

Psychological Review

VOLUME 88 NUMBER 5 SEPTEMBER 1981

An Interactive Activation Model of Context Effects in Letter Perception: Part 1. An Account of Basic Findings

James L. McClelland and David E. Rumelhart
University of California, San Diego

A model of context effects in perception is applied to the perception of letters in various contexts. In the model, perception results from excitatory and inhibitory interactions of detectors for visual features, letters, and words. A visual input excites detectors for visual features in the display. These excite detectors for letters consistent with the active features. The letter detectors in turn excite detectors for consistent words. Active word detectors mutually inhibit each other and send feedback to the letter level, strengthening activation and hence perceptibility of their constituent letters. Computer simulation of the model exhibits the perceptual advantage for letters in words over unrelated contexts and is consistent with the basic facts about the word advantage. Most importantly, the model produces facilitation for letters in pronounceable pseudowords as well as words. Pseudowords activate detectors for words that are consistent with most of the active letters, and feedback from the activated words strengthens the activations of the letters in the pseudoword. The model thus accounts for apparently rule-governed performance without any actual rules.

As we perceive, we are continually extracting sensory information to guide our attempts to determine what is before us. In addition, we bring to perception a wealth of knowledge about the objects we might see or hear and the larger units in which these objects co-occur. As one of us has argued for the case of reading (Rumelhart, 1977), our knowledge of the objects we might be perceiving works together with the sensory information in the perceptual process. Exactly how does the knowledge that we have inter-

act with the input? And how does this interaction facilitate perception?

In this two-part article we have attempted to take a few steps toward answering these questions. We consider one specific example of the interaction of knowledge and perception—the perception of letters in words and other contexts. In Part 1 we examine the main findings in the literature on perception of letters in context and develop a model called the interactive activation model to account for these effects. In Part 2 (Rumelhart & McClelland, in press) we extend the model in several ways. We present a set of studies introducing a new technique for studying the perception of letters in context, independently varying the duration and timing of the context and target letters. We show how the model fares in accounting for the results of these experiments and discuss how the model may be extended to account for a variety of phenomena. We also present an experiment that tests—and supports—a

Preparation of this article was supported by National Science Foundation Grants BNS-76-14830 and BNS-79-24062 to J. L. McClelland and Grant BNS-76-15024 to D. E. Rumelhart, and by the Office of Naval Research under contract N00014-79-C-0323. We would like to thank Don Norman, James Johnston, and members of the LNR research group for helpful discussions of much of the material covered in this article.

Requests for reprints may be sent to James L. McClelland or David E. Rumelhart at Department of Psychology, C-009, University of California, San Diego, La Jolla, California 92093.

counterintuitive prediction of the model. Finally, we consider how the mechanisms developed in the course of exploring our model of word perception might be extended to perception of other sorts of stimuli.

Basic Findings on the Role of Context in Perception of Letters

The notion that knowledge and familiarity play a role in perception has often been supported by experiments on the perception of letters in words (Bruner, 1957; Neisser, 1967). It has been known for nearly 100 years that it is possible to identify letters in words more accurately than letters in random letter sequences under tachistoscopic presentation conditions (Cattell, 1886; see Huey, 1908, and Neisser, 1967, for reviews). However, until recently such effects were obtained using whole reports of all of the letters presented. These reports are subject to guessing biases, so that it was possible to imagine that familiarity did not determine how much was seen but only how much could be inferred from a fragmentary percept. In addition, for longer stimuli, full reports are subject to forgetting. We may see more letters than we can actually report in the case of nonwords, but when the letters form a word, we may be able to retain as a single unit the item whose spelling may simply be read out from long-term memory. Thus, despite strong arguments to the contrary by proponents of the view that familiar context really does influence perception, it has been possible until recently to imagine that the context in which a letter was presented influences only the accuracy of postperceptual processes and not the process of perception itself.

