Putting it all together: A unified account of word recognition and reaction-time distributions

Dennis Norris

MRC Cognition and Brain Sciences Unit

Cambridge, UK

Running head: Putting it all together

Contact Information:
Dennis Norris
MRC Cognition and Brain Sciences Unit,
15 Chaucer Road,
Cambridge, CB2 7EF
U.K.
Tel: +44 1223 355 294
Fax: (+44) 11123 359 062
Email: dennis.norris@mrc-cbu.cam.ac.uk

Abstract

Ratcliff, Gomez and McKoon (2004) suggested much of what goes on in lexical decision is attributable to decision processes, and may not be particularly informative about word recognition. They proposed that lexical decision should be characterized by a decision process, taking the form of a drift-diffusion model (Ratcliff, 1978), which operates on the output of lexical model. The present paper argues that the distinction between perception and decision-making is unnecessary, and that it is possible to give a unified account of both lexical processing and decision making. This claim is supported by formal arguments, and reinforced by simulations showing how the Bayesian Reader model (Norris, 2006) can be extended to fit the data on RT distributions collected by Ratcliff, Gomez and McKoon, simply by adding extra sources of noise. The Bayesian Reader gives an integrated explanation of both word recognition and decision making, using fewer parameters than the diffusion model. It can be thought of as a Bayesian diffusion model, which subsumes Ratcliff's drift-diffusion model as a special case.

Introduction

Lexical decision is the most widely used task for studying visual word recognition. But could it possibly be the case that this task really tells us very little about the process of word recognition? This is the implication of a recent paper by Ratcliff, Gomez and McKoon (2004). Ratcliff, et al. have argued that much of what goes on in lexical decision is attributable entirely to the decision process itself, and "the lexical decision task may have nothing to say about lexical representations or about lexical processes such as lexical access."(p160).

Even the word frequency effect, the cornerstone of models of word recognition, might have little to do with the process of lexical access. Although there have been previous claims that lexical decision might be strongly influenced by decision processes (e.g. Balota & Chumbley, 1984), Ratcliff et al's arguments carry more force because they are backed up by detailed quantitative modeling of data from an extensive series of experiments. Furthermore, their claims go beyond casting doubts on the value of the task and have significant implications for models of lexical processing. In fact, Ratcliff et al. argue that there are no current models of word recognition that are compatible with their findings.

Here I will show that their data is consistent with the idea embodied in many current models of word recognition that lexical decision involves integrating evidence (or activation) directly from lexical representations, as in the Bayesian Reader (Norris, 2006), the Multiple Read-Out Model (Grainger & Jacobs, 1996), the Dual-route cascaded model (Coltheart, Rastle, Perry, Langdon & Ziegler, 2001) or REM-LD (Wagenmakers, Steyvers, Raaijmakers, Shiffrin, van Rijn, & Zeelenberg, 2004). In all of these models,

lexical decision is driven by the same lexical processes that determine the identity of individual words in tasks such as perceptual identification or reading.

Ratcliff et al. support their claims by showing how Ratcliff's (1978) diffusion model (DM) can provide very precise fits to reaction time (RT) distributions and error rates in experiments using the lexical decision task. Using the DM, Ratcliff et al. were able to model the effect of variables such as word frequency, list composition, type of nonword, and stimulus repetition, all of which have been considered as providing valuable insights into the word recognition process. They suggested that factors such as word frequency cause variations in a ‘wordness’ signal that determines the drift rate of the decision process modeled by the DM, and not the process of word recognition itself. Higher frequency words generate larger wordness values than lower frequency words, and nonwords produce negative values. In addition, wordness is also modulated by the nature of the nonwords used in the experiment. In Ratcliff et al.'s words "The lexical system that feeds information to the decision process may have many facets, but once information is output from the system, it can be considered unidimensional" (p176, lines 22-25). However, these conclusions do not follow from any direct test of the models themselves, but depend on the assumption that the DM gives the correct explanation of the decision processes involved in lexical decision. If it does, then the output of the lexical system would indeed have to be unidimensional. The alternative explanation offered here follows from the fact that the DM can be considered to be a special case of a more general decision procedure that can make optimal decisions based on input from multiple sources, rather than being reliant on a unidimensional input. In the case of lexical decision, those multiple sources are the lexical representations themselves. This

analysis explains why it is that factors such as word frequency should appear to influence wordness or drift rate. This opens the way for a completely different interpretation of Ratcliff et al's data. It may seem too simple to be true, but the differences in RT distributions for words of different frequencies are actually due to word frequency after all. More accurately, the effect of 'word frequency' comes about because higher frequency words have larger priors than low frequency words. I will make the case for this more direct interpretation of frequency effects in lexical decision by illustrating the formal relationship between the DM and the Bayes-optimal decision process embodied in the Bayesian Reader. This shows how effects that are captured by variability in the drift rate in data fitted by the DM, can be directly caused by differences in word frequency. Although Ratcliff et al. are correct in attributing the effect of word frequency to something akin to drift in the DM, the implications for models of word recognition are rather different from those that they suggested.

To illustrate that this is more than just a theoretical possibility, I will also present simulations using a modified version of the Bayesian Reader. These simulations have no parameters corresponding to drift rate. Instead, the differences in the RT distributions of words of different frequencies follow directly from the frequencies of those words given in CELEX (Baayen, Piepenbrock & Gulikers, 1995).

Although Ratcliff et al. criticize existing models of word recognition for not being able to fit their data, the DM offers only a partial explanation of the data. As Ratcliff et al. themselves admit "The model can account for the data, and it can provide an explanation of the decision process, but it does not provide insights into lexical representations or how they are accessed"(176). The main burden of explanation still lies with an

unspecified lexical system that can produce drift rates modulated by the various factors known to influence lexical decision. So, although Ratcliff et al. suggest that "The interpretation of data offered by the diffusion model is much simpler and much less mysterious than has been the case with many alternative theoretical accounts of processing" (p180), the mystery is as great as ever. If Ratcliff et al. are correct, we know nothing at all about the most scientifically interesting component of lexical decision: how words are recognized. Furthermore, there is a deep mystery at the heart of Ratcliff et al's interpretation: What is this thing called wordness that drives the decision process, and why should the lexical system produce a measure of wordness? Stranger still is their assumption that word frequency, or even lexical status, has no influence on the time taken for the lexical system to generate the wordness value.

Ratcliff et al. assume that wordness is unidimensional signal that combines measures of 'lexical strength' and 'orthographic wordlikeness' (see Figure 4 of Ratcliff et al.). The wordness signal is specific to the lexical decision task. But, whatever wordness is, it, or at least the 'lexical strength' component, looks remarkably like word frequency. Across a range of tasks, including lexical decision, word frequency effects are very similar (Monsell, Doyle & Haggard, 1990; Schilling, Rayner & Chumbley, 1998). For example, one of the most well established characteristics of lexical decision is that there is an approximately logarithmic relation between RT and log frequency (e.g. Balota, Cortese, Sergent-Marshall, Spieler & Yap, 2004; Murray & Forster, 2004; Whaley, 1978). Coincidentally, this is exactly the same function as is observed in tasks requiring identification of individual words, such as perceptual identification (Howes & Solomon, 1951), reading aloud (Balota, Cortese, Hutchison, Neely, Nelson, & Simpson, 2000;

Balota & Spieler, 1998; McCusker, 1977; Spieler & Balota, 1997, 2000), and eye movements (Murray, 2001, 2007; Murray & Forster, 2008). These tasks cannot be performed by a binary decision process like the DM because the perceptual system needs to identify which particular word is present. There is an active debate about whether the form of this relationship is really logarithmic, or might instead be a function of rank frequency (Murray & Forster, 2004, 2008), or contextual diversity (Adelman & Brown, 2008; Adelman & Brown & Quesada, 2006). However, the difference between these functions is small and, for present purposes, the critical point is that it appears to be the same whether the task is lexical decision or word identification. So, is wordness really frequency? It can't be. Nonwords are assumed to have negative values of wordness and lexical strength, with nonwords that are orthographically less like words having larger negative values of wordlikeness. Frequency cannot be negative. Furthermore, Ratcliff et al's DM simulations reveal that the values of wordness (or drift rates) for words change as a function of the nature of the nonwords used. So wordness is not even a fixed property of individual words. The wordness of words has to be modulated by the orthographic wordlikeness of the nonwords used in the experiment. Quite how this comes about is not specified.

