.[8] However, the model predictions in the figure were derived in the same way as for actual data, by using different criteria and deriving corresponding hit and false alarm rates, as in Figure 4c, producing a normal ROC, as in Figure 4d, and fitting a regression line. This procedure is of course much less sensitive to the shape of the extreme tails than is the ratio of standard deviations.

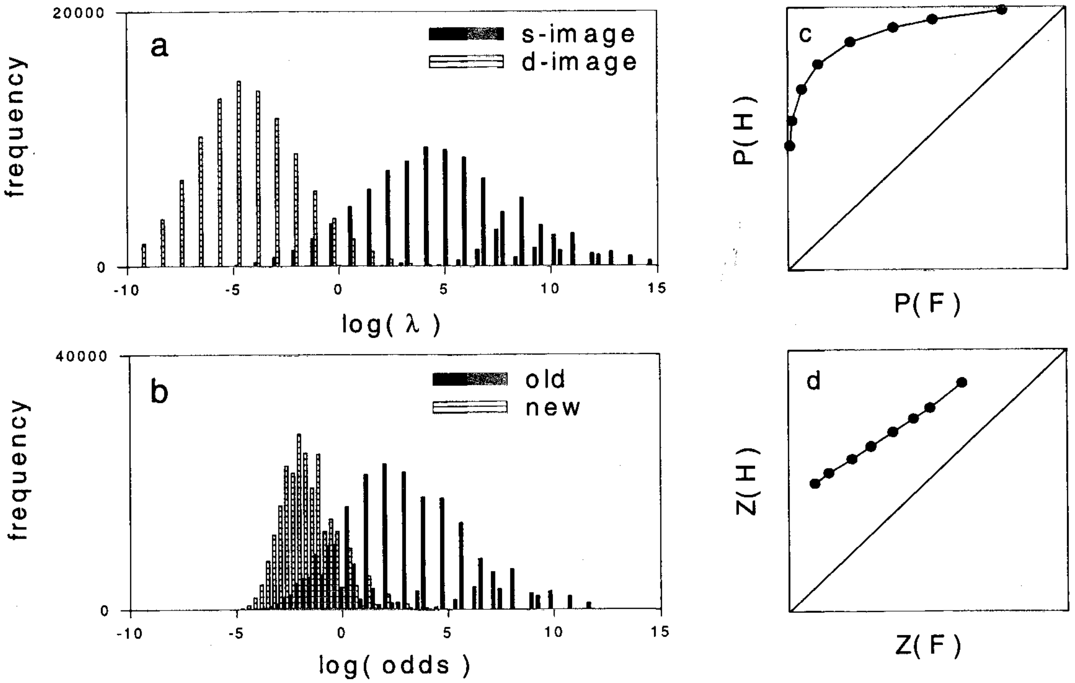


Figure 4. Panel a: distribution of log likelihood ratios for s-images (filled bars) and d-images (open bars), for the parameters used in Figure 3 (strong high-frequency words) and list length = 20. Panel b: distribution of log odds, corresponding to panel a. Panel c: ROC for the distributions in panels a and b: probability of a hit, *P*(H), graphed against probability of a false alarm, *P*(F). Panel d: *z*-ROC corresponding to panel c (slope = .72 = NRS); the points are those of panel c graphed on normal transformed axes.

Strength and List Strength

1. Slower presentations, or more repetitions, increase d', a universal finding illustrated in Figure 5, row 1, panel 1 with data from Ratcliff et al. (1990, Experiment 4). The relevant data here are given in the points within the panel that are labeled "pure weak" and "pure strong." *Pure weak* refers to words all presented quickly (or singly). *Pure strong* refers to words all presented slowly (or multiply, in spaced repetitions). All models predict such a main effect of strength, for various reasons. The simulation results in Figure 5, row 1, panel 2 are for a list length of 40, for high-frequency words ($g_H = .45$, but note that the calculations at retrieval are based on the base rate value of $g = .4$). Words on pure weak lists are assumed to have 7 storage attempts. Words on pure strong lists are assumed to have 10 storage attempts. REM reproduces the observed pattern of results because storage of more features tends to increase the likelihood ratio for an image of the test word (an s-image) and decrease the likelihood ratios for images of other words (d-images).

2. Stronger other items do not harm recognition of an item, and may help. Representative data from Ratcliff et al. (1990, Experiment 4) are shown in Figure 5, row 1, panel 1, in the comparison of the data points on the left side to each other, and in the comparison of the data points on the right side to each other. The central data points, labeled "mixed weak" and "mixed strong," come from a list with the same number of different words as for the pure lists, but half of the words are presented quickly, and half slowly (or twice). Weak words in mixed lists have stronger other words on their list than do words in pure weak lists; words in pure strong lists have stronger other words on their lists than do strong words in mixed lists. If strengthening items, or repeating items, has the same effect as does adding more items to the list, then stronger other items ought to hurt performance; but this does not occur.

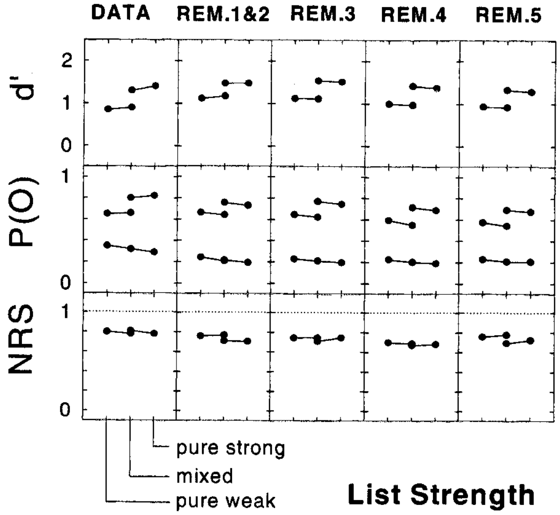


Figure 5. Strength and list strength data and predictions. Mixed weak: weak items from a list of half weak, and half strong words. Mixed strong: strong items from a list of half weak and half strong words. Row 1, *d'*. Row 2, *P*(H) (upper points) and *P*(F) (lower points); the two connected values of *P*(H) represent words of equal strength in lists of differing strength. Row 3, NRS. Column 1, rows 1 and 2: data from Ratcliff et al. (1994, Experiment 4; 2-sec and 5-sec study times). Row 3: data from Ratcliff et al. (1992, Experiment 1, 1-sec and 5-sec study times). Columns 2–5: predictions from various REM models (7 units of study for weak, and 10 units of study for strong). (See text.)

This finding has been termed the list strength effect by Ratcliff et al. (1990). (More accurate would be the terminology *list strength noneffect*, or even *list strength reverse effect.*) Shiffrin et al. (1990) demonstrated that a number of extant models do not predict this result, including the then current version of SAM, largely because these models predict that extra strength will have the same qualitative effect as do extra items. They amended SAM by adding a differentiation assumption: the tendency for a word cue to activate an image of a different word drops as the strength of storage of that image rises (the idea being that the dissimilarity of the cue and image becomes more evident when the image is stronger); this effect was offset by an increase in the tendency for the context cue to activate the same image. The structure of the REM model automatically produces an effect of differentiation, because stronger words have more features stored, and hence their images are less confusable with the test word, when the two are not the same word.

To apply REM to the mixed list paradigm, it is assumed that the mixed list of 40 high-frequency words comprises 20 weak words (7 storage attempts each) and 20 strong words (10 storage attempts each). It is assumed that the "system" is unaware of the experimental manipulation of strength, and that it carries out calculations as if the list words had been presented at equal (but unknown) strength (i.e., Equations 2, 3, and 4 are applied without alteration). The resultant REM predictions for d' are given in Figure 5, row 1, panel 2, and they appear quite close to the observed pattern.

The list strength findings have proved difficult to handle for most models. As has been stated above, Shiffrin et al. (1990) added a differentiation assumption to SAM to handle the results. McClelland and Chappell (1994) used a model similar to REM; it predicts the findings for similar reasons: stronger different items have more mismatching features and contribute less likelihood; this is of course a form of differentiation. Chappell and Humphreys (1994) presented a neural net model that will be described later; essentially, it predicts the effect because the association between the context and an item's representation is assumed to increase only at the first presentation, or only once regardless of study time. As opposed to differentiation, this assumption seems to have been added primarily to produce the observed list strength finding. All these approaches share an assumption that different words have separate memory traces, and that repetitions of words, even at spaced intervals, produce storage effects that are superimposed in the same trace.[9]

3. A mirror effect is typically observed: stronger items have higher hit rates and lower false alarm rates, as the

data in Figure 5, row 2, panel 1 from Ratcliff et al. (1990, Experiment 4) show. Since the criterion can vary between strong and weak lists, and since for mixed lists the distractors cannot be classified as "strong" or "weak," most models can predict this mirror effect relatively easily by assuming appropriate movement of the criterion between lists. However, the REM model predicts such a mirror effect without criterion adjustment, using its default setting of 1.0 for the odds, as is illustrated in the predictions shown in Figure 5, row 2, panel 2. The details of the predictions may be off just slightly, but the error is small enough so that the assessment of this comparison ought to await quantitative fits of the model.