The perceptual advantage of letters in words. The seminal experiment of Reicher (1969) suggests that context does actually influence perceptual processing. Reicher presented target letters in words, unpronounceable nonwords, and alone, following the presentation of the target display with a presentation of a patterned mask. The subject was then tested on a single letter in the display, using a forced choice between two alternative letters. Both alternatives fit the context to form an item of the type pre-

sented, so that, for example in the case of a word presentation, the alternative would also form a word in the context.

Forced-choice performance was more accurate for letters in words than for letters in nonwords or even for single letters. Since both alternatives made a word with the context, it is not possible to argue that the effect is due to postperceptual guessing based on equivalent information extracted about the target letter in the different conditions. It appears that subjects actually come away with more information relevant to a choice between the alternatives when the target letter is a part of a word. And, since one of the control conditions was a single letter, it is not reasonable to argue that the effect is due to forgetting letters that have been perceived. It is hard to see how a single letter, once perceived, could be subject to a greater forgetting than a letter in a word.

Reicher's (1969) finding seems to suggest that perception of a letter can be facilitated by presenting it in the context of a word. It appears, then, that our knowledge about words can influence the process of perception. Our model presents a way of bringing such knowledge to bear. The basic idea is that the presentation of a string of letters begins the process of activating detectors for letters that are consistent with the visual input. As these activations grow stronger, they begin to activate detectors for words that are consistent with the letters, if there are any. The active word detectors then produce feedback, which reinforces the activations of the detectors for the letters in the word. Letters in words are more perceptible, because they receive more activation than representations of either single letters or letters in an unrelated context.

Reicher's basic finding has been investigated and extended in a large number of studies, and there now appears to be a set of important related findings that must also be explained.

Irrelevance of word shape. The effect seems to be independent of the familiarity of the word as a visual configuration. The word advantage over nonwords is obtained for words in lowercase type, words in uppercase type, or words in a mixture of upper-

and lowercase (Adams, 1979; McClelland, 1976).

Role of patterned masking. The word advantage over single letters and nonwords appears to depend upon the visual masking conditions used (Johnston & McClelland, 1973; Massaro & Klitzke, 1979; see also Juola, Leavitt, & Choe, 1974; Taylor & Chabot, 1978). The word advantage is quite large when the target appears in a distinct, high-contrast display followed by a patterned mask of similar characteristics. However, the word advantage over single letters is actually reversed, and the word advantage over nonwords becomes quite small when the target is indistinct, low in contrast, and/or followed by a blank, nonpatterned field.

Extension to pronounceable pseudo-words. The word advantage also applies to pronounceable nonwords, such as *REET* or *MAVE*. A large number of studies (e.g., Aderman & Smith, 1971; Baron & Thurston, 1973; Spoehr & Smith, 1975) have shown that letters in pronounceable nonwords (also called pseudowords) have a large advantage over letters in unpronounceable nonwords (also called unrelated letter strings), and three studies (Carr, Davidson, & Hawkins, 1978; Massaro & Klitzke, 1979; McClelland & Johnston, 1977) have obtained an advantage for letters in pseudowords over single letters.

Absence of effects of contextual constraint under patterned-mask conditions. One important finding, which rules out several of the models that have been proposed previously, is the finding that letters in highly constraining word contexts have little or no advantage over letters in weakly constraining word contexts under the distinct-target/patterned-mask conditions that produce a large word advantage (Johnston, 1978; see also Estes, 1975). For example, if the set of possible stimuli contains only words, the context *_HIP* constrains the first letter to be either an *S*, a *C*, or a *W*; whereas the context *_INK* is compatible with 12 to 14 letters (the exact number depends on what counts as a word). We might expect that the former, more strongly constraining context would produce superior detection of a target letter. But in a very carefully controlled and executed study, Johnston (1978)

found no such effect. Although constraints do influence performance under other conditions (e.g., Broadbent & Gregory, 1968), they do not appear to make a difference under the distinct-target/patterned-mask conditions of the Johnston study.