In the Bayesian Reader, the similarity between lexical decision and tasks involving word identification isn't just a coincidence – they have the same underlying explanation. Norris (2006) showed that the Bayesian Reader predicts a logarithmic relation between frequency and RT in both lexical decision, and in tasks requiring identification (e.g. eye movements in reading). In those simulations, frequency (or priors) was estimated simply on the basis of the raw frequency counts in CELEX (Baayen, Piepenbrock & Gulikers,

1995). The model naturally produces the logarithmic function even though the representation of frequency is linear. The reason the model produces a logarithmic frequency function is that it performs word recognition optimally (Baum & Veeravalli, 1994; Veeravalli, Tartakovsky & Dragalin, 1995; see Adelman & Brown, 2008, for a simpler derivation). Regardless of whether the task involves the identification of specific words, or lexical decision, an optimal system will give rise to a logarithmic relation between ease of identification and frequency. Optimality provides the critical link between the Bayesian Reader and the DM. Both make optimal decisions based on the accumulation of noisy data. However, in the DM optimal performance can only be achieved when the input is unidimensional. The Bayesian Reader, on the other hand, makes optimal decisions based on multidimensional input (see Norris (2006) and Norris & McQueen (2008) for a discussion of what constitutes optimality in word identification and lexical decision). It is equally applicable to binary decisions, as in lexical decision, or decisions involving many alternatives, such as identifying which specific word is present. When dealing with input that varies on only a single dimension, the two models are equivalent. The problem with using the DM to model lexical decision is that the input to the perceptual system is not unidimensional. The input consists of letter strings, and there is no single perceptual dimension along which words and nonwords can be partitioned. Whether a letter string is a word or nonword is little more than an accident of the vocabulary. The only way to perform the task accurately is to determine which orthographic forms corresponds to a real words, and which do not. In order to achieve this, the DM needs assistance from an unspecified lexical system.

What kind of lexical model could provide the input to the DM?

Ratcliff et al. claim that "we can focus on models that simply hand the decision process a single value of goodness of match" (p 181). This conclusion follows from the nature of the DM parameters derived from fitting the model to the data. Word frequency, lexical status, and the nature of the nonwords used in the task, all manifest themselves purely in a change in drift rate. They have no effect on other parameters in the DM such as the amount of time taken by processes outside the decision mechanism. In the DM the drift rate remains constant throughout each trial, and the task of a lexical model that could interface with the DM is to provide a single value that determines drift rate. All output from the lexicon must therefore be funneled through the bottleneck of a single value of drift. Because drift is constant, and factors such as frequency have do not alter the time at which the diffusion process begins, the output of the lexicon must always be delivered at a fixed point in time. Any lexical model that could provide the required input to the DM would therefore have some very peculiar properties indeed. Lexical processing would be constrained to always take exactly the same amount of time, regardless of word frequency, the difficulty of the discrimination between words and nonwords, or even whether the target was a word or a nonword. At the end of that time the lexical model would output a single fixed value of goodness of match (drift) which would not be updated in the light of any further analysis that might be performed.

Although the 'goodness of match' or wordness value indicates whether the target is a word or a nonword, the lexical system must not act directly on that value. Instead, it must output that value to the diffusion model. The lexical model thus has access to information about the lexical status of the input. However, it is not permitted to act on that

information itself. Instead it must instruct the diffusion model to make a ‘yes’ or a ‘no’ response, and also tell it how quickly to make that response. Furthermore, given that the drift rate in the DM remains constant over time, the lexical system is never allowed to revise or update the wordness value. The lexical system has to output a single value at a fixed point in time and clamp that value until a response is made.

One reason why the lexical model might need to feed into a diffusion process in order to make lexical decisions is that the wordness value produced by the lexicon might be noisy. In that case a diffusion process would provide an optimal procedure for accumulating the noisy evidence to make a decision. Indeed, this does appear to the interpretation favored by Ratcliff et al. But why would the lexical system output a fixed wordness value that was subject to noise, and then not carry out any further processing? This still begs the question of how and why the lexicon might output a wordness value at all, other than because that is what is required to drive the DM.

The idea that lexical processing takes the same amount of time regardless of frequency, or even lexical status, is counterintuitive to say the least. Fortunately, there is an alternative interpretation that maintains the underlying spirit of the DM analysis while being more consistent with existing models of lexical processing. As will be shown below, the DM is a special case of the optimal decision process embodied in the Bayesian Reader. The Bayesian Reader can also be considered to be a random-walk or diffusion model, and it is much more closely related to the DM than it is to other models of word recognition. Indeed, in a later paper Wagenmakers, Ratcliff, Gomez and McKoon (2008) noted that “Given its conceptual similarity to the diffusion model (i.e., optimal decision making based on sequential sampling of noisy information), we expect that the Bayesian

Reader would be able to capture many of the qualitative patterns of results obtained in the current study" (p158).

Despite the formal similarity of the DM and the Bayesian Reader, in the present context the two analyses have radically different theoretical implications for the relationship between lexical and decision processes. In the DM, lexical processing operates in a fixed time period before the decision process can begin. That is, lexical processing takes place during part of the non-decision component of the DM (given by the parameter T_{er}). In the Bayesian Reader, lexical processing and decision processes operate simultaneously and are fully integrated. The analogous parameter in the Bayesian Reader analysis represents operations not involved in either decision or lexical processes: the duration of lexical processing and the duration of decision processing are one and the same thing. In the Bayesian Reader factors such as word frequency alter the duration of lexical processing, just as one would expect on the basis of any of the current models of word recognition. Furthermore, frequency alters lexical processing in a way that guarantees that it will appear to influence drift rate when the data are fitted by the DM.

By taking the more general approach of the Bayesian Reader, and integrating evidence from multiple sources (words), it is possible to offer an explanation of lexical decision that encompasses a complete model of word recognition. Importantly, this goes beyond simply generating a set of parameters that describe performance, and delivers an explanation of how and why it is that various lexical factors can modulate drift rate. There is no need to bolt a separate decision process onto a lexical model, and no need to rely on the notion of wordness: perception and decision-making follow the same principles. Indeed, they are one and the same thing.

In one sense the present enterprise can be seen as rising to the challenge set out by Ratcliff et al. to produce a lexical model that can account for their data. Both models are designed to make optimal decisions based on the accumulation of noisy data. They are both members of the family of random-walk or diffusion models.

My central concern with the analysis presented by Ratcliff et al is that, by assuming a separation between lexical processing and decision making, they have undersold the potential of this class of model to offer insights into the mechanisms of perceptual decision making. By choosing to model lexical decision with the unidimensional DM, and a fixed drift rate they have placed an artificial constraint on the form of the explanations that might emerge.

The Diffusion Model

The operation of the DM is represented graphically in Figure 1. The DM can be thought of as accumulating noisy evidence from the input. The input to the diffusion process is given by the drift rate, *v,* which determines the mean rate of approach to the decision boundary. The operation of the DM is perhaps easiest to appreciate in terms of a random walk, the discrete version of a diffusion model. At each point in time, the model takes a discrete step toward either an upper or lower response boundary. The direction of the step is determined by adding noise to the drift. The average rate of approach to the decision boundary is therefore given by the drift, but there will be random variation around this mean. It is this random variation that produces the RT distribution, and leads to the possibility that, on some trials, the walk may reach the wrong decision boundary

resulting in an error. In the full version of the DM used to simulate the lexical decision data, there are several extra parameters representing additional sources of noise in the decision process. The full set of DM parameters used in the lexical decision simulations is: boundary separation, a; nondecision component of RT, T_{er}; standard deviation in drift rate across trials, η; range of distribution of starting point, s_z; range of distribution in the nondecision component of RT, s_t; starting point, z; drift rates for each condition, v_i.

Insert figure 1 about here

Ratcliff et al found that, within each of their experiments, the difference between words of different frequency could be captured entirely by changes in the drift rate parameter of the DM. A further important finding was that, in between-experiment comparisons, the effect of nonword type also manifested itself in a change in the drift parameter. When nonwords were random letter strings, the drift rates of words were larger than when the nonwords were pronounceable letter strings. Similar data and analyses have also been presented by Yap, Balota, Cortese and Watson (2006). Yap et al. interpreted their data as evidence for diffusion model rather than the two-process model proposed by Balota and Chumbley (1984).

Optimal decision making and Sequential Probability Ratio Tests

As noted above, an important property of the DM is that it can implement the optimal procedure for making binary decisions between alternative inputs varying in their value on a single dimension. When the only source of noise in the DM is in the input, the DM is a continuous version of the Sequential Probability Ratio Test (SPRT, Wald, 1947). The

SPRT is the optimal procedure for making a decision with a given level of accuracy (say 95%), based on the minimum number of samples.