4. The normal ROC slope (NRS) is less than one. In the literature, any variation in slope with strength is small in magnitude. Glanzer and Kim (1997) found a drop in slope of about .1 as strength moved from weak to strong, possibly related to the fact that their d' values were quite low, but we illustrate the NRS effect in Figure 5, row 3, panel 1 with data from Ratcliff et al. (1992, Experiment 1), which do not show such a drop. These findings are also difficult for most models to handle, but are predicted quite well by REM, as in Figure 5, row 3, panel 2.

Natural Language Word Frequency

1. Words of higher frequency are recognized less well than words of lower frequency, whether or not the high- and low-frequency words are mixed in the study list (for d' calculations, hit and false alarm rates for words of equal frequency are utilized). Data for mixed lists are illustrated in Figure 6, row 1, column 1 (from Glanzer & Adams, 1990, Experiment 2).

The literature is filled with factors that might contribute to the word frequency effect. The idea that high-frequency words have more common features than low-frequency words is the simplest factor for us to implement within the present version of REM. In addition, high-frequency words almost certainly have more common feature values, given that they occur more frequently in the language by definition. For the purposes of simulation thus far, the feature values for all words have been generated with a g_H value of .45, representing high-frequency words. This value can be compared with the system value of g of .4, used for calculations during retrieval, representing long-run experience with all words. For the purposes of producing word frequency predictions, a set of low-frequency words is generated with a value of $g_L = .325$. This will tend to produce lower (more probable) values for the high-frequency words, so that the matching feature values for high-frequency words will tend to be lower in value and contribute less evidence in favor of an s-image; in short, d' is lower for high-frequency items because more common feature values are less diagnostic. The REM predictions for a list of 40 strong words (20 pairs with 10 storage attempts per word) are given in Figure 6, row 1, column 2.

2. A mirror effect is seen: high-frequency words give lower hit rates and higher false alarm rates, even in mixed lists. The effect is illustrated in Figure 6, row 2, column 1 with mixed list data from Glanzer and Adams (1990, Experiment 2). This is the prototypical mirror effect studied by Glanzer and his colleagues in many settings. In extant models, it can only be explained in ad hoc fashion, usually by assuming that the subject or system assesses the frequency of the test word and then adjusts the criterion (or, in the case of the model of Glanzer & Adams, 1990, the likelihood ratios) so as to produce the observed effects.

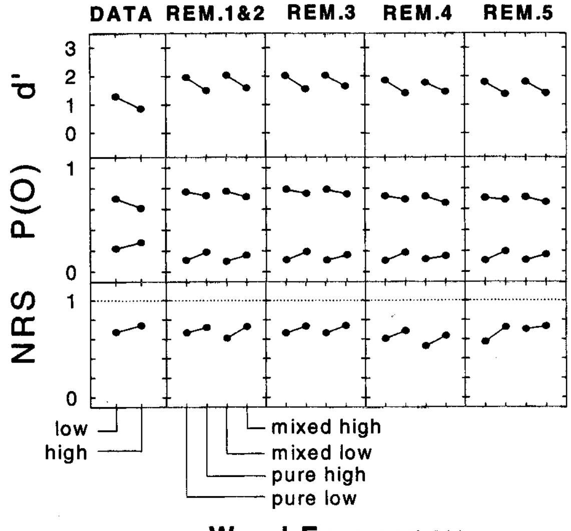


Figure 6. Word frequency data and predictions. Row 1: d'. Row 2: P(H) (upper points) and P(F) (lower points). Row 3: NRS. Column 1, rows 1 and 2: d', and hit and false alarms, respectively, from Glanzer and Adams (1990, Experiment 2, mixed list). Column 1, row 3: NRS data from Ratcliff et al. (1994, Experiments 4 and 5). Columns 2–5: predictions from various REM models (see text). REM parameters for Column 2 are those of Figure 3, except $t = 10$, $n = 40$, $g_H = .45$, $g_L = .325$.

REM predicts a mirror effect using its single default criterion of odds of 1.0, as in Figure 6, row 2, column 2. Several factors are operating together to produce the predictions: (1) High-frequency words have more common features, so the matching values for s-images tend to be smaller and more probable, reducing the likelihood ratio. (2) Factor one is larger than a compensating factor: s-images of high-frequency words have slightly more matching feature values, because errors in storage tend to produce common values, increasing the probability of accidentally matching a high-frequency feature value (since these tend to be common also). (3) The situation with d-images is different. There are only chance matches of feature values in these cases. However, when the test word has more common values, there will be a slight increase in the number of chance matches, increasing the likelihood ratio.[10]

3. The NRS is less than one, perhaps higher for high- than for low-frequency words. Mixed list data from Ratcliff et al. (1994, Experiments 4 and 5), are shown in Figure 6, row 3, panel 1; such data pose problems for a num-

ber of models, as pointed out by the authors. The general form of the data is predicted by REM, as in Figure 6, row 3, panel 2. Ratcliff et al. (1994) also noted that the NRS (i.e., *z*-ROC slope) for high- versus low-frequency distractors was less than 1.0 (.86), indicating higher standard deviations for low-frequency new words. REM predicts an NRS in this casè of 0.82. The higher variability for low-frequency words is related to the fact that less common matching values produce some unusually high likelihood ratios.

Many factors may contribute to the word frequency effect (including the fact that such words have generally been encountered more recently outside the experiment, a factor discussed in connection with REM.5). Glanzer et al. (1993) present a feature sampling account of the effect that is quite different in kind from the present approach. McClelland and Chappell (1994) discussed one factor similar to the present feature frequency approach, and another that could probably be implemented in the REM framework (this model will be reviewed in the Discussion section). Thus the demonstrations of this section should be viewed as some evidence for the plausibility of the feature frequency factor, rather than as evidence against other approaches.

Assessment

With very little structure and very few parameters, REM produces qualitatively correct predictions for standard recognition phenomena, including effects that have posed great difficulties for current models. Because the model has been simplified and limited to only a few key parameters, it is possible to see how the essential structure inherent in the model produces predictions for the key phenomena. Nonetheless, many of the simplifications are difficult to defend, so we turn now to models that relax some of the restrictive assumptions. This will lead us to more reasonable models, and to models that can be applied to a much wider variety of paradigms. Although it is usually possible to find normative Bayesian solutions for these cases, the solutions have such great combinatoric complexity that it is impossible to run the corresponding simulations in real time. Therefore, in each of these cases, simplified mathematical expressions and approaches are adopted that allow the simulations to be carried out. The resultant models could be interpreted either as approximations to a normative model, or as new models in their own right.

REM.2
Associative Images

For predictions of cued recall, free recall, or associative recognition, among other paradigms, it is important to allow one word of a studied pair of words to serve as a cue for retrieval of the other word of the pair. Our first extension of REM will provide a mechanism by which such associative storage and recall could occur, but the extended model will be fit only to the recognition phenomena already discussed.

In order to explore issues of association, consider a paradigm in which *n* word pairs (i.e., 2*n* words) are presented for study (in a nonpaired list also, the subject may of course code and rehearse pairs of items, especially adjacent items). Let the studied vector consist of the concatenation of the separate vectors for the two words presented together (the ordering of the two word vectors will not matter, as we shall see shortly). As in REM.1, each word is represented by 20 feature values, so the concatenated vector consists of 40 featurė values.

Storage now proceeds as before, with each feature having a chance u^* of being stored at each storage attempt. The general theory allows for the possibility of limited capacity, which could be implemented by placing limits on the total number of features available to be stored, or by letting the rate of storage vary inversely with number of features. However, the present simulation simply lets storage proceed independently for each part of the two-word vector. The resultant episodic image that is stored consists of 40 feature positions, many filled with zeroes (nothing was stored), and some feature values stored (possibly incorrectly) for each of the two words.

When one word is presented for recognition testing, its vector of length 20 is matched in parallel to each of the stored pair images, of length 40. The problem of alignment now becomes critical, since the test word might match either of the two vectors concatenated in the stored image. We assume that the test vector is compared in turn with each half of the pair vector. Under this assumption, it is conceptually clear (and easy to show) that the situation with a list of *n* pairs is identical to one in which the list has 2*n* single items. Thus the normative solution for single-word recognition testing after storage of word pairs is simply to treat the *n* pair images as if they consist of 2*n* single images. The odds are just the average of the 2*n* likelihood ratios for the 2*n* part images that make up the *n* pair images. The predictions are therefore identical to those shown in column 2 of Figures 3, 5, and 6, for an equal total number of list words (the number of pairs presented is one half the total number of words). Since the predictions of REM.2 are those of REM.1, they are not repeated.

Given these results, one might wonder why the concept of pair images has been introduced. The answer is of course to allow associations to be represented in the model. One obvious application is cued recall. Although we shall not model recall in this article, it is easy to see how the storing of pair images can enable cued recall to take place: A probe with one member of a studied pair is likely to match its own pair image. If this image is sampled (as in the SAM model of Raaijmakers & Shiffrin, 1980), recovery of the information in the pair image would provide a basis for recall of the associated word. A second application of pair storage occurs when pairs must be recognized, a situation that occurs in the following section.

REM.3
Superimposition of Similar Images

In order to predict the list strength findings, it was assumed that repetitions of a given word are all stored in the same image. To be precise, two spaced presentations were treated just as if the word had been presented once for twice the time (in both cases, 10 storage attempts compared with 7 for a once-presented, weak, item). Our theory ought to provide a mechanism through which such superimposition of spaced repetitions might occur.