To be successful, any model of word perception must provide an account not only for Reicher's (1969) basic effect but for these related findings as well. Our model accounts for all of these effects. We begin by presenting the model in abstract form. We then focus on the specific version of the model implemented in our simulation program and consider some of the details. Subsequently, we turn to detailed considerations of the findings we have discussed in this section.

The Interactive Activation Model

We approach the phenomena of word perception with a number of basic assumptions that we want to incorporate into the model. First, we assume that perceptual processing takes place within a system in which there are several levels of processing, each concerned with forming a representation of the input at a different level of abstraction. For visual word perception, we assume that there is a visual feature level, a letter level, and a word level, as well as higher levels of processing that provide "top-down" input to the word level.

Second, we assume that visual perception involves parallel processing. There are two different senses in which we view perception as parallel. We assume that visual perception is spatially parallel. That is, we assume that information covering a region in space at least large enough to contain a four-letter word is processed simultaneously. In addition, we assume that visual processing occurs at several levels at the same time. Thus, our model of word perception is spatially parallel (i.e., capable of processing several letters of a word at one time) and involves processes that operate simultaneously at several different levels. Thus, for example, processing at the letter level presumably occurs simultaneously with processing at the word level and with processing at the feature level.

Third, we assume that perception is fundamentally an *interactive* process. That is,

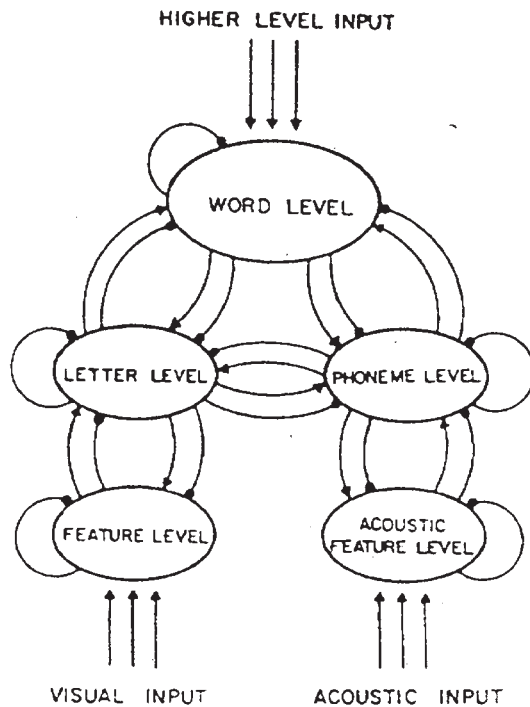


Figure 1. A sketch of some of the processing levels involved in visual and auditory word perception, with interconnections.

we assume that "top-down" or "conceptually driven" processing works simultaneously and in conjunction with "bottom-up" or "data driven" processing to provide a sort of multiplicity of constraints that jointly determine what we perceive. Thus, for example, we assume that knowledge about the words of the language interacts with the incoming featural information in codetermining the nature and time course of the perception of the letters in the word.

Finally, we wish to implement these assumptions by using a relatively simple method of interaction between sources of knowledge whose only "currency" is simple excitatory and inhibitory activations of a neural type.

Figure 1 shows the general conception of the model. Perceptual processing is assumed to occur in a set of interacting levels, each communicating with several others. Communication proceeds through a spreading activation mechanism in which activation at one level spreads to neighboring levels. The communication can consist of both excit-

atory and inhibitory messages. Excitatory messages increase the activation level of their recipients. Inhibitory messages decrease the activation level of their recipients. The arrows in the diagram represent excitatory connections, and the circular ends of the connections represent inhibitory connections. The intralevel inhibitory loop represents a kind of lateral inhibition in which incompatible units at the same level compete. For example, since a string of four letters can be interpreted as at most one four-letter word, the various possible words mutually inhibit one another and in that way compete as possible interpretations of the string.