The SPRT formed the basis for early models of choice reaction time. Stone's (1960) theory of choice-reaction time drew directly on the SPRT and likelihood ratio tests. This work was developed further by Laming (1968), who described his ideas in terms of a random walk, where each step was determined by the accumulation of evidence. Related ideas have been proposed by Carpenter in his LATER model (Carpenter, 1981, 1999). The DM represents a further development of these principles by incorporating extra sources of internal noise into the decision process. Recently, Bogacz, Brown, Moehlis, Holmes and Cohen (2006) have provided a very thorough mathematical treatment of decision making in two-alternative forced-choice tasks, and the reader is referred to that paper for a more in-depth analysis. A more informal treatment is given in Bogacz (2007).

The Bayesian Reader is also a random walk model and a form of SPRT. The Bayesian Reader implements a form of Multi-hypothesis Sequential Probability Ratio test (MSPRT, Baum & Veeravalli, 1994; Dragalin, Tartakovsky & Veeravalli, 1999, 2000; Bogacz, 2007). Under conditions where the model has the task of identifying individual words, the hypotheses are words. This can be thought of as a random walk in multiple dimensions, where there is a separate boundary for each word. As noted earlier, the basic Bayesian decision procedure can be applied to two-alternative forced choice decisions. Both the Bayesian Reader and the DM are models of how optimal decisions are made on the basis of samples from a noisy input. The best way to appreciate the relationship between the Bayesian Reader and the DM is therefore in terms of how binary decisions are performed in the discrete SPRT.

The decision process can be characterized as computing whether it is more likely that the sequence of observations was produced by sampling from distributions with mean values *W1* or *W2* (Figure 2). The difference between *W1* and *W2* is the drift rate, or strength of the signal. The likelihoods $f(y_i| w_1)$ and $f(y_i| w_2)$ are given by the heights of the density functions at the point given by the value of the sample. The likelihood of the sequence of samples y_0 - y_n is then given by the product of the likelihood ratios of the individual samples.

$$\frac{l_{w1}}{l_{w2}} = \frac{f(y_1 \mid w_1) f(y_2 \mid w_1) f(y_3 \mid w_1) \ldots f(y_n \mid w_1)}{f(y_1 \mid w_2) f(y_2 \mid w_2) f(y_3 \mid w_2) \ldots f(y_n \mid w_2)} \qquad (1)$$

This can also be expressed in terms of computing the summed log-likelihood ratio:

$$\log Z = \log \frac{f(y_1 \mid w_1)}{f(y_1 \mid w_2)} + \log \frac{f(y_2 \mid w_1)}{f(y_2 \mid w_2)} + \log \frac{f(y_3 \mid w_1)}{f(y_3 \mid w_2)} \ldots \qquad (2)$$

This is then a random walk where the size and direction of the step at each point is given by the log-likelihood ratio:

$$I^n = I^{n-1} + \log \frac{f(y_n \mid w_1)}{f(y_n \mid w_2)} \qquad (3)$$

and decisions can be made whenever the summed log-likelihood ratio crosses the boundaries given by Z_1 and Z_2. Alternatively, the random walk could be on a probability scale rather than a likelihood scale, where

$$P(w_1 \mid y_o \ldots y_n) = \frac{l_{w_1}}{l_{w_1} + l_{w_2}} \ . \qquad (4)$$

Insert figure 2 about here

Optimal lexical decision

Consider how lexical decision should be performed optimally. Assume that there is a very simple lexicon where possible word forms vary only on a single perceptual dimension. Imagine that equally separated points on that dimension correspond to possible letter strings, and half of the points correspond to words. The remaining points, where the letter strings do not correspond to words, must therefore be nonwords. Now we have a set of values $w_1 \ldots w_m$, corresponding to words, and a set $nw_1 \ldots nw_m$ corresponding to nonwords. For the purposes of exposition I will assume that the set of possible nonwords is known, and that the likelihood of each nonword can be calculated explicitly. However, as shown in Norris (2006), a variety of heuristic procedures can be used to estimate nonword likelihoods. How should one make an optimal decision as to whether the input was generated by a word or a nonword? If the values on the dimension representing words and nonwords are intermingled it will no longer be possible to test a simple hypothesis such as "Was the input generated by *W1* or *W2*?". This means that we can no longer use the SPRT or DM directly on the input values. The appropriate procedure is to ask whether the total evidence for words is greater than the evidence for nonwords. That is, the question now becomes: "Is the summed likelihood of the words greater or less than the summed likelihood of the nonwords?". In terms of the random walk formulation this leads to:

$$I^n = I^{n-1} + \log\left(\frac{f(y_n \mid w_1) + f(y_n \mid w_2) + f(y_n \mid w_3) + \ldots f(y_n \mid w_m)}{f(y_n \mid nw_1) + f(y_n \mid nw_2) + f(y_n \mid nw_3) + \ldots f(y_n \mid nw_m)} \ldots \right). \qquad (5)$$

Importantly, we can now extend this to allow for the possibility that words might differ in frequency of occurrence:

$$I^n = I^{n-1} + \log\left(\frac{f(y_n \mid w_1)P(w_1) + \ldots f(y_n \mid w_m)P(w_m)}{f(y_n \mid nw_1)P(nw_1) + \ldots f(y_n \mid nw_m)P(nw_m)} \right), \qquad (6)$$

where $P(w_i)$ is the probability of the ith word and $P(nw_j)$ is the probability of the ith nonword. Assuming that, overall, words and nonwords occur equally often, there will be no initial bias towards either a word or a nonword response.

$$\sum_{i=0}^{i=m} P(w_i) = \sum_{i=0}^{i=m} P(nw_i) = 0.5 \ . \qquad (7)$$

Equation (7) corresponds to the case where $P(w_i) = P(nw_i) = 1/2m$.

Now when the input corresponds to a real word, the effective step size will increase as a function of frequency of the word and, to a lesser extent, of the frequency of any neighboring words that also contribute to the overall likelihood. When the target word is a high frequency word, the numerator of Equation 6 will be larger than when the target is a low frequency word. Equation 6 is effectively a log-likelihood ratio formulation of the Bayesian Reader, and the $f(y_n \mid w_m)P(w_m)$ are derived from individual lexical entries. There is no need to collapse the output of the lexicon onto a single dimension that is then

input to a separate decision process. Equation 6 maps directly from lexical representations to the likelihood ratio that determines the response. The critical point here is that when the SPRT is extended to perform the kind of decision required by the lexical decision task, it follows automatically that higher frequency words will tend to approach the decision boundary faster than lower frequency words. That is, when the data are fitted with the DM, high frequency words will appear to have a larger drift rate than lower frequency words. Note that one critical difference between the two formulations is that in the Bayesian Reader the decision variable is being continuously modulated by the evolving analysis in the lexicon, whereas in Ratcliff et al's analysis the lexicon must first complete its processing in a fixed amount of time before producing a single constant value that can then be passed to the decision process. A consequence of this is that the DM performs the lexical decision task sub-optimally. Given the value of drift produced by the lexicon, the DM makes optimal use of that information. However, the lexicon is constrained to identify all words in the same amount of time, regardless of frequency or of similarity to other words or to nonwords in the experiment. An optimally designed lexicon should require less evidence to recognize high frequency than low frequency words (see Norris, 2006, for a full explanation). This combination of DM and lexical model must therefore perform less efficiently than the Bayesian Reader.

Both Ratcliff et al. and Yap, Balota, Cortese and Watson (2006) found that drift rates also tended to increase when less wordlike nonwords were used. This also follows directly from the analysis presented here. In our simple illustration, the easiest way to make the words and nonwords less similar is to increase the distance between them. If nonwords are further away from the words then, when a word is presented, the likelihood

of the nonwords will be less than when the words and nonwords are closer together. So, when nonwords are, for example, consonant strings rather than pronounceable nonwords, the denominator of Equation 6 will decrease and the effective drift rate will increase. Both word frequency and the kind of nonwords used will therefore alter drift. In the DM analysis these factors are assumed to influence lexical strength and orthographic wordlikeness respectively. These then have to be lumped together into the single dimension of 'wordness'. There is no explanation for why they should have similar effects, nor how it is that the lexicon produces the necessary signals. In contrast, the analysis in terms of the optimal decision procedure provides a clear and transparent account of how these quite distinct factors will both influence the effective drift rate. There is no need to boil the lexical information down to a unidimensional signal that can then be input to a decision process: the lexical information itself drives the decision directly. This fills the explanatory gap in the DM analysis of lexical decision data. The DM fits indicate that frequency and nonword type both influence the rate of approach to the decision bound (drift rates), and the optimal analysis shows how this can follow automatically from a decision process that integrates information directly from lexical representations. It is important to appreciate that the optimal decision procedure is not in some sense providing input to the DM. The optimal decision procedure replaces the DM entirely. The Bayesian Reader does output a single value, but that is the final estimate of the probability that the input is a word. No further processing is required. Nothing in equation 6 corresponds directly to a value of drift that can be input to the DM. The log likelihood ratio in equation 6 is not the value of wordness, but the size of the next step of

the random walk. Furthermore, this value may change as processing progresses. In the DM the drift rate is fixed.