We propose the following mechanism: When a word presented for study calls to mind a previous image, and when that image matches the presented word features to a high enough degree, then all storage occurs in the recovered image, and a new image is not stored.

This general idea is implemented as follows: During presentation of a given pair during the study phase, the double-word vector (40 features) is compared with all pair images already stored. We model the typical study in which pairs are repeated in the same order. Therefore, only the presented alignment is used during retrieval to calculate a likelihood ratio for each image.

The likelihood ratios are then summed and averaged, as usual, to produce an odds. If the presented pair is recognized as "old" (odds > 1), an attempt is made to find the previous image. Probably this ought to be done by a recall process, but that would exceed the scope of this article. Therefore, we approximated the situation as follows: when a positive recognition decision is made for a pair, new information is stored in the previous pair image with the largest likelihood ratio. In this case (i.e., when the new storage is added to an earlier image), the current storage attempts are reduced in number (to 3, compared with 7 that occur during presentation one, consistent with the view that less storage effort is devoted to words recognized to be repetitions). When the current word pair is not recognized as a repetition, it is treated as a new presentation, and 7 new storage attempts take place. It should be noted that this procedure will occasionally result in words being stored in earlier images of different words.

The simulation was run using the parameters of REM.1 (or REM.2). Some statistics concerning the numbers of times that a twice presented word is stored in various types of images are as follows, for the case of high-frequency words: The probability that a word pair is stored with its own earlier image ranges from .92 for $n = 8$ to .79 for $n = 80$, where n is list length. The probability that a first-presented pair is stored with an image of some earlier presented pair ranges from .03 for $n = 8$ to .08 for $n = 80$. The model predictions are given in the third column of Figures 3, 5, and 6. On the whole, the qualitative accord with the data is again good.[11]

The practical and pragmatic complications of using rules for superimposition of traces are considerable, in terms of the number of types of images that end up being stored. These complications impede clear exposition, expand the need for data analysis, and slow the simulations (especially for REM.5, when extralist words are introduced into the simulation). For these reasons, and because the superimposition rules do not much alter the qualitative predictions, the subsequent simulations revert to the simpler assumptions of REM.1 and REM.2: all repetitions within the list are superimposed in a single image.

REM.4
Context Features and Activation Threshold

A "real" episodic memory would contain untold numbers of episodic images (see REM.5 below for an implementation). It seems unlikely that the system would activate all of these and take them all into account in the calculations. In addition, if extralist images were incorporated in the simulation, these would vary in similarity (feature overlap) on both content and context features, making normative calculations difficult if not impossible, and likely distorting any attempt at simplification. It seems evident that matters would be simplified greatly if a way were found to restrict calculations largely to the images from the list. Two steps are involved in doing so: the introduction of context features that vary as time passes, thereby allowing discrimination of recent items from older ones, and the introduction of a threshold for activation. Without the introduction of extralist images, of course, neither of these additions to REM is required, since there are no other images from which the list images need segregation. However, it clarifies the exposition to add these steps first, and then add extralist images.

Activation Threshold

For an image to be activated, its likelihood ratio must exceed τ_1.

Context Features

In order to produce sufficient differentiation of list images from extralist images, one has to combine the proper degree of context change, context feature diagnosticity (i.e., the value of g for context features), and the number of context features. We found it convenient (in terms of time to run the simulation) to append 40 context features to the 40 word features of the previous models, and to set the g for context features, g_c, equal to .2 (used at both storage and retrieval; we could have used .4, as for word features, but would then have needed a greater number of context features). The context features are assumed to have a fixed set of values for all the presented word pairs, and assumed to have the same values for each test word. Under this assumption, both targets and distractors share a set of features that have a high probability of matching.

If only list images are activated, and the system "knows" which are the context features and that these are common to all the activated images, the normative solution is simple: all the context features should be ignored entirely, and only the word features should be used to calculate likelihood ratios. The obvious drawback of such a model is the fact that the use of word features alone does not have the potential to segregate list images from extralist images (which will be needed in REM.5). Thus we focus on models in which the context features do play a role in the calculations.

By far the simplest model to implement involves partitioning the retrieval process into two pieces, with activation of images based on context features only, and recognition decisions based on word features only. (Some alternative approaches are discussed in Appendix B).

The basic idea runs as follows. The subject partitions the probe set into two sets of features: *set designating* and *word designating*. The set-designating features are used to activate images in a given *region* of memory; as a result, all the activated images tend to contain these features in common. The set-designating features could contain both context features and the word features that are in common to all words on the recent list; in the present simulation, it is convenient to let them consist of context features only, since the word features are all assumed to be independent. In most laboratory studies, the set-designating features tend to remain the same for all test words during a given test period. The word-designating features are used to assess whether the corresponding word is in the activated set of images.

Specifically, a memory probe is first made with only context features. Equation 4 is used and the threshold τ_1 applied. The set of images passing threshold is the activated set. Next the word features are used in Equation 4, applied only to this activated set. Then Equation 2 is used to produce an odds and make a recognition decision. In order to activate most, but not all, list images (and, looking ahead, only a few extralist images), a relatively high value of τ_1 was adopted: e^8 (= 2,980.9). For this value, regardless of frequency, 92% of the images of strong words from the list and 81% of the images of weak words from the list become activated. As we shall see, assumptions about context change can be made so that with this parameter only a few extralist images are activated.[12] This model does an excellent job. The parameter values other than τ_1 and g_c were those of the previous variants. The predictions are given in column 4 in Figures 3, 5, and 6.

How to think of the "two-phase" approach of REM.4 is an open question. One could take the view that there are not actually two phases, and that this method is simply an approximation to an optimal Bayesian solution (see Appendix B). Alternatively, one could argue that this is a valid model in its own right, but without an implication that the two phases occur in temporal order (with each taking measurable time). Finally, one could take the view that this is a model with two successive phases, each taking measurable time. We will not try to discuss this issue in this article.

REM.5
Approximating a "Real" Episodic Memory With Extralist Images

Although the images considered thus far have been restricted to the most recently presented list, memory must contain untold numbers of other episodic images. When selective access to images of words on the recent list is required, presumably the availability and use of context cues provide the basis for selection. Note that there are excellent reasons to want at least some prior list images to join the activated set: in recognition tasks, false alarms are enhanced for distractors from lists just prior to the current list; in recall tasks, intrusions are often observed from lists prior to the current list, in accord with the recency of those lists. For example, in Nobel (1996), about 39% of incorrect responses in cued recall were words from other pairs in the current list, but 28% of intrusions were words from the immediately preceding list.

We approximated a real set of episodic memories by storing a sequence of 20,000 images, with context features whose values gradually drifted over time. Then additional drift was allowed to occur, and the context was fixed during presentation and test of the current list. Storage strength for the extralist words was varied randomly, over a relevant range (images with too few features stored would never reach the likelihood threshold of e^8, and extralist images with too many features stored would tend not to reach the threshold because of the drift in context values). Word frequency was incorporated in binary fashion: 10,000 words were presented (i.e., stored) once each—these were defined to be low-frequency words; mixed randomly with these were 100 words presented (i.e., stored) 100 times each—these were defined to be high-frequency words. To reduce the complexity of the simulation, all 20,000 images were stored as separate images, and no features were added to earlier presented images (superimposition, and storage in earlier images, would probably not occur often enough to be a significant factor, given that the high-frequency items are represented at long intervals with differing contexts). Half the words to be presented on the list were chosen randomly from the 10,000 low-frequency words, and half were chosen randomly from the 100 high-frequency words. Half the distractors at test were chosen randomly from the remaining low-frequency words, and half were chosen randomly from the remaining high-frequency words.

Context drift. The 40 context features were assigned values with Equation 1, with g = .2, and these were assigned to the first (oldest) word stored. Before each subsequent word, each context feature was assumed to have a small probability (.008) of fluctuation; if fluctuation occurred, the current value was replaced by a value chosen at random from the generating distribution (Equation 1, with g = .2). To model the difference between extralist context and list context, 140 extra time steps of fluctuation were allowed to occur before the context values were fixed for use in storage and test of the current list.[13]

Strength of storage. Each image stored was allowed a randomly assigned number of storage attempts, chosen from a uniform distribution from 5 to 15.

The other parameter values used for this case were the same as those of Model REM.4. The predictions are about equivalent to those for the previous models; they are shown in the last column of Figures 3, 5, and 6.

The predictions tend to be similar to those for earlier variants because, on the average, only 4.98 extralist images are activated (about one half of which are high frequency). Almost all activated extralist images are due to recently presented words (in the most recent 100 extralist presentations). Almost none (.023) of the extralist activated images are s-images (extralist images of the test word). The few that are activated are essentially all high frequency (since these are the only s-images likely to have occurred in the most recent 100 extralist positions); this fact contributes slightly to the word frequency effect, but not to a large enough degree to alter substantially the patterns of predictions (see Figure 6).

DISCUSSION

It would have been possible to introduce REM in the form of REM.5. If this had been done, it would have been difficult to determine which assumptions were responsible for the various predictions. By adding assumptions successively to the basic model, one can see that none of them are critical: the basic pattern of predictions is present in the simplest form of the model, and the successive additions to the model do little to alter the predicted patterns. It is interesting that very little in the way of parameter adjustment was needed as the successive model variants were introduced (not that there are many parameters with which to tinker). There is of course no reason why parameter values should be the same for different models, but the fact that little adjustment was required suggests that the basic form of the predictions is quite robust. The reason for this robustness may lie in the fact that the variants are all reasonable approximations to an optimal Bayesian solution (though we cannot verify this speculation at the moment).