It is clear that many levels are important in reading and perception in general, and the interactions among these levels are important for many phenomena. However, a theoretical analysis of all of these interactions introduces an order of complexity that obscures comprehension. For this reason, we have restricted the present analysis to an examination of the interaction between a single pair of levels, the word and letter levels. We have found that we can account for the phenomena reviewed above by considering only the interactions between letter level and word level elements. Therefore, for the present we have elaborated the model only on these two levels, as illustrated in Figure 2. We have delayed consideration of the effects of higher level processes and phonological processes, and we have ignored the reciprocity of activation that may occur between word and letter levels and any other levels of the system. We consider aspects of the fuller model including these influences in Part 2 (Rumelhart & McClelland, in press).

Specific Assumptions

Representation assumptions. For every relevant unit in the system we assume there is an entity called a *node*. We assume that there is a node for each word we know, and that there is a node for each letter in each letter position within a four-letter string.

The nodes are organized into levels. There are *word level* nodes and *letter level* nodes. Each node has connections to a number of

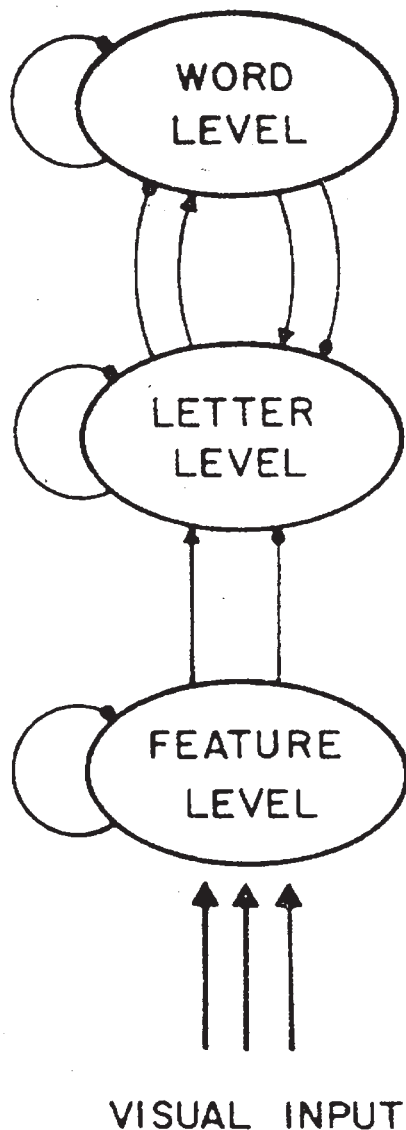


Figure 2. The simplified processing system.

other nodes. The nodes to which a node connects are called its *neighbors*. Each connection is two-way. There are two kinds of connections: *excitatory* and *inhibitory*. If two nodes suggest each other's existence (in the way that the node for the word *the* suggests the node for an initial *t* and vice versa), then the connections are excitatory. If two nodes are inconsistent with one another (in the way that the node for the word *the* and the node for the word *boy* are inconsistent), then the relationship is inhibitory. Note that we iden-

tify nodes according to the units they detect, printing them in italics; stimuli presented to the system are in uppercase letters.

Connections may occur within levels or between adjacent levels. There are no connections between nonadjacent levels. Connections within the word level are mutually inhibitory, since only one word can occur at any one place at any one time. Connections between the word level and letter level may be either inhibitory or excitatory (depending on whether the letter is a part of the word in the appropriate letter position). We call the set of nodes with excitatory connections to a given node its *excitatory neighbors* and the set of nodes with inhibitory connections to a given node its *inhibitory neighbors*.

A subset of the neighbors of the letter *t* is illustrated in Figure 3. Again, excitatory connections are represented by the arrows ending with points, and inhibitory connections are represented by the arrows ending with dots. We emphasize that this is a small subset of the neighborhood of the initial *t*. The picture of the whole neighborhood, including all the connections among neighbors and their connections to their neighbors, is much too complicated to present in a two-dimensional figure.