If data produced by an optimal system is fitted by the DM, then the DM should be able to provide good fits to conditions varying in frequency and nonword type by varying the drift rates for each condition. This is exactly what Ratcliff et al. did. However, given appropriate representations of the stimuli, in an optimal system the differences between the conditions should emerge naturally as a consequence of the frequencies of the words and the similarity of the words and nonwords. There should be no need to have parameters explicitly representing drift. Although this conclusion follows necessarily from the formal argument presented so far, we can provide a more concrete demonstration that an optimal model should behave as predicted by simulating the data from Ratcliff et al. with the Bayesian Reader. Even though there are some limitations imposed by the way the model is currently implemented, the simulations behave exactly as the formal derivations would suggest.

The Bayesian Reader

Although the Bayesian Reader is not specifically a model of lexical decision, when performing lexical decision it implements a version of the optimal procedure described above. There are two main respects in which the Bayesian Reader differs from this procedure. The first is that the Bayesian Reader uses a multidimensional input representation. Words are represented as a concatenation of letter vectors. In the current implementation, each letter is represented by a 26 element vector where one element is set to 1.0, and the remainder to 0.0. A five-letter word is therefore represented as a vector

of 130 elements or features. Note that there is no theoretical commitment as to the exact form of the featural representation. The input to the model is generated from the vector representing the input letter string. Successive input samples are generated by adding Gaussian noise to the input vector. The ratio of the amount of noise relative to the difference between the vector representations of words versus nonwords corresponds roughly to the drift rate in the DM, in that it is a measure of the strength of the input signal. The larger the noise, the more samples will be required to identify the word. Second, there is no explicit representation of nonwords in the lexicon. In the original version of the model, nonword likelihood was estimated by a procedure that calculates the likelihood that the input was generated by a letter string that is at least one letter different from a real word. In the simulations reported here, nonword likelihood is estimated from the probabilities of the strings of letters that make up the words. The full procedure is described in the Appendix. This method of lexical decision takes advantage of the fact that the sum of the probabilities of all possible strings of letters (both words and nonwords) must be 1.0. There is only a finite amount of probability to go round, and an estimate of the likelihood that the input is a nonword can be derived from what's 'left over' after summing the probability of the strings that form words. Nonword likelihoods can therefore be estimated entirely from what is known about words. Note that this component of the calculations computes likelihoods of words and nonwords independently of their frequencies, and these likelihoods have to be multiplied by the priors (word frequencies) to compute the final P(input is a word | input).

The computations required to perform lexical decision are actually an essential part of normal everyday reading. If a reader encounters the letter string "worb" it isn't sufficient

just to categorize it as ‘word’ on the basis that this is the closest word. The reader must be able to determine whether the input is a known word, or a new word or typographical error (cf Chaffin, Morris & Seely, 2001). This means that it isn’t necessary to postulate extra task-specific processes for lexical decisions because the necessary decisions are an integral part of lexical processing itself. Indeed, they are a necessary part of object recognition in general. An important component of successful object recognition is to be able to decide whether an object is one that is already familiar, or a new object that requires a new representation. As Norris and McQueen (2008) explained, exactly the same procedure is required in spoken word recognition. In fact, it is particularly important in recognizing continuous speech because any failure to appreciate that part of an utterance corresponds to an unknown word can lead to a complete misanalysis of the whole input.

Norris (2006) reported that the RTs produced by the original Bayesian Reader model have a very small variance. That model would not be able to simulate the data from Ratcliff et al. Given the formal similarity between the DM and Bayesian Reader, one might wonder why the Bayesian Reader is unable to simulate RT distributions. However, the DM is much more than an implementation of the SPRT. The ability of the diffusion model to produce accurate simulations of the distributions of both correct and error RTs depends on the way the model incorporates different sources of variability. All of the different sources of noise in the DM have been found to be essential in order to generate accurate simulations of RT distributions (Ratcliff & McKoon, 2008).

In the basic form of the Bayesian Reader, the only source of variability is in the sampling noise. Additional sources of variability analogous to those in the DM can be

added to the Bayesian Reader in a quite straightforward way. For example, there is no problem adding variability in the starting point (T_{er}) , or in the initial probability (the equivalent of s_z). However, it is less straightforward to incorporate variability in drift (η). The drift rate in the model is effectively determined by the sampling noise and the distance between words and the average location of nonwords. This can be altered on a trial-by-trial basis by varying the sampling noise, but the sampling noise can never be negative and create the equivalent of a negative drift used in the DM.

For present purposes this is not too important, because the consequence is likely to be restricted largely to error RTs. As explained in the text accompanying Figure 1 of Ratcliff et al., variability in starting point (s_z) and drift rate (η) have opposing effects on the speed or error responses. If errors are produced mainly by variability in drift rate, error responses will be slow; if they are produced by variability in starting position, they will be fast. In the simulations reported here, the errors are produced largely by variability in the starting probability. These errors are sometimes much faster than observed in the data. Because this discrepancy is almost certainly nothing more than a consequence of the way trial-by-trial variability in drift is implemented (i.e. it tells us where the noise is in the system, and what form the noise distribution might take), the simulations did not try to fit error RTs, but error RTs are reported for completeness.

There are a number of ways in which the model might be modified to produce negative drift for words, and positive drift for nonwords. One possible procedure would be to add trial-by-trial variability into the representation of frequency. This could be considered to result from noise in the memory retrieval process. If noise sometimes led to words having an effective frequency of near zero on a particular trial, this would act like

negative drift and lead those words to be classified as nonwords, and the error responses would tend to be slow. If internal noise sometimes led to the nonwords being treated as words with a low frequency, this would produce slow error responses to nonwords. However, there is no clear basis on which to choose a noise function. For example, should the noise be linear or Gaussian; should it be added to linear frequencies, or log frequencies? So, there is scope for modifying the Bayesian Reader to incorporate sources of variability more closely analogous to those used in the DM, however, no effort has been made to systematically investigate these possibilities. At noted above, the only theoretical issue that this would speak to is to establish the exact locus and form of the noise in the system.

It is important to note that the way the Bayesian Reader maps onto the DM differs according to whether the task is lexical decision or identification. Adelman and Brown (2008) also presented an analysis of the Bayesian Reader in terms of a random walk, and they suggested that frequency would correspond to the starting point of a multiway random walk. However, their analysis is only appropriate for the case where the model is required to identify individual words. Variations in the starting point correspond to biases to respond with different words – high frequency words having stronger biases than low-frequency words. However, in lexical decision the responses are word/nonword decisions, not specific words. The starting point should always correspond to a probability of 0.5 that the target is a word, for the simple reason that words and nonwords are equally likely.

Simulations

Having established the points of similarity and difference between the DM and the Bayesian Reader, the next section presents simulations[1] to demonstrate that, when combined with similar sources of noise to those used in the DM, the Bayesian Reader can simulate the data from Ratcliff et al. simply by taking account of the differences in priors between words of different frequencies. For brevity, only the simulations of Ratcliff et al.'s experiments 1 and 2 will be reported. Ratcliff et al.'s experiments manipulated word-frequency, the nature of the nonwords, and the proportion of words of different frequencies. Experiments 1 and 2 both contained high-frequency, low-frequency, and very low-frequency words. In experiment 1, the nonwords were pronounceable pseudowords, and in experiment 2 they were unpronounceable letter strings. The data from these experiments, collated from Ratcliff et al.'s tables 3 and 5, are reproduced in tables 1 and 2, along with the DM simulations and the new simulations reported here. The form of the RT distributions is captured by the quantile RTs. Quantile RTs for the simulations are all derived from the simulated data in exactly the same way as for the human data.