It is particularly interesting that the default criterion of odds of 1.0 proves adequate to produce mirror effects throughout the series of models. REM.1 was derived in such a way that odds of 1.0 ought to have been the default criterion (for reasons that are discussed in the theory of signal detection; see, e.g., Green & Swets, 1974). However, the variants are implemented not on the basis of normative solutions, but as approximations based on simplicity and ease of simulation. To predict accurately, such approximations could have required a default criterion setting at some value other than 1.0; this would not have been a problem, as long as the value was fixed across the set of predictions for that model. This has not yet proved necessary, partly because of the two-phase approach used in REM.4 and REM.5.

Of course, whatever the default setting for the criterion, all the REM variants incorporate the assumption that the subject can move the criterion to other values, if there is reason to do so. Since list length and strength (for example) are readily apparent to the subject, it would be no surprise if the subject would change the criterion setting when these factors were varied. However, these factors are probably specific to experimental studies, and there is little opportunity in life to learn the correct mapping from these factors to a criterion setting that would produce mirror effects. Thus a theory like the present one in which the criterion need not move confers certain advantages.

The criterion excepted, the parameters of the model have thus far been treated as system parameters not under the control of the subject. In addition, with the exception that word frequency was assumed to be correlated with feature value base rates (high or low g), the system parameters were not changed across tasks. The general idea is that the system parameters are learned over developmental time and change only slowly. Whether the subject should be allowed some control over their value is a matter to be explored in the future.

The purpose of this article has not been to fit a particular set of data in quantitative fashion, but to demonstrate the fundamental properties of a new model. The fact that this model (actually a variety of model variants) is capable of reproducing the basic qualitative trends in the recognition memory literature with almost no parameter adjustment and almost no parameters provides a certain validity to the approach. The fact that a number of these predictions have been difficult or impossible to obtain within other models testifies to the promise of the REM model. A necessary next step, of course, will be the quantitative application of the model to particular data sets.

We justified the REM approach by arguing that the system retrieves in "optimal" fashion, at least for the initial variant of the model. In reality, of course, even for the simplest variant, retrieval is only optimal in light of the restrictive assumptions concerning what information is available to the system, and it is at best only approximately optimal for the later variants. Thus we would not want to argue seriously that our retrieval system is "optimal." Nonetheless, the use of an "optimality" approach provided a principled reason for the functional form of our model, a form giving a simple and rather elegant description of a data set that has previously provided headaches for theorists.

The relation of the present model to other models of explicit recognition could be the subject of an entire article, but it deserves at least a few comments here. The REM model shares with many models the assumption that some single value, calculated in parallel over the stored information for the list words, provides the basis for a recognition decision. These models include those assuming separate storage for different episodes, such as SAM (Gillund & Shiffrin, 1984; Shiffrin et al., 1990), MINERVA (Hintzman, 1988), Chappell and Humphreys's (1994), and McClelland and Chappell's (1994), and those assuming composite storage, such as the MATRIX model

(Humphreys et al., 1989), TODAM (Murdock, 1982), and CHARM (Eich, 1982, 1985). A variety of autoassociative neural net models fall into this category as well (e.g., that of Chappell & Humphreys, 1994). The present model seems to be unique among these in having a default placement of the decision variable (centered about odds of 1.0), even though the criterion can be moved at will.

A few previous recognition models have incorporated assumptions allowing the prediction of the list strength effect—the SAM model, that of McClelland and Chappell (1994), and that of Chappell and Humphreys (1994). (Hintzman's 1988 MINERVA model does not predict this result, but it could probably be altered to do so, as discussed in Shiffrin et al., 1990.) The SAM model assumed differentiation—stronger images are activated less strongly by a cue representing a different word (offsetting an increase in the strength of the context cue to that image). The REM model incorporates a form of differentiation, based on the word features, but does not include an offsetting effect based on context features (since the context features are used for activation, but not for calculating the odds within the set of activated images).

The model of McClelland and Chappell (1994) (abbreviated M–C in the following) is in many respects quite similar to the REM model, particularly in its mathematical form. As in SAM and REM, words are stored in separate images, and repetitions of a word are stored in the same image. As in REM, likelihood ratios are calculated for these images. However, in M–C, each image calculates its own odds that it is the test word (whereas in REM the odds calculation is a global one, that the test item is one of the list words). The separate odds in M–C are combined by a simple rule: if any odds value is greater than some criterion value, respond "old." The difference between REM and M–C, then, is largely conceptual at this level of analysis: in M–C, each image calculation must take into account knowledge of the experimental situation, such as the estimated number of words on the list.

In REM, each image calculation is based not on the experimental situation but on factors that the system could have been expected to have learned over a lifetime of experience. The rule for combining these (calculating an average) requires only a count of the total number of activated images, an internally available quantity, and there is no explicit need to build in knowledge of the experimental situation in these basic calculations. The subject's experimental knowledge can then be superimposed on the result of the calculations.

It must be admitted, however, that the mathematical form of the two approaches is quite similar, with differences largely in the details. Note that M–C uses a plausible but arbitrary combination rule and as a result ends up with a decision scale that is not centered on odds of 1.0, as in REM, but this does not provide a good reason to prefer one model over the other.

M–C and REM basically account for list length, strength, and list strength effects in similar ways. M–C has features with binary values (0 and 1) but can also use the feature frequency approach to the word frequency effect, and this is indeed one of the approaches they suggest. The other approach that they suggest involves variation of a parameter not present in REM—namely, the proportion of features sampled at test. This approach appears to work for pure lists, but we have some question whether it will be acceptable in mixed list situations.

The model of Chappell and Humphreys (1994) is relatively straightforward in initial conception, although it is implemented with a welter of special processes, assumptions, and parameters. There is a lexicon implemented as a special kind of autoassociator, which when probed by inputs does not always converge on a stored word. The lexicon contains, in effect, separate word traces. The words in the lexicon are associated to two kinds of peripheral patterns: word features and context features. Study causes strengthening of the lexical representation and also strengthening, in principle, of the peripheral associations to the lexical traces. However, in practice, the word features are not strengthened in association, and the context features are given just one chance at strengthening, regardless of study time or repetition. Recognition occurs if the joint use of context features plus word features causes convergence on some word in the lexicon. Strength effects occur because of strengthening of the lexical representations. Length effects occur because additional context-to-lexicon associations reduce the probability of convergence on any single word. The list strength effect is built in by assumption, since the context association to the lexicon does not vary with repetition or study time. The other interesting feature of this model is the intersection property: since context selects words from the lexicon that have been studied in that context, the use of context features in the probe selects all the list words, and the word features then try to select the lexical entry from these. This aspect of the model is similar to REM.4. The complexities of the Chappell and Humphreys model prevent any additional discussion in this article.

TOWARD A GENERAL MODEL OF GENERIC AND IMPLICIT MEMORY

Free and cued recall tasks, and generic and implicit memory tasks, all require relatively complete and accurate knowledge of words, in addition to the incomplete and error prone episodic vectors that are used to carry out episodic recognition. We propose that such general knowledge is stored in vectors similar to the episodic ones (albeit more complete and more accurate). Although they are not different in kind, we term such word vectors *lexical/semantic*, and describe a simple means by which such lexical/semantic vectors can grow from the storage of multiple episodic storage events:

1. Features are stored in separate images due to the operation of attention operations in short-term memory that help parse experience into episodes.

2. Features are stored in already existing images that are accessed when an item is presented, if the accessed images are sufficiently similar to the presented item. There are two important cases to consider:

2.1. When a word is presented above threshold it will access its own lexical/semantic image with a probability approaching 1.0. Storage of current episodic information (both context and content features) then occurs in this lexical/semantic image. Such storage plays a critical role because it provides the main mechanism for implicit memory effects. For example, if some current context features are stored in a word's lexical/semantic vector, then, when that same word is presented later in a similar context, the probe features will match their corresponding lexical/semantic image better.

2.2. On occasion, the current presentation will access the episodic image of an earlier presentation of the same word. Given sufficient similarity to the present probe, current event features will be added to this previous episodic image: this is the critical mechanism by which episodic images gradually grow to become lexical/ semantic images over developmental time.

3. Regardless of any storage that might occur in previous images that are accessed, features are normally stored in a new episodic image as well. There is, however, one important exception: when a previous episodic image is accessed that is not assessed to be substantively different from the present storage episode, the current information is simply added to the previous image, and a new episodic image is not formed. This assumption is essential to predict the list strength effect, as described in REM.3.

To summarize, when a word is presented, storage of at least some of the current event information usually occurs in several kinds of images: in the lexical/semantic image of the presented word, in a previous episodic image of the presented word (given sufficient similarity), and in a new episodic image (unless a previous episodic image has very high similarity).