Activation assumptions. There is associated with each node a momentary activation value. This value is a real number, and for node *i* we will represent it by $a_i(t)$. Any node with a positive activation value is said to be *active*. In the absence of inputs from its neighbors, all nodes are assumed to decay back to an inactive state, that is, to an activation value at or below zero. This resting level may differ from node to node and corresponds to a kind of a priori bias (Broadbent, 1967) determined by frequency of activation of the node over the long term. Thus, for example, the nodes for high-frequency words have resting levels higher than those for low-frequency words. In any case, the resting level for node *i* is represented by r_i . For units not at rest, decay back to the resting level occurs at some rate θ_i .

When the neighbors of a node are active, they influence the activation of the node by either excitation or inhibition, depending on their relation to the node. These excitatory and inhibitory influences combine by a sim-

ple weighted average to yield a net input to the unit, which may be either excitatory (greater than zero) or inhibitory. In mathematical notation, if we let $n_i(t)$ represent the net input to the unit, we can write the equation for its value as

$$n_i(t) = \sum_j \alpha_{ij} e_j(t) - \sum_k \gamma_{ik} i_k(t), \quad (1)$$

where $e_j(t)$ is the activation of an active excitatory neighbor of the node, each $i_k(t)$ is the activation of an active inhibitory neighbor of the node, and α_{ij} and γ_{ik} are associated weight constants. Inactive nodes have no influence on their neighbors. Only nodes in an active state have any effects, either excitatory or inhibitory.

The net input to a node drives the activation of the node up or down, depending on whether it is positive or negative. The degree of the effect of the input on the node is modulated by the node's current activity level to keep the input to the node from driving it beyond some maximum and minimum values (Grossberg, 1978). When the net input is excitatory, $n_i(t) > 0$, the effect on the node, $\epsilon_i(t)$, is given by

$$\epsilon_i(t) = n_i(t)(M - a_i(t)), \quad (2)$$

where M is the maximum activation level of the unit. The modulation has the desired effect, because as the activation of the unit approaches the maximum, the effect of the input is reduced to zero. M can be thought