The parameters used in these simulations were: yes threshold, no threshold, variation in initial probability that the input is a word (uniform noise. This parameter determined the range of the initial value of P(a word)), trial-by-trial variation in sampling noise (Gaussian noise), nondecision time, noise on the nondecision time (uniform noise), and the slope relating model steps to RT. The sampling noise of the model was fixed at 3.0. There are no parameters representing frequency or type of nonword. Optimization was performed using the same Chi square minimization procedure as described on p168 of

Ratcliff et al., with the exception that only the quantiles for correct responses were fitted. Optimization was carried out using the Appspack package (Gray & Kolda, 2006; Kolda, 2005).

The simulations were based on a set of stimuli constructed along the principles used by Ratcliff et al. It was necessary to construct a new set of stimuli for the simulations as the current implementation of the Bayesian Reader can only deal with stimuli of a single length. Word frequencies were derived from CELEX (Baayen, Piepenbrock, & Gulikers, 1995) rather than the Kucera and Francis (Kucera & Francis, 1967) counts used by Ratcliff et al.. The model's lexicon was the same as used in the simulations reported in Norris (2006) and contained 4106 5-letter words. Two hundred words were selected in each of the high, low and very-low-frequency bands. Mean frequencies per-million were almost identical to those of the words used by Ratcliff et al.. Stimulus selection and matching was performed using the 'Match' program (van Casteren & Davis, 2007). Pseudowords were created by randomly changing vowels in the word targets, and random letter strings were formed by selecting letters at random with the constraint that none was pronounceable. There were therefore 600 nonwords of each type. For a complete iteration all words were run through the model 60 times and nonwords 20 times, so as to generate the same number of simulated data-points in each condition. Each of these data-points was then used to generate a further 20 data-points by adding extra time to the RT as determined by the noise in the non-decision time.

The simulations are a very challenging test of the model. All of the data will be simulated using the real frequencies of the words. In Ratcliff et al.'s DM fits, drift rates were free parameters. There are no comparable free parameters in the current

simulations. Any differences between the conditions within each experiment are entirely determined by the properties of the actual items used in the simulations. Note that whereas the all of the Bayesian Reader simulations require seven free parameters, the DM required nine parameters to fit the data from experiments 1, 2 and 5, and ten for experiments 3, 4 and 6.

The results of the simulations are shown in tables 1 and 2, and the best fitting parameters are shown in table 3. Remarkably, given the constraints, the simulations fit the data very well.

Insert tables 1-3 about here

Ratcliff et al. suggested that there were six main features of the data for modeling:

1. For words, accuracy increased and RT decreased (for both correct and error responses) as word frequency increased, and this was true whether the nonwords were random letter strings or pseudowords. The differences between the high- and low-frequency conditions were larger when the nonwords were pseudowords.

2. For words, RTs were shorter and accuracy was higher when the nonwords were random letter strings than when they were pseudowords.

3. For nonwords, correct responses had about the same RTs as correct responses for the slowest words. Responses were faster for random letter strings than for pseudowords, and accuracy was a little higher.

4. Most of the differences in RTs that occurred with increased word frequency were due to decreased skew of the RT distribution.

5. However, when the nonwords were pseudowords, there was a moderately large effect of frequency on the leading edge of the RT distribution: The leading edge for high frequency words was shorter by 40 ms than the leading edges of the RT distributions for lower frequency words and nonwords. When the nonwords were random letter strings, the differences were considerably smaller, about 12-14 ms, but still significant.

6. With random letter strings, error RTs were shorter than correct RTs. Error RTs were also shorter than correct RTs with pseudowords but only for fast subjects; for slow subjects, error RTs were about the same as or longer than correct RTs.

The Bayesian Reader simulations capture all of these characteristics of the data, apart from the pattern of error RTs. Error RTs in the Bayesian Reader are consistently faster than correct RTs. This is also true of simulations of experiments 4-6, which are not reported here. In contrast, the DM has no more difficulty simulating error RTs than correct RTs. As noted earlier, this is to be expected from the different way in which trial-by-trial variation in drift rate is implemented.

The success of the Bayesian Reader simulations is a considerable achievement. Instead of adding a decision process to a lexical model, all that has been required is to add noise to an existing model of word recognition. Similar moves might also be possible within MROM (Grainger & Jacobs, 1996) or REM-LD (Wagenmakers et al., 2004). The Bayesian Reader already simulates such as the form of the function relating frequency to RT, and performance in different forms of masked priming task (Norris & Kinoshita, 2008), and how neighborhood effects can change from predominantly facilitatory in lexical decision (for reviews see Andrews 1997, Perea & Rosa, 2000) to inhibitory in natural reading (Pollatsek, Perea & Binder, 1999). The same model can

simulate RT distributions just by adding variability. Importantly, the ability of the model to simulate the changes in RT distribution, and how they vary between conditions, does not require any free parameters corresponding to the drift rates in the DM. In the DM there is a separate drift rate for each condition in each experiment. In the Bayesian Reader these differences follow automatically from the frequencies of the words, as given in CELEX, and from the nature of the nonwords used in the simulations.

The current simulations also address a shortcoming of the lexical decision simulations reported in Norris (2006). In those simulations RTs for nonwords tended to be faster than to words. However, the response thresholds were kept constant throughout that set of simulations. Here we see that when the model is fitted directly to data, it correctly simulates the relative speed of word and nonword responses.

The Bayesian Reader is a much stronger theory than the DM. If frequency had been found to have an effect purely on the non-decision component, this would not have been a problem for the DM, it would simply be taken as a fact about the data. In contrast, such a finding would have refuted the explanation of word frequency given by the Bayesian Reader. The Bayesian Reader has to predict that frequency will appear to change drift rate when the data are fitted by the DM. If frequency were found to alter the non-decision component this would imply that there was an additional source of frequency effect outside the core of the Bayesian Reader. The same applies to the explanation of why nonword type appears to influence drift. In the DM, nonword type could potentially influence any of the model parameters, and this would not have challenged the basic principles of the model. In the Bayesian Reader, nonword type and frequency must have similar effects.

Historically, almost all studies of word recognition have used mean RT as the dependent measure. Balota and Spieler (1999) advocated going "beyond measures of central tendency" and taking advantage of the extra information that can be gained from studying the distribution of RTs. However, most studies that have analyzed the form RT distributions have relied on descriptive statistics such as higher order moments, or fitting ex-Gaussian distributions (Andrews & Heathcote, 2001; Balota & Spieler, 1999; Plourde & Besner, 1997; Yap & Balota, 2007; Yap, Balota, Cortese & Watson, 2006; Yap & Balota, Tse & Besner, 2008). None of these studies has fitted a model of word recognition to the data. The success of the simulations presented here suggests that we are now in a position to go "beyond descriptive statistics" and to use data on RT distributions to evaluate the models directly.

Frequency, priors, and contextual diversity

The function relating word frequency to RT in word recognition tasks is approximately logarithmic. However, there is an ongoing and vigorous debate as to the exact form of the function. Murray and Forster (2004, 2008) claim that RT is a function of rank frequency, as predicted by their search model. Adelman and Brown (2008a, 2008b) claim that a measure of contextual diversity gives a better account of the speed of word recognition than does either rank frequency or log frequency. Contextual diversity is a measure of how many different contexts a word appears in, and higher diversity is associated with faster RTs. Both Murray and Forster and Adelman and Brown imply that the marginally inferior fit between log frequency and RT relative to rank frequency or contextual diversity is evidence against the Bayesian Reader. However, as was made

clear in Norris (2006), the Bayesian Reader does not make predictions about word frequency itself, but about the way word recognition should be influenced by priors. In the simulations reported in that paper, the prior probability of each word was estimated by its frequency of occurrence in CELEX, but Norris pointed out that the model should really use "an estimate of the expected probability of encountering each word in the current context" (p334). Frequency can be an unreliable estimate of prior probability if the distribution of the occurrence of words is not uniform, that is, if words vary in contextual diversity. Consider the case of two words with equal raw frequencies where one always occurs in clusters of ten. The frequencies are the same but, at any point outside a cluster, the probability of encountering the clustered word is one tenth that of the uniformly distributed word. Likewise, the average wait for a bus gets longer in the rush hour when buses come in threes (Eastway, Wyndham & Rice, 1998). A Bayesian model predicts that RT should be influenced by contextual diversity, because variations in contextual diversity will alter priors.

Conclusion

Ratcliff et al fitted the DM to lexical decision data and showed that the effects of both word frequency, and of the similarity of the words and nonwords, can be captured entirely by changes in drift rate. Here I have shown that this pattern of data is exactly what would be expected from a system that performed lexical decision optimally. Simulations using the Bayesian Reader support this formal argument by confirming that the expected shifts in RT distributions do indeed follow directly from word frequency.