With these assumptions, it is possible to see how a set of lexical/semantic vectors could develop with experience. These vectors will of course contain both content and context features. The context features are extracted from many events with different feature values, and hence the context part of the lexical/semantic vector is made up of a kind of composite context that matches no one identifiable situation. Thus these images can appear to the subject to be "decontextualized." The other issue concerns the way in which errors in storage can be corrected, as a lexical/semantic image develops. We had assumed in REM, at least as a first approximation, that a feature once stored retains its value subsequently. (This assumption helps explain the phenomenon of "registration without learning" reported by Hintzman, Curran, & Oppy, 1992, though this effect will not be modeled in this article.) In order to allow error correction, however, this assumption is partially relaxed. We assume that feature values tend to be retained once stored, but that storage errors can be corrected through the operation of explicit attention given the feature in question.

Retrieval from lexical/semantic traces can operate in much the same way as for episodic traces: based on the probe features, a likelihood ratio can be calculated for each image. These likelihood ratios can then subserve retrieval in ways that are described in the following sections. A rather delicate issue concerns the management of retrieval so that lexical/semantic traces can be accessed at one moment, and episodic traces at another, so that the image that best matches the probe does not interfere unduly with access to other images a moment later. These matters are also addressed briefly in the following.

EXTENSIONS

We first give a few pointers concerning the way in which REM might be extended to other explicit, episodic, memory paradigms. For extensions to multiple-item recognition tests and to cued recall, it is useful to direct discussion toward a paradigm in which the subject studies a list of pairs of words, AB, CD, EF, and so forth. Following such study, the subject can be tested for (1) cued recall, A–? (2) single-item recognition, A versus X; (3) a variety of double-item recognition tests: AB versus XY (pair recognition), AB versus AX (one–new), AX versus XY (one–old), and AB versus CF (associative recognition).

Multiple-Item Recognition Tests

In testing with multiple words, questions concerning capacity limits become critical. To facilitate discussion, assume that the set of studied context features is $\{C_s\}$, of size N_{Cs}, and that the sets of studied word one and word two features are $\{W_{1s}\}$ and $\{W_{2s}\}$, of sizes N_{W1s} and N_{W2s}. Assume that the sets of features and their numbers used in the probe at retrieval are corresponding: $\{C_r\}$, N_{cr}, $\{W_{1r}\}$, $\{W_{2r}\}$, N_{W1r}, N_{W2r}. Up to now, the sets and their sizes have been assumed to be the same at storage and retrieval. Further, the sets and their sizes have not varied depending on the composition of the probe set at retrieval (e.g., whether context is joined by one or two words). There are, however, both conceptual and empirical reasons to believe that there ought to be capacity limitations as the number of test cues increases. Whether the limits begin to have effect as low as two words (three cues, including context) is an open question. One way to instantiate a capacity limitation at retrieval involves limiting the total number of features that can constitute a probe. Suppose that for single-word testing the probe consists of $N_{cr} + N_{W1r}$ features. For a double-word probe, suppose that the number of features in the probe, N_p, is in the range from $N_{cr} + N_{W1r}$ at a minimum to $N_{cr} + N_{W1r} + N_{W2r}$ at a maximum. Whatever the limit, N_p, the subject would strategically allocate this number among the three types of features, possibly selecting which features of each type are in the probe (as well as how many).

We have looked briefly at one model of this sort: for simplicity, context features were ignored. When two words are both used in a memory probe, and the order of the two words is not necessarily the studied order, assume that the two words are aligned with a given memory image in both orders; the two resulting likelihood ratios are then summed and divided by two, the result representing the likelihood ratio for the image in question. However, assume that the proportion of the total features from the two words that can be used in the probe is gamma, where gamma is between .5 and 1.0 (the unused features in the joint probe having their values set to zero). To this point in the article, the only time in which two words are used together in a probe occurs in REM.3, during study, when each pair is assessed for recognition. In REM.3, it was implicitly assumed that gamma was 1.0 (since all the features of both words were used in the probe).

The simplest assumption for multiple-item recognition testing involves cuing jointly with the available features of both words. Some preliminary simulations proved enlightening: when gamma is 1.0 (unlimited cuing capacity), performance (d') is much too high for paired testing relative to single-word testing. When gamma is .5 (strongly shared cuing capacity), performance (d') is too low for paired, one–new, one–old, and associative recognition relative to single-word testing. Testing of just a few intermediate values of gamma did not reveal a value that would fit the various cases successfully.

As an alternative model when two words are presented for recognition, suppose that the subject probes memory separately with the two words and then combines the results (for example, in one–new testing, a "new" response could be given if at least one of the words was judged to be new). Our initial simulations with this approach showed that performance was qualitatively in line with data. For associative recognition, of course, such a strategic approach cannot work, because single probes are all "old" and equally familiar. We suggest that in this case the subject adopts a different strategy, in which the test words are used one at a time as cues for cued recall.[14]

Cued Recall

Within the general REM framework, a model for cued recall could be developed that borrowed important components from the SAM model (e.g., Raaijmakers & Shiffrin, 1980). However, within REM, the interaction between retrieval from lexical/semantic and episodic images can be laid out explicitly. Before proceeding, we propose an additional mechanism to reflect the idea of "associative encoding" such as the use of mnemonics, special linking imagery, and the like; in other words, we wish to allow two (or more) words stored within one image to be associated to a degree beyond that produced by their mere co-occurrence. In particular, assume that some of the features stored when one is studying a pair of words are marked as "associative." Associative marking results from coding strategies and builds up as time spent in coding increases. The marked features are used to help the recovery of target features when the correct image has been sampled, as described below.

The first step in cued recall, as in recognition, involves the rapid and sure access to the lexical/semantic image of the test word. Let us assume that this image receives temporary inhibition (for a few hundred milliseconds), so that it will not dominate immediately subsequent retrieval from other lexical/semantic or episodic images. The subsequent stages of retrieval proceed using as a probe the retrieved content features of the test word, the sensory features representing the input of the test word, and current context features: (1) These probe features are used as in recognition to generate a set of likelihood ratios, one for each image (associative marking plays no role yet). The activated images tend to include episodic images and exclude the lexical/semantic images, because, except for the image matching the test word (which has been inhibited), the lexical/semantic images of other words mismatch on most word features and many context features. (2) The likelihood ratios are compressed[15] (raised to a fractional power, η) and sampled according to a Luce choice rule (as in SAM; the probability of choosing image k is $\lambda_k^\eta/\sum\lambda_j^\eta$). (3) Features of the sampled image are recovered. We assume that all the features corresponding to the test word are recovered, as are of all the associatively marked features of the target part of the image. In addition, a proportion of the nonmarked features in the target part of the image are recovered (the proportion based on the likelihood ratio for this image). (4) A new likelihood ratio is calculated according to the match to the test word features plus context only. If the ratio is above a threshold, the sampled image is accepted as the "correct" one; if not, either another sample is taken, or recall attempts cease (when the number of unsuccessful samples exceeds a criterial value). (5) The currently activated image is inhibited temporarily. (6) Once an image is accepted, the remaining recovered features (plus context) are used in a probe of lexical/semantic memory, in an attempt to produce an overt recall. (This phase of return to the lexical/semantic images might require some inhibition of the weight given to the current context features.) The likelihood ratios are calculated as usual. The ratios are compressed and sampled. When a reasonable number of word features are in the probe, there will again be a strong tendency to sample the lexical/semantic representation of that word, in preference to other lexical images, or even other episodic images (since the matching episodic image has been inhibited). The image sampled is output as a recall if its likelihood ratio is above a threshold value. (7) The system returns to Step 6, and sampling with the same set of cues continues. (8) Sampling ceases, and recall stops, if a threshold number of unsuccessful samples occurs. (It is conceivable that the threshold for cessation depends on the number of word features that have been recovered.)

This cued recall model is meant to illustrate one plausible way in which retrieval from episodic images and retrieval from lexical/semantic images could work hand in hand to allow recall to take place. At various times during this process that lasts up to several seconds in duration, it is necessary for the retrieval system to focus primarily on episodic or primarily on lexical/semantic images, to "find" images other than the first and strongest one accessed, and to operate without undue distortion caused by the presence of one or more images that match the probe cues very well. A partial account of the way in which such retrieval could be managed might involve temporary inhibition of the image just sampled (which could operate automatically). Other factors could operate as well, such as rapid attention shifts between context and content cues, or the possibility that access to different types of images could occur in parallel. These possibilities would have to be explored through appropriate simulations, and must be left to future research.

Lexical/Semantic Access, and Implicit Memory

We take the view that the same general mechanisms should apply in episodic, generic, semantic, procedural, and implicit memory tasks, even though many of these require that retrieval be focused on general knowledge rather than recent events. However, in tasks primarily requiring access to lexical/semantic images, it seems likely that the reliance on context features in the probe will be reduced. Note that it is probably impossible to remove context cues entirely, since they are continuously present in the internal and external environment, and their presence may reflect unintentional "leakage" rather than any intentional strategy. Another way to focus retrieval on lexical/semantic images might involve manipulation of thresholds. If there is a threshold for activation based on total number of relevant features, it is conceivable that the subject can increase this threshold to the point at which most episodic images would not be activated. In either event, lexical/semantic retrieval can then proceed according to the rules already described. Judgments such as lexicality and general word familiarity might be based on summed likelihood ratios (as in episodic recognition), and recall could be carried out as it is for episodic free and cued recall tasks.