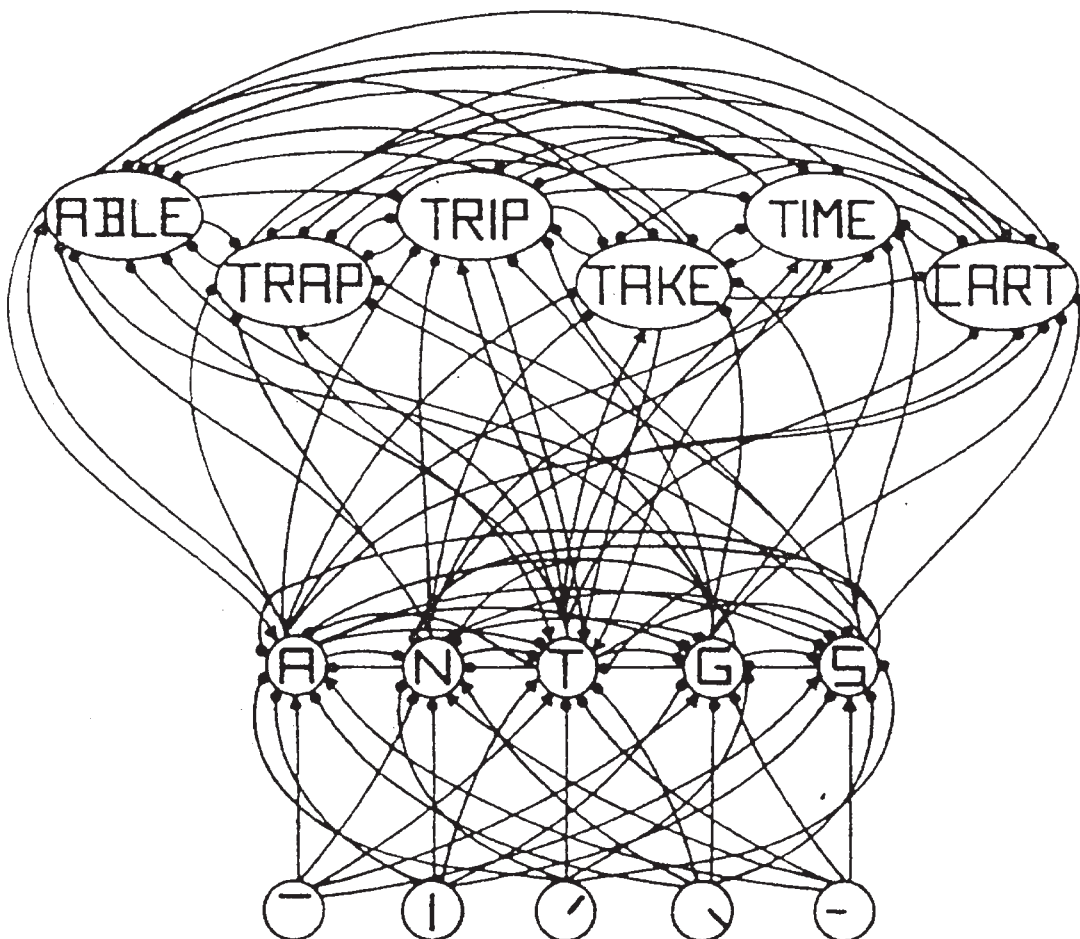


Figure 3. A few of the neighbors of the node for the letter *T* in the first position in a word, and their interconnections.

of as a basic scale factor of the model, and we have set its value to 1.0.

In the case where the input is inhibitory, $n_i(t) < 0$, the effect of the input on the node is given by

$$\epsilon_i(t) = n_i(t)(a_i(t) - m), \quad (3)$$

where m is the minimum activation of the unit.

The new value of the activation of a node at time $t + \Delta t$ is equal to the value at time t , minus the decay, plus the influence of its neighbors at time t :

$$a_i(t + \Delta t) = a_i(t) - \Theta_i(a_i(t) - r_i) + \epsilon_i(t). \quad (4)$$

Input assumptions. Upon presentation of a stimulus, a set of featural inputs is made available to the system. Each feature in the display will be detected with some probability p . For simplicity it is assumed that feature detection occurs, if it is to occur at all, immediately after onset of the stimulus. The probability that any given feature will be detected is assumed to vary with the visual quality of the display. Features that are detected begin sending activation to all letter nodes that contain that feature. All letter level nodes that do not contain the extracted feature are inhibited.

It is assumed that features are binary and that we can extract either the presence or absence of a particular feature. So, for example, when viewing the letter *R* we can extract, among other features, the presence of a diagonal line segment in the lower right corner and the absence of a horizontal line across the bottom. In this way the model honors the conceptual distinction between knowing that a feature is absent and not knowing whether a feature is present.

Presentation of a new display following an old one results in the probabilistic extraction of the set of features present in the new display. These features, when extracted, replace the old ones in corresponding positions. Thus, the presentation of an *E* following the *R* described above would result in the replacement of the two features described above with their opposites.

On making responses. One of the more

problematic aspects of a model such as this one is a specification of how these relatively complex patterns of activity might be related to the content of percepts and the sorts of response probabilities we observe in experiments. We assume that responses and perhaps the contents of perceptual experience depend on the temporal integration of the pattern of activation over all of the nodes. The integration process is assumed to occur slowly enough that brief activations may come and go without necessarily becoming accessible for purposes of responding or entering perceptual experience. However, as the activation lasts longer and longer, the probability that it will be reportable increases. Specifically, we think of the integration process as taking a running average of the activation of the node over previous time:

$$\bar{a}_i(t) = \int_{-\infty}^t a_i(x) e^{-\mu(t-x)r} dx. \quad (5)$$

In this equation, the variable x represents preceding time, varying between $-\infty$ and time t . The exponential portion of the expression weights the contribution of the activation of the node in previous time intervals: Essentially, its effect is to reduce the contribution of prior activations as they recede further back in time. The parameter r represents the relative weighting given to old and new information and determines how quickly the output values change in response to changes in the activations of the underlying nodes. The larger the value of r , the more quickly the output values change. *Response strength*, in the sense of Luce's choice model (Luce, 1959), is an exponential function of the running average activation:

$$s_i(t) = e^{\mu \bar{a}_i(t)}. \quad (6)$$

The parameter μ determines how rapidly response strength grows with increases in activation. Following Luce's formulation, we assume that the probability of making a response based on node i is given by

$$p(R_i, t) = \frac{s_i(t)}{\sum_{j \in L} s_j(t)}, \quad (7)$$

where L represents the set of nodes competing at the same level with node i .

Most of the experiments we will be considering test subjects' performance on one of the letters in a word or other type of display. In accounting for these results, we have adopted the assumption that responding is always based on the output of the letter level, rather than the output of the word level or some combination of the two. The forced choice is assumed to be based only on this letter-level information. The subject compares the letter selected for the appropriate position against the forced-choice alternatives. If the letter selected is one of the alternatives, then that alternative is chosen in the forced choice. If it is not one of the alternatives, then the model assumes that one of the alternatives would simply be chosen at random.

One somewhat problematical issue involves deciding when to read out the results of processing and select response letters for each letter position. When a target display is simply turned on and left on until the subject responds, and when there is no pressure to respond quickly, we assume that the subject simply waits until the output strengths have reached their asymptotic values. However, when a target display is presented briefly followed by a patterned mask, the activations produced by the target are transient, as we shall see. Under these conditions, we assume that the subject learns through experience in the practice phase of the experiment to read out the results of processing at a time that allows the subject to optimize performance. For simplicity, we have assumed that readout occurs in parallel for all four letter positions.

The Operation of the Model

Now, consider what happens when an input reaches the system. Assume that at time t_0 all prior inputs have had an opportunity to decay, so that the entire system is in its quiescent state, and each node is at its resting level. The presentation of a stimulus initiates a process in which certain features are extracted and excitatory and inhibitory pressures begin to act upon the letter-level nodes. The activation levels of certain letter nodes are pushed above their resting levels. Others receive predominantly inhibitory inputs and

are pushed below their resting levels. These letter nodes, in turn, begin to send activation to those word-level nodes they are consistent with and inhibit those word nodes they are not consistent with. In addition, within a given letter position channel, the various letter nodes attempt to suppress each other, with the strongest ones getting the upper hand. As word-level nodes become active, they in turn compete with one another and send feedback down to the letter-level nodes. If the input features were close to those for one particular set of letters and those letters were consistent with those forming a particular word, the positive feedback in the system will work to rapidly converge on the appropriate set of letters and the appropriate word. If not, they will compete with each other, and perhaps no single set of letters or single word will get enough activation to dominate the others. In this case the various active units might strangle each other through mutual inhibition.

At any point during processing, the results of perceptual processing may be read out from the pattern of activations at the letter level into a buffer, where they may be kept through rehearsal or used as the basis for overt reports. The accuracy of this process depends on a running average of the activations of the correct node and of other competing nodes.

Simulations

Although the model is in essence quite simple, the interactions among the various nodes can become complex, so that the model is not susceptible to a simple intuitive or even mathematical analysis. Instead, we have relied on computer simulations to study the behavior of the model and to see if it is consistent with the empirical data. A description of the actual computer program is given in the Appendix.

For purposes of these simulations, we have made a number of simplifying assumptions. These additional assumptions fall into three classes: (a) discrete rather than continuous time, (b) simplified feature analysis of the input font, and (c) a limited lexicon.

The simulation operates in discrete time slices, or ticks, updating the activations of

all of the nodes in the system once each cycle on the basis of the values on the previous cycle. Obviously, this is simply a matter of computational convenience and not a fundamental assumption. We have endeavored to keep the time slices "thin" enough so that the model's behavior is