Despite limitations in the way the Bayesian Reader is implemented, with the exception of error RTs, it fits the data almost as well as the DM. Given the close formal relationship

between the two, this shouldn't be too surprising. Both models are based on the idea that optimal decisions are made by accumulating noisy evidence. However, whereas the noisy evidence that the DM takes as input is the output of the lexicon, the Bayesian Reader works directly on the perceptual input. The Bayesian Reader makes optimal decisions about the stimuli; the DM makes optimal decisions only about the output of the lexicon. The difference between these closely related models therefore has profound implications for the nature of lexical processing. The DM assumes that lexical processing operates before the decision process and passes on a single value representing wordness. In the Bayesian Reader, lexical and decision processes are completely integrated.

Although the fits from the DM are better than the Bayesian Reader, the Bayesian Reader captures the main features of the data very accurately. The significant advantage of the Bayesian Reader is that it models the entire process of word recognition and lexical decision. However, the most important feature of the model is not simply that it can simulate RT distributions, but that it can explain why word-frequency and nonword-type should influence the distributions in the way that they do: this is exactly how an optimally designed perceptual system should operate. The model explains why it is that word frequency and type of nonword should manifest themselves as changes in drift rate without having to invoke the concept of wordness. Furthermore, the model does all this using fewer parameters than the DM. The DM only models the decision process, and it requires input from an unspecified lexical model before it can give a full explanation of the data. The problem with the DM analysis presented by Ratcliff et al. is simply that it didn't go far enough – the underlying principles can be extended to give a far more complete explanation of the data. Diffusion models can be much more than just a way of

parameterizing RT distributions, and can be developed into complete models of perceptual processing. There is no need for a distinction between a lexical system and a decision process: There is just one integrated system which can be characterized as an optimal Bayesian diffusion process. The lexical decision task, and, indeed, Ratcliff et al.'s own data, gives us important insights into fundamental properties of how words are recognized. When used carefully in conjunction with quantitative modeling, the lexical decision task can continue to be a valuable source of information about visual word recognition.

References

Adelman, J. S., & Brown, G. D. (2008). Modeling lexical decision: the form of frequency and diversity effects. *Psychological Review, 115*(1), 214-229.

Adelman, J. S., Brown, G. D., & Quesada, J. F. (2006). Contextual diversity, not word frequency, determines word-naming and lexical decision times. *Psychological Science, 17*(9), 814-823.

Andrews, S. (1997). The effect of orthographic similarity on lexical retrieval: Resolving neighborhood conflicts. *Psychonomic Bulletin & Review, 4*(439-461).

Andrews, S., & Heathcote, A. (2001). Distinguishing common and task-specific processes in word identification: A matter of some moment? *Journal of Experimental Psychology: Human Perception and Performance, 27*(2), 514-544.

Baayen, R. H., Piepenbrock, R., & Gulikers, L. (1995). *The CELEX Lexical Database (Release 2) CDROM*. Philadelphia, PA: Linguistic Data Consortium, University of Pennsylvania.

Balota, D. A., & Chumbley, J. I. (1984). Are lexical decisions a good measure of lexical access? The role of word frequency in the neglected decision stage. *Journal of Experimental Psychology: Human Perception and Performance, 10*(3), 340-357.

Balota, D. A., Cortese, M. J., Sergent-Marshall, S. D., Spieler, D. H., & Yap, M. (2004). Visual word recognition of single-syllable words. *Journal of Experimental Psychology: General, 133*(2), 283-316.

Balota, D. A., & Spieler, D. H. (1998). The utility of item-level analyses in model evaluation: A reply to Seidenberg and Plaut. *Psychological Science, 9*, 238-240.

Balota, D. A., & Spieler, D. H. (1999). Word frequency, repetition, and lexicality effects in word recognition tasks: Beyond measures of central tendency. *Journal of Experimental Psychology: General, 128*(1), 32-55.

Baum, C. W., & Veeravalli, V. (1994). A sequential procedure for multihypothesis testing. *IEEE Transactions on Information Theory, 40*(6), 1994-2007.

Bogacz, R. (2007). Optimal decision-making theories: linking neurobiology with behaviour. *Trends in Cognitive Science, 11*(3), 118-125.

Bogacz, R., Brown, E., Moehlis, J., Holmes, P., & Cohen, J. D. (2006). The physics of optimal decision making: a formal analysis of models of performance in two-alternative forced-choice tasks. *Psychological Review, 113*(4), 700-765.

Carpenter, R. H. S. (1981). Oculomotor Procrastination. In D. F. Fisher, R. A. Monty & J. W. Senders (Eds.), *Eye Movements: Cognition and Visual Perception* (pp. 237-246). Hillsdale: Lawrence Erlbaum.

Carpenter, R. H. S. (1999). A neural mechanism that randomises behaviour. *Journal of Consciousness Studies, 6*(1), 13-22.

Chaffin, R., Morris, R. K., & Seely, R. E. (2001). Learning new word meanings from context: A study of eye movements. *Journal of Experimental Psychology: Learning Memory and Cognition, 27*(1), 225-235.

Coltheart, M., Rastle, K., Perry, C., Langdon, R., & Ziegler, J. (2001). DRC: A dual route cascaded model of visual word recognition and reading aloud. *Psychological Review, 108*(1), 204-256.

Dragalin, V., Tartakovsky, A., & Veeravalli, V. (1999). Multihypothesis sequential probability ratio tests, part I: Asymptotoc optimality. *IEE Transactions on Information Theory, 45*, 2448-2461.

Dragalin, V., Tartakovsky, A., & Veeravalli, V. (2000). Multihypothesis sequential probability ratio tests, part 2: Accurate asymptotic expansions for the expected sample size. *IEE Transactions on Information Theory, 46*, 1366-1383.

Eastway, R., Wyndham, J., & Rice, T. (1998). *Why do buses come in threes?* London, UK: Robson Books.

Grainger, J., & Jacobs, A. M. (1996). Orthographic processing in visual word recognition: a multiple read-out model. *Psychological Review, 103*(3), 518-565.

Gray, A., & Kolda, T. G. (2006). Asynchronous parallel pattern search for derivative-free optimization. . *ACM Transactions on Mathematical Software,, 32*(3), 485-507.

Kolda, T. G. (2005). Revisiting asynchronous parallel pattern search for nonlinear optimization. *SIAM Journal on Optimization, 16*(2), 563-586.

Kucera, H., & Francis, W. N. (1967). *Computational analysis of present-day American English*. Providence: Brown University Press.

Laming, D. R. J. (1968). *Information theory of choice-reaction times.* New York: Wiley.

Marr, D. (1982). *Vision: A computational investigation into the human representation and processing of visual information*. San Francisco: Freeman & Co.

Murray, W. S. (2001, November). *The rank hypothesis: Evidence from reaction times, error rates and eye movements.* Paper presented at the 42nd Annual Meeting of the Psychonomic Society, Orlando, FL.

Murray, W. S. (2007, August). *Lexical access reflected in the eye movement record.* Paper presented at the 14th European Conference on Eye Movements, Potsdam, Germany.

Murray, W. S., & Forster, K. I. (2004). Serial mechanisms in lexical access: The rank hypothesis. *Psychological Review, 111*(3), 721-756.

Murray, W. S., & Forster, K. I. (2008). The rank hypothesis and lexical decision: a reply to Adelman and Brown (2008). *Psychological Review, 115*(1), 240-252.

Nelder, J. A., & Mead, R. (1965). A Simplex Method for Function Minimization. *The Computer Journal, 7*, 308-313.

Norris, D. (2006). The Bayesian Reader: explaining word recognition as an optimal Bayesian decision process. *Psychological Review, 113*(2), 327-357.

Norris, D., & McQueen, J. M. (2008). Shortlist B: a Bayesian model of continuous speech recognition. *Psychological Review, 115*(2), 357-395.

Norris, D., & S., K. (2008). Perception as evidence accumulation and Bayesian inference: Insights from masked priming. *Journal of Experimental Psychology: General, 137*(3), 434-455.

Pelli, D. G., Farell, B., & Moore, D. C. (2003). The remarkable inefficiency of word recognition. *Nature, 423*(6941), 752-756.

Perea, M., & Rosa, E. (2000). The effects of orthographic neighborhood in reading and laboratory identification tasks: A review. *Psicologica, 21*, 237-340.