Lexical decision tasks are an interesting case. For normal lexical decision tasks in which the test items are presented above threshold, the activation and subsequent sampling of the relevant image, if there is one, will occur quickly and with high probability, so a process akin to cued recall, based especially on the recovery phase following sampling, could be posited to govern the decision. On the other hand, a process analogous to episodic recognition would also make a plausible model: the lexical decision could be based on the sum of the likelihood ratios for activated images (such a model might be necessary for lexical decisions when the presented word is at threshold). These alternative models require exploration in future research.

In implicit memory tasks, we argue that access to lexical/semantic images is the basis for decisions, but that such images have been modified by recent episodic presentations. For example, suppose the task is threshold identification. The probe cues would be visual form features plus context. Repetition priming effects (i.e., the gain caused by presentation of the flashed word in a recent list) would be due to the alterations in the lexical/semantic image of the flashed word that took place during its list presentation. The context and new content features that were added to the lexical/semantic image during the list presentation would increase the match to the probe cue at the subsequent threshold test.

These speculations and comments concerning extrapolations of the present model into domains other than that of explicit, episodic, recognition are of course the proper domain of future articles and research, but they help provide a more general context in which to view the present theoretical research effort. Focusing only on the specific recognition model, it seems safe to say that it provides a quite simple and robust account of the basic phenomena of recognition memory, including phenomena that have caused difficulties for other recent models.

REFERENCES

ANDERSON, J. R. (1990). *The adaptive character of thought.* Hillsdale, NJ: Erlbaum.

BOWLES, N. L. & GLANZER, M. (1983). An analysis of interference in recognition memory. *Memory & Cognition*, **11**, 307-315.

CHAPPELL, M., & HUMPHREYS, M. S. (1994). An auto-associative neural network for sparse representations: Analysis and application to models of recognition and cued recall. *Psychological Review*, **101**, 103-128.

EICH, J. M. (1982). A composite holographic associative recall model. *Psychological Review*, **89**, 627-661.

EICH, J. M. (1985). Levels of processing, encoding specificity, elaboration, and CHARM. *Psychological Review*, **92**, 1-38.

GILLUND, G., & SHIFFRIN, R. M. (1984). A retrieval model for both recognition and recall. *Psychological Review*, **91**, 1-67.

GLANZER, M., & ADAMS, J. K. (1990). The mirror effect in recognition memory: Data and theory. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, **16**, 5-16.

GLANZER, M., ADAMS, J. K., IVERSON, G. J., & KIM, K. (1993). The regularities of recognition memory. *Psychological Review*, **100**, 546-567.

GLANZER, M., & KIM, K. (1997). *Slope of receiver operating characteristics in recognition memory.* Manuscript submitted for publication.

GREEN, D. M., & SWETS, J. A. (1974). *Signal detection theory and psychophysics.* New York: Krieger.

GRONLUND, S. D., & ELAM, L. E. (1994). List-length effect: Recognition accuracy and variance of underlying distributions. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, **16**, 1-15.

HINTZMAN, D. L. (1988). Judgments of frequency and recognition memory in a multiple-trace memory model. *Psychological Review*, **95**, 528-551.

HINTZMAN, D. L., CAULTON, D. A., & CURRAN, T. (1994). Retrieval constraints and the mirror effect. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, **20**, 275-289.

HINTZMAN, D. L., CURRAN, T., & OPPY, B. (1992). Effects of similarity and repetition on memory: Registration without learning? *Journal of Experimental Psychology: Learning, Memory, & Cognition*, **18**, 667-680.

HUMPHREYS, M. S., BAIN, J. D., & PIKE, R. (1989). Different ways to cue a coherent memory system: A theory for episodic, semantic, and procedural tasks. *Psychological Review*, **96**, 208-233.

HUMPHREYS, M. S., PIKE, R., BAIN, J. D., & TEHAN, G. (1989). Global matching: A comparison of the SAM, Minerva II, Matrix, and TODAM models. *Journal of Mathematical Psychology*, **33**, 36-67.

LOCKHART, R. S., & MURDOCK, B. B. (1970). Memory and the theory of signal detection. *Psychological Bulletin*, **74**, 100-109.

MCCLELLAND, J. L., & CHAPPELL, M. (1994, November). *Bayesian recognition.* Poster presented at the 35th Annual Meeting of the Psychonomic Society.

MURDOCK, B. B. (1982). A theory for the storage and retrieval of item and associative information. *Psychological Review*, **89**, 609-626.

MURNANE, K., & SHIFFRIN, R. M. (1991a). Interference and the representation of events in memory. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, **17**, 855-874.

Murnane, K., & Shiffrin, R. M. (1991b). Word repetitions in sentence recognition. *Memory & Cognition*, **19**, 119-130.

Nobel, P. A. (1996). *Response times in recognition and recall.* Unpublished doctoral dissertation, Indiana University.

Raaijmakers, J. G. W., & Shiffrin, R. M. (1980). SAM: A theory of probabilistic search of associative memory. In G. H. Bower (Ed.), *The psychology of learning and motivation* (Vol. 14, pp. 207-262). New York: Academic Press.

Raaijmakers, J. G. W., & Shiffrin, R. M. (1981). Search of associative memory. *Psychological Review*, **88**, 93-134.

Ratcliff, R., Clark, S. E., & Shiffrin, R. M. (1990). List-strength effect: I. Data and discussion. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, **16**, 163-178.

Ratcliff, R., McKoon, G., & Tindall, M. (1994). Empirical generality of data from recognition memory receiver-operating characteristic functions and implications for the global memory models. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, **20**, 763-785.

Ratcliff, R., Sheu, C.-F., & Gronlund, S. D. (1992). Testing global memory models using ROC curves. *Psychological Review*, **99**, 518-535.

Shiffrin, R. M., Ratcliff, R., & Clark, S. E. (1990). List-strength effect: II. Theoretical mechanisms. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, **16**, 179-195.

NOTES

1. Dr. Raaijmakers is Professor of Psychology at the University of Amsterdam; Dr. Schooler has been a NATO postdoctoral fellow at the University of Amsterdam, and a postdoctoral trainee on an NIMH Training Grant in Modeling of Cognitive Processes at Indiana University. Other researchers on the general project include Dave Huber at Indiana, and Chris Schrijnemakers, Rene Zeelenberg, and Diane Pecher at the University of Amsterdam.

2. The precise form of this distribution is probably not critical; we looked less thoroughly at a two-point distribution, with values of 0 and 1, and it seemed to produce predictions similar to those reported in the paper for most of the phenomena under discussion.

3. The assumption that no error correction occurs is made for simplicity and expediency, but is not likely to be strictly true. Relaxations of this assumption are discussed later in the article.

4. For convenience, it has been assumed that the error in the system occurs during storage. It would not change the model in any way to assume instead that the error occurs during retrieval, or occurs during both storage and retrieval. Reference to Equation 3 makes it clear that the critical probability is defined in terms of the probability of matching feature values between test probe and image.

5. In this model, information concerning presentation frequency would have to be encoded in the feature values of the single image. The possibility that a repeated word might be represented by several images is incorporated in REM.3.

6. The degree to which recall of a particular image is used to supplement a decision based on familiarity is a topic of considerable recent research. Although we adopt the one-process approach here, we do not wish to rule out the two-process approach. Even if two processes are used, the one-process model might produce accurate predictions if those trials on which image recall produces a positive response are trials on which a response based on familiarity would have also produced a positive response (most of the time). Later in this article we discuss recall, and it will be seen that a correlation of this kind is most likely.

7. The reliability of empirical slope estimates is unclear. We have found it necessary to collect an amount of pseudodata several times the amount usually collected empirically in order to obtain stable theoretical predictions.

8. Since the distractor odds are an average of n samples from the d-image likelihood ratio distribution, and the target odds are the same except that one of the d-image samples is replaced by an s-image sample, it is easy to show that, in theory, Var(Distractor Odds)/Var(Target Odds) = $n/[n - 1 + \mathrm{Var}(\lambda_T)/\mathrm{Var}(\lambda_D)]$. The square root of this expression is the theoretical ratio of standard deviations (assuming that the variances are finite), but this ratio has little relation to the NRS calculated from multiple criteria. Note that any monotonic transformation of the odds and criteria, including the log transform, produces identical d' values, hit and false alarm rates, ROC curves, and NRSs; The approximate linearity of the normal ROCs (despite the non-normal odds distributions) may be due in part to the existence of a transform of the raw odds which approximately converts the major portion of the center of both distributions to normal distributions (as seen in Figure 4, a log transform does not quite do the job). For a related discussion of these issues, see Lockhart and Murdock (1970).

9. The SAM model incorporated a tradeoff of item and context strength, so that it did not necessarily predict a change in the list strength effect as the strength difference between strong and weak words increased. REM (and probably the other models relying on differentiation mechanisms) predicts an increasing gain due to stronger other words, as the strength difference between strong and weak words increases. This prediction has not been adequately tested.

10. This model predicts changes in the mirror effect for word frequency with list length. Bowles and Glanzer (1983) observed an effect of this sort, but it was nonsignificant. In any event, the lengths that they used did not provide a strong test of this prediction.

11. One might wonder what would happen if the study list contained word pairs but repetitions of individual words occurred as part of rearranged pairs, so that no pair was repeated. Murnane and Shiffrin (1991b) studied this issue and discovered that this form of repetition does produce a list strength effect (repeated items harm recognition of other items). This finding is consistent with the mechanism suggested in this article, since an entire pair (say, AB) would not as often match a rearranged pair (say, AC) well enough to trigger the common storage mechanism.

12. In some model variants, such as those discussed in Appendix B, it is necessary to have a second threshold that must also be passed: The number of nonzero features in the image that have corresponding nonzero features in the memory probe must exceed τ_2 (some small number). In REM.4, τ_1 is set to such a high value that weak images would never exceed threshold, so τ_2, even were it incorporated in the present model, would not come into play.

13. There are of course many situations in which it would be useful to reinstate some older context in order to access memory. How older context might be reinstated is an interesting question with several possible answers, but it cannot be taken up in this article.

14. There is a good deal of data suggesting that a recall strategy is indeed used in associative recognition. To take just one example from our laboratory, Peter Nobel (1996) has shown that the distributions of response times in associative recognition are much slower and more skewed than in other recognition tests, and that they are similar to those seen in cued recall tests.

15. It is easy to show that the probability of an image containing the test word, given that some image on the list does, is just $\lambda_k/\sum\lambda_j$. If so, it would be optimal to sample first the image with the highest λ. Simulations revealed that cued recall would be too efficient with this assumption. In addition, some preliminary simulations suggested that probabilistic sampling with $\eta = 1.0$ also led to too efficient sampling, but this issue has to be explored more carefully.

APPENDIX A

In this Appendix we present the basic derivations for REM.1, and discussion of related models.

REM.1

The odds for the test word being old (O) over new (N) equals the likelihood ratio of observing the data D (representing the set of D_j for all images, where D_j is the set of matching and mismatching values for image j) for an old or new test times the prior odds for an old or new test (prior odds indicated by a subscript of o).

$$\frac{P(\mathrm{O}\,|\,\mathrm{D})}{P(\mathrm{N}\,|\,\mathrm{D})}=\frac{P(\mathrm{D}\,|\,\mathrm{O})}{P(\mathrm{D}\,|\,\mathrm{N})}\frac{P_{\mathrm{o}}(\mathrm{O})}{P_{\mathrm{o}}(\mathrm{N})}. \tag{A1}$$

The prior odds will usually be the odds of an old item being provided during the test, and we shall assume this to be 1.0, as is true in most studies. Furthermore, when an old item is tested, there is an equal probability that its image will be any of the images from 1 to n. Let S_j and N_j represent the events that image j is an s-image (an image stored for the test word) and a d-image (an image stored for some word other than the test word), respectively:

$$\frac{P(\mathrm{O}\,|\,\mathrm{D})}{P(\mathrm{N}\,|\,\mathrm{D})}=\frac{P(\mathrm{D}\,|\,\mathrm{O})}{P(\mathrm{D}\,|\,\mathrm{N})}=\sum_{j=1}^{n}\frac{P(\mathrm{D}\,|\,\mathrm{S}_j)P(\mathrm{S}_j)}{P(\mathrm{D}\,|\,\mathrm{N})}=\frac{1}{n}\sum_{j=1}^{n}\frac{P(\mathrm{D}\,|\,\mathrm{S}_j)}{P(\mathrm{D}\,|\,\mathrm{N})}$$

$$=\frac{1}{n}\sum_{j=1}^{n}\frac{P(\mathrm{D}_j\,|\,\mathrm{S}_j)\prod_{i\neq j}P(\mathrm{D}_i\,|\,\mathrm{N}_i)}{P(\mathrm{D}_j\,|\,\mathrm{N}_j)\prod_{i\neq j}P(\mathrm{D}_i\,|\,\mathrm{N}_i)}=\frac{1}{n}\sum_{j=1}^{n}\frac{P(\mathrm{D}_j\,|\,\mathrm{S}_j)}{P(\mathrm{D}_j\,|\,\mathrm{N}_j)}=\frac{1}{n}\sum_{j=1}^{n}\lambda_j. \tag{A2}$$

where $P(\mathrm{D}_j\,|\,\mathrm{S}_j)/P(\mathrm{D}_j\,|\,\mathrm{N}_j)=\lambda_j$. Now let V_k be the value of the kth feature in the probe, V_{kj} be the value of the kth feature in the jth image, m be the number of features in the probe, and M and Q be the set of indices for the nonzero features that match and mismatch, respectively.

Because a zero entry implies that a feature did not get stored, a zero entry provides no differential evidence that the image is an s-image rather than a d-image, so:

$$\lambda_j=\prod_{k=1}^{m}\frac{P(V_{kj}\,|\,\mathrm{S}_j,V_k)}{P(V_{kj}\,|\,\mathrm{N}_j,V_k)}=\prod_{k\in M}\frac{P(V_{kj}\,|\,\mathrm{S}_j,V_k)}{P(V_{kj}\,|\,\mathrm{N}_j,V_k)}\prod_{k\in Q}\frac{P(V_{kj}\,|\,\mathrm{S}_j,V_k)}{P(V_{kj}\,|\,\mathrm{N}_j,V_k)}, \tag{A3}$$

where the first product is for the features that match, and the second for the features that mismatch. Equation A3 is the same as Equation 3 in the main text, but written in a different format.

A mismatching feature in an s-image must not have been copied and hence must have had a value stored "randomly." Let $g(V)$ be the probability of storing value V by the "random" process. Make the assumption that the process of random storage produces feature values that have the same distribution as those in the population at large. Then

$$P(V_{kj}\,|\,\mathrm{S}_j,V_k)=(1-c)P(V_{kj}\,|\,\mathrm{N}_j,V_k). \tag{A4}$$

Let n_{jq} be the number of nonzero features in the jth image whose value mismatches the corresponding value in the probe. Then Equations A3 and A4 give

$$\lambda_j=(1-c)^{n_{jq}}\prod_{k\in M}\frac{P(V_{kj}\,|\,\mathrm{S}_j,V_k)}{P(V_{kj}\,|\,\mathrm{N}_j,V_k)}. \tag{A5}$$

Suppose that the system carries out calculations as if a d-image and an s-image had been stored as the result of equal amounts of study time. Then, for a d-image, the probability of storing a value is u (based on m attempts at storage with probability u^* each attempt), and the probability that the result will be V is $g(V)$. For an s-image having a feature with value V, the probability of storing it is u (again based on m attempts at storage with probability u^* each attempt), and the probability of copying it is c, and of storing it "randomly" with value V is $g(V)$. The us cancel in the numerator and denominator, giving

$$\lambda_j=(1-c)^{n_{jq}}\prod_{k\in M}\frac{c+(1-c)g(V_{kj})}{g(V_{kj})}. \tag{A6}$$

If $g(V)$ has a geometric distribution, as given by Equation 1 in the main text, then

$$\lambda_j=(1-c)^{n_{jq}}\prod_{k\in M}\frac{c+(1-c)g(1-g)^{V_{kj}-1}}{g(1-g)^{V_{kj}-1}}. \tag{A7}$$

Equation A7 is the same as Equation 4A in the main text. In Equations 4A and A7, all features are multiplied separately; in Equation 4B, the features are grouped according to equal value.

Note—There are a number of points in these derivations where other assumptions would have produced quite different results. For example, we have been assuming that the probe contains a complete set of feature values. If the probe should contain a randomly chosen subset of features, but a subset similar to that chosen for that same word at study, one would expect the s-image to contain many more nonzero features than d-images would. In this case, the number of features found in an image provides evidence concerning whether it is an s-image, regardless of the degree of matching of those features. To take a related example, suppose that there are two levels of strength represented in the images (say half the words are studied for longer times than others). Then an optimal calculation by a system that "knew" this fact would assign a probability based on each possible partition of the images into the "strong" and "weak" sets.

Distributions of the Likelihood Ratios, $\boldsymbol{\lambda}_j$

For a d-image, termed N_j,

$$E[\lambda_j \mid \mathrm{N}_j] = \sum_{\mathrm{D}_j} P(\lambda_j \mid \mathrm{D}_j) P(\mathrm{D}_j \mid \mathrm{N}_j) = \sum_{\mathrm{D}_j} \frac{P(\mathrm{D}_j \mid \mathrm{S}_j)}{P(\mathrm{D}_j \mid \mathrm{N}_j)} P(\mathrm{D}_j \mid \mathrm{N}_j) = \sum_{\mathrm{D}_j} P(\mathrm{D}_j \mid \mathrm{S}_j) = 1.0. \tag{A8}$$

Similarly, for an s-image, termed S_j,

$$E\left[\frac{1}{\lambda_j} \mid \mathrm{S}_j\right] = \sum_{\mathrm{D}_j} P(\mathrm{D}_j \mid \mathrm{S}_j) \frac{P(\mathrm{D}_j \mid \mathrm{N}_j)}{P(\mathrm{D}_j \mid \mathrm{S}_j)} = \sum_{\mathrm{D}_j} P(\mathrm{D}_j \mid \mathrm{N}_j) = 1.0. \tag{A9}$$

Although these distributions have a mean of 1.0, they are markedly skewed. To obtain an odds, n of these values are averaged. For distractors the mean obviously stays at 1.0, but by the central limit theorem the average, and hence the odds, tends toward the normal distribution as n increases. To obtain the odds for targets, an average is taken of one sample from the s-image distribution and $n-1$ samples from the d-image distribution. The mean will be $(n-1)/n + (1/n)E[\lambda_j \mid \mathrm{S}_j]$, which drops toward 1.0 as n increases and, by the central limit theorem, becomes increasingly normal.