Plourde, C. E., & Besner, D. (1997). On the locus of the word frequency effect in visual word recognition. *Canadian Journal of Experimental Psychology, 51*, 181–194.

Ratcliff, R. (1978). A theory of memory retrieval. *Psychological Review, 85*, 59-109.

Ratcliff, R., Gomez, P., & McKoon, G. (2004). A diffusion model account of the lexical decision task. *Psychological Review, 111*(1), 159-182.

Ratcliff, R., & McKoon, G. (2008). The diffusion decision model: theory and data for two-choice decision tasks. *Neural Computation, 20*(4), 873-922.

Schilling, H. H., Rayner, K., & Chumbley, J. I. (1998). Comparing naming, lexical decision, and eye fixation times: word frequency effects and individual differences. *Memory & Cognition, 26*(6), 1270-1281.

Spieler, D. H., & Balota, D. A. (1997). Bringing computational models of word naming down to the item level. *Psychological Science, 8*, 411-416.

Spieler, D. H., & Balota, D. A. (2000). Factors influencing word naming in younger and older adults. *Psychology of Aging, 15*(2), 225-231.

Stone, M. (1960). Models for choice reaction time. *Psychometrika, 25*, 251–260.

van Casteren, M., & Davis, M. H. (2007). Match: a program to assist in matching the conditions of factorial experiments. *Behavior Research Methods, 39*(4), 973-978.

Veeravalli, V., & Baum, C. W. (1995). Asymptotic efficiency of a sequential multihypothesis test. *IEE Transactions on Information Theory, 41*(6), 1994-1997.

Wagenmakers, E. J., Ratcliff, R., Gomez, P., & McKoon, G. (2008). A diffusion model account of criterion shifts in the lexical decision task. *Journal of Memory and Language, 58*(1), 140-159.

Wagenmakers, E. J., Steyvers, M., Raaijmakers, J. G., Shiffrin, R. M., van Rijn, H., & Zeelenberg, R. (2004). A model for evidence accumulation in the lexical decision task. *Cognitive Psychology, 48*(3), 332-367.

Wald, A. (1947). *Sequential analysis*. New York: Wiley.

Whaley, C. P. (1978). Word-nonword classification time. *Journal of Verbal Learning and Verbal Behavior, 17*(2), 143-154.

Yap, M. J., & Balota, D. A. (2007). Additive and interactive effects on response time distributions in visual word recognition. *Journal of Experimental Psychology: Learning Memory and Cognition, 33*(2), 274-296.

Yap, M. J., Balota, D. A., Cortese, M. J., & Watson, J. M. (2006). Single- versus dual-process models of lexical decision performance: insights from response time distributional analysis. *Journal of Experimental Psychology: Human Perception and Performance, 32*(6), 1324-1344.

Yap, M. J., Balota, D. A., Tse, C. S., & Besner, D. (2008). On the additive effects of stimulus quality and word frequency in lexical decision: Evidence for opposing interactive influences revealed by RT distributional analyses. *Journal of Experimental Psychology: Learning Memory and Cognition, 34*(3), 495-513.

Footnote

1. The version of the Bayesian Reader program used for running these simulations is available here:

http://www.mrc-cbu.cam.ac.uk/people/dennis.norris/personal/BayesianReader

Acknowledgements

Thanks to Maarten van Casteren for programming the Bayesian Reader, Ian Nimmo-Smith for valuable advice, the reviewers for their valuable suggestions for improving the manuscript.

Appendix. Lexical decision in the Bayesian Reader.

Representation of letters and words

In the simulations reported here, words are represented by a concatenation of position-specific letter vectors. Each letter is represented by a 26 element binary vector where one coordinate is set to 1 and the remainder to 0. Each coordinate can be thought of representing a letter feature. Five-letter words are therefore represented by a 130 dimensional vector. For simplicity, we assume that features will always be either 0 or 1, although this is not necessary. The calculations below start by computing the likelihoods that each feature has the values 0 and 1. Letter probabilities can then be computed from the products of the feature likelihoods in each letter, and word probabilities from the probabilities of the letters in the word.

Sampling

Each sample presented to the model is derived by adding an independent sample of zero-mean Gaussian noise with standard deviation σ to each coordinate of the input vector. For each coordinate separately, calculate the mean value $\bar{x}$ of the samples received so far (x_1-x_t), and the standard error of the mean of those samples, σ_M. This is analogous to the situation depicted in Figure 2, where $W_1 = 1$, $W_2 = 0$, (the features are all either 1 or 0) and $\mu = \bar{x}$, and σ_M provides a measure of the dispersion of the distributions.

σ_M is calculated in the usual way

$$\sigma_M = \sigma / \sqrt{t} \qquad (A1)$$

where *t* is the number of time-steps/samples, and σ is given by :

$$\sigma^2 = \sum_{i=1}^{i=t} (\bar{x} - x_i)^2 \Big/ t - 1 \qquad (A2)$$

Then the likelihood of each feature (height of the likelihood function) having a given state is given by

$$LF_j = f(F_j) = \frac{e^{d_j^2 / 2\sigma_M^2}}{\sigma_M \sqrt{2\pi}}. \qquad (A3)$$

Where d_j is the distance between the mean and the value of each feature state (0 or 1):

$$d_j = \bar{x} - F_j \qquad (A4)$$

The probability of each letter is given by the product of the likelihood that each feature within that letter takes the correct feature value, divided by the sum of those products for all letters.

$$P(L_x) = P(FS_x) \Big/ \sum_{k=1}^{k=26} P(FS_k) \qquad (A5)$$

Where FS_x is the product of the likelihoods of each features in $letter_x$ having the correct value.

$$FS_x = \prod_{y=1}^{y=26} LF_{xy} \qquad (A6)$$

An analogous calculation computes word probabilities from the products of the letters within those words. Word probabilities are calculated using Bayes' theorem,

$$P(W_x) = P(W_x) \times P(LS_x) \Big/ \sum_{k=1}^{k=m} (P(W_k) \times P(LS_k)) \qquad (A7)$$

where m is the number of words in the lexicon, $P(W_k)$ is the probability of the k_{th} word in the lexicon, and LS_w is product of the probabilities of the sequence of letters in the that word. Note that because σ_M can fluctuate wildly during the first few samples, probabilities are only calculated after the model has received 10 samples.

Computing the probability that the input is a word

In the original implementation of the model, the procedure for calculating the likelihood that the input was a nonword involved two terms. The first was the likelihood that the input was produced by a *virtual-nonword* which was located as close as possible to the mean of the input samples, but was as least some minimum distance from the nearest word. In the simulations reported in Norris (2006), that distance always corresponded to the distance between two letter-strings differing by a single letter. An additional *background-nonword* term was incorporated which made allowance for the possibility that nonwords could be located anywhere in lexical space. That is, nonwords should be considered to be selected from some large set of nonwords that could potentially appear in the experiment. The combination of the *virtual-nonword* and the *background-nonwords* is therefore designed to capture the fact that, in the lexical

decision task, nonwords can appear anywhere in lexical space, but are generally similar to words.

Because the new implementation is based on letter probabilities there is no straightforward representation of distance in lexical space, and likelihoods can only readily be computed for specific letter-strings. For simplicity and speed of computation, the present simulations therefore take advantage of the fact that the sum of letter probabilities at each position, and the sum of the probabilities of all possible letter-strings, must both necessarily be 1.0. If a particular string of letters has been identified, each with P(letter|input) = 1.0, and these letters form a word, say WORK, then the input must be a word. If the final letter in the string is ambiguous between K and D (P(K|input) = 0.5, P(D|input) = 0.5) then the input must be a word because the sum of the letter-string probabilities corresponding to words is 1.0. In contrast, if the input is WORB and final letter is completely inconsistent with any word, (P(B|input) = 1.0), then the input must be a nonword. The sum of the letter-string probabilities corresponding to words could therefore be used to compute the likelihood that the input is a word because the summed probability of the letter-strings corresponding to nonwords is 1-summed probability of letter-strings corresponding to words.

The sum of the word and nonword priors is each set to 0.5, as the target is equally likely to be a word as a nonword. The nonword likelihood is then given by

$$NL = 1.0 - \sum_{i=1}^{i=m} P(LS_i) \qquad (A8)$$

The likelihood that the input is a word is given by

$$WL = \sum_{i=1}^{i=m} (P(W_i) \times P(LS_i)) \qquad (A9)$$

(where $P(W_i)$ is now the prior adjusted for the fact that the sum of all word priors must be 0.5) and the probability that the input is a word is given by

$$P(a\,word) = WL/(WL + NL) \qquad (A10)$$

Table 1 Bayesian Reader simulation of Ratcliff, Gomez & McKoon, Experiment 1

Quantile	.1	.3	.5	.7	.9	% correct	Correct RT	Error RT
Pseudowords								
Data	492	556	613	691	884	.928	661	698
Bayesian Reader	479	541	589	665	865	.929	640	578
Ratcliff et al.	484	550	608	690	869	.889	650	668
High-frequency words								
Data	453	501	542	591	710	.971	571	636
Bayesian Reader	470	521	561	606	709	.974	580	516
Ratcliff et al.	458	512	554	602	701	.984	570	593
Low-frequency words								
Data	487	547	597	670	841	.910	639	682
Bayesian Reader	484	545	596	671	836	.912	626	529
Ratcliff et al.	479	544	601	681	858	.910	642	674
Very low-frequency words								
Data	497	565	632	717	912	.804	679	686
Bayesian Reader	490	561	628	731	938	.857	682	545
Ratcliff et al.	489	561	629	729	950	.785	683	711

Table 1 shows data from Ratcliff, Gomez & McKoon's Experiment 1, along with simulated data from the Bayesian Reader and Ratcliff, Gomez & McKoon's diffusion model (Ratcliff et al.). The stimuli were pronounceable pseudowords, high-frequency words, low-frequency words and very low-frequency words.

Table 2 Bayesian Reader simulation of Ratcliff, Gomez & McKoon , Experiment 2

Quantile	.1	.3	.5	.7	.9	% correct	Correct RT	Error RT
Random letter strings								
Data	432	489	536	597	745	.956	575	591
Bayesian Reader	436	500	549	603	763	.957	576	486
Ratcliff et al.	432	489	538	604	749	.957	582	606
High-frequency words								
Data	433	482	526	575	684	.968	549	497
Bayesian Reader	436	488	532	572	669	.961	547	509
Ratcliff et al.	434	486	528	580	683	.973	556	514
Low-frequency words								
Data	446	502	552	616	762	.951	589	590
Bayesian Reader	443	503	551	608	764	.927	585	529
Ratcliff et al.	443	501	551	618	761	.943	591	566
Very low-frequency words								
Data	451	511	566	640	817	.931	609	625
Bayesian Reader	446	509	561	632	825	.885	610	553
Ratcliff et al.	448	510	566	642	811	.915	614	599

Table 2 shows data from Ratcliff, Gomez & McKoon's Experiment 2, along with simulated data from the Bayesian Reader, and Ratcliff, Gomez & McKoon's diffusion model (Ratcliff et al.). The stimuli were random letter strings, high-frequency words, low-frequency words and very low-frequency words.

Table 3. Parameter values for fits of the Bayesian Reader model

	Yes	No	SDSD	Bias Noise	Non-decision	Start noise	Slope
E1	0.900	0.01	2.088	0.900	421	156	0.929
E2	0.906	0.01	3.025	0.625	381	180	0.981

Table 3 shows the parameter values for the fits of the Bayesian Reader model to Ratcliff et al.'s experiments 1 and 2.

Note. Yes = yes threshold, No = no threshold, SDSD standard deviation of the trial-by-trial change in the sampling noise, Bias Noise = range of the initial probability that the input is a word, Non-decision = non-decision time, Start-noise = noise in starting time, Slope = slope relating RT to samples. Note that with the current lexical decision procedure there is no longer a close correspondence between Yes/No threshold values and the probability of correct responding.

Figure captions

Figure 1.

The diffusion model, showing three simulated paths with drift rate v, boundary separation a, and starting point z. (reproduced from Ratcliff, R., & McKoon, G., 2007). The diffusion decision model: Theory and data for two-choice decision tasks. *Neural Computation*, 2008.)

Figure 2.

Illustration of word recognition in a lexicon of two words which differ along a single perceptual dimension. W1 is the value of word 1, W2 is the value of word 2, μ is the mean of the samples received at this point, and the two density functions are identical and are determined by the sampling distribution of the mean (σ_M).

Figure 1

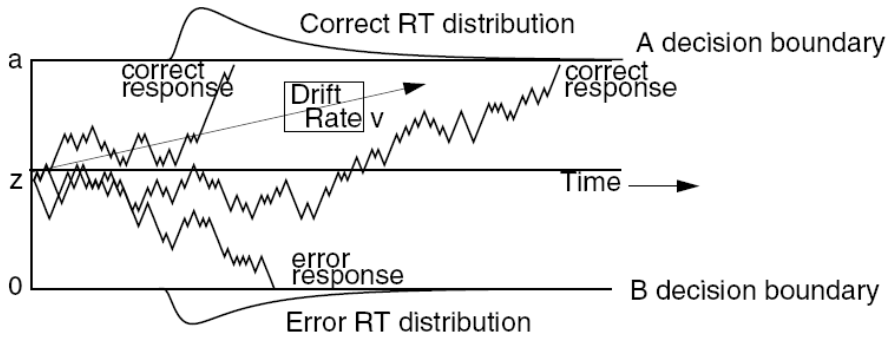
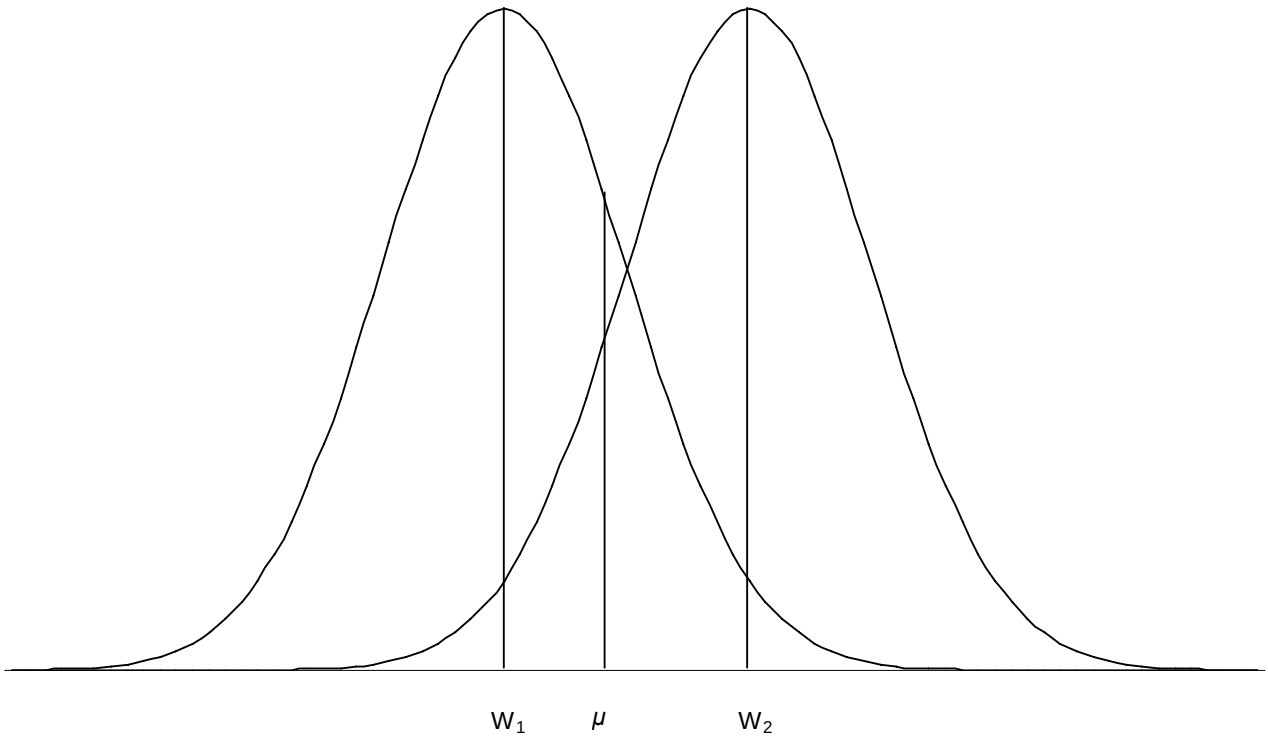

Correct RT distribution
A decision boundary
a
correct
response
Drift
Rate v
correct
response
z
Time
error
response
0
B decision boundary
Error RT distribution

Figure 2



Footnote:

In the Bayesian Reader simulations the standard error of the mean of the samples was calculated from the input samples. The simulations were also run by giving the model knowledge of the actual sampling standard deviation on each trial, but this did not improve the fits to the error RTs.