A Note on More Complicated Models

The simplest versions of REM allow the odds to be derived in a pleasingly simple form, consisting of an average of likelihood ratios. A variety of more complex models can be proposed, in which the system is allowed to contain different categories of images of different types (e.g., on and off the list; different levels of storage strength; different degrees of similarity or overlap of features). If one makes precise assumptions about what the system "knows" about these factors, it is usually possible to write down the expression for the odds in a combinatoric form. The problem is that there the expressions do not simplify, and there is a combinatoric explosion of terms to be calculated, making simulation of the exact solution impossible in real time. This can be illustrated with one of many possible examples: suppose the system knows that half of the list items are strong, and half of them weak. When one writes down the probability of, say, the data D given a new test word, one must consider every partition of the images into two equal-sized sets, one assumed to represent the strong images and one the weak images, writing down the probability of each times the probability of that partition (which is one over the number of partitions). The number of partitions tends to be inordinately large, making real-time simulations impractical. In addition, the top and bottom of the odds expression each contains a huge sum of terms, so that the simple cancellation that occurred for the simple model does not occur, and the results no longer consist of some function of the likelihood ratios for individual images. These considerations led us to the "approximate" solutions used in REM.3, REM.4, and REM.5, in which the system continues to use the likelihood ratios in the same way, although the underlying justification is no longer strictly accurate.

APPENDIX B

In this appendix, we present an alternative treatment of the case in which all activated images tend to have a subset of features in common (such as context features).

In this class of models, it is not assumed that the system "knows" which are the context features. As a result, under the assumption that an image is a d-image, the observed degree of matching for a given feature could be due to either of two cases: (1) the feature in question is common to all images (and hence the resultant probability should be calculated as if the feature were in an s-image), or (2) the feature in question is not one of the common features (and hence the resultant probability should be calculated as in the basic REM model for d-image features).

It is not hard to adjust the basic likelihood equation to take into account this mixture of possibilities. All one need do is replace the terms in the denominator of Equation A3 with an appropriate weighted average of the terms in the numerator and denominator, where the weighting is determined by the probability of a common feature in the total set of features. Let b be the assumed proportion of common features among the whole set of features. For simplicity, we assume that b reflects the proportion of context features. Also for simplicity, let us assume that the g value for context features is the same as for word features (even though in REM.4 and REM.5 we allowed a different value of g for context features, to keep the size of the vectors and the time to run the simulations low). Thus for 20 word features and 40 context features, b would equal $40/(40+20) = 2/3$. The expression for the likelihood ratio then becomes

$$\begin{aligned}\lambda_j &= \prod_{k=1}^{m} \frac{P(V_{kj} \mid \mathrm{S}_j, V_k)}{bP(V_{kj} \mid \mathrm{S}_j, V_k) + (1-b)P(V_{kj} \mid \mathrm{N}_j, V_k)} \\ &= \prod_{k \in M} \frac{P(V_{kj} \mid \mathrm{S}_j, V_k)}{bP(V_{kj} \mid \mathrm{S}_j, V_k) + (1-b)P(V_{kj} \mid \mathrm{N}_j, V_k)} \prod_{k \in Q} \frac{P(V_{kj} \mid \mathrm{S}_j, V_k)}{bP(V_{kj} \mid \mathrm{S}_j, V_k) + (1-b)P(V_{kj} \mid \mathrm{N}_j, V_k)} \\ &= \prod_{k \in M} \left[\frac{P_{km}(i)}{bP_{km}(i) + (1-b)P_{kd}(i)} \right] \prod_{k \in Q} \left[\frac{P_{kq}(i)}{bP_{kq}(i) + (1-b)P_{kd}(i)} \right] \\ &= \left[\frac{1-c}{1-bc} \right]^{n_{jq}} \prod_{i=1}^{\infty} \left[\frac{c + (1-c)g(1-g)^{i-1}}{bc + (1-bc)g(1-g)^{i-1}} \right]^{n_{ijm}}. \end{aligned} \tag{B1}$$

As usual, the odds is just the average of these terms over the activated images. It should be noted that this version of the model considerably dilutes the diagnosticity of a matching feature with a large (i.e., unlikely) value. In the previous model, such a feature would produce a large factor in favor of the image being an s-image, since such a match would be unlikely to occur by chance. In the present model, such a match could occur because this feature was one of the ones in common to all the activated images (e.g., one of the context features). This change in diagnosticity is reflected in the denominator of Equation B1.

Because feature frequency no longer plays a large role in this model, the feature frequency approach to the word frequency effect is no longer adequate. We apply this model, therefore, to length, strength, and list strength effects only. Some other approach to the word frequency effect would therefore have to be added to this model, but we do not do so in this article.

In some respects, this model might be preferred to the one presented in the main text, because it does not require a two-phase approach to decision, with context features being used to select a set of activated images, and word features being used to make a decision within this set. Instead, a single calculation of likelihood ratios is made across all images having enough features to pass a threshold for number of relevant features; the likelihood ratios above a likelihood threshold determine the activated images. The threshold for number of relevant features is needed because the likelihood threshold in this model needs to be set well below 1.0. There are presumably many extralist images that are stored very weakly, with almost no features; these weak images would have to have likelihood ratios not too far from 1.0 (because they have few factors in the products in Equation B1) and hence would pass the likelihood threshold. If these images were not eliminated from consideration, they would swamp the likelihood ratios for the list images. Thus the model has one extra param-

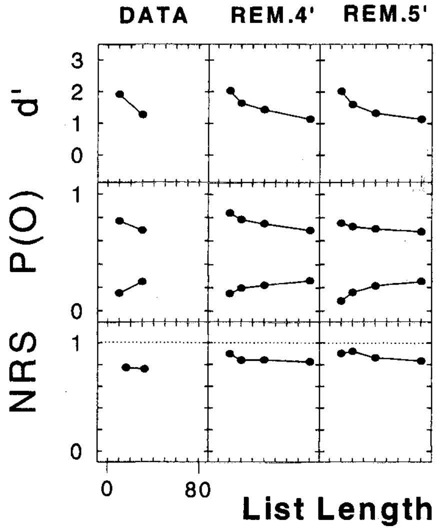


Figure B1. List length data (repeated from Figure 3) and predictions from the models presented in Appendix B. Columns 2 and 3: predictions corresponding to those in Figure 3 in columns 4 and 5, respectively (see Appendix B). The parameters were those of Figure 3 for REM.4 and REM.5, except g = 0.3; c = 0.9; $\tau_1 = e^{-4}$; τ_2 = 12; b = 2/3. The number of time steps of fluctuation between the list context and the most recent extralist item was 80; the context drift rate was 0.008.

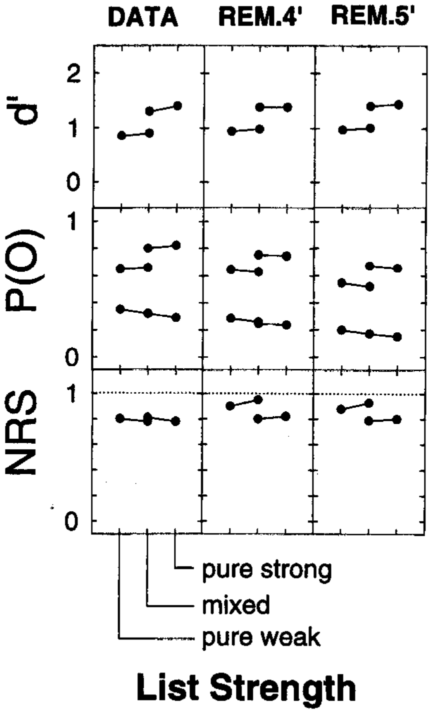


Figure B2. Strength and list strength data (repeated from Figure 5) and predictions from the models presented in Appendix B. Columns 2 and 3: predictions corresponding to those in Figure 5 in columns 4 and 5, respectively (see Appendix B). The parameters were those of Figure 5 for REM.4 and REM.5, except g = 0.3; c = 0.9; $\tau_1 = e^{-4}$; τ_2 = 12; b = 2/3. The number of time steps of fluctuation between the list context and the most recent extralist item was 80; the context drift rate was 0.008.

eter, τ_2, the threshold for number of relevant features (the number of features that have a nonzero value in both probe and image). The final decision is of course based as usual on the average of the likelihood ratios for the activated images.

We have simulated this model, and it does an excellent job: its predictions for length, strength, and list strength are as good as those shown for REM.1 through REM.5, as is illustrated in Figures B1 and B2. The parameters that differ from those in the main text were: $g = 0.3$; $c = 0.9$; $\tau_1 = e^{-4}$; $\tau_2 = 12$; the number of time steps of fluctuation between the list context and the most recent extralist item was 80; the context drift rate was 0.008.

Although this model has the advantage of a "single-pass" approach to determining retrieval, it does require that the parameter b, which is incorporated in the calculations of the likelihood ratio for each image, be adjusted properly for each experimental situation. That is, some manipulation that changes the expected number of common features would have to be allowed to change the value of b. Furthermore, any change in the subject's choice of number of context features versus content features would also have to be allowed to change b. This model will have to be contrasted with the model in the main text in future research.

(Manuscript received April 20, 1995;
revision accepted for publication January 31, 1997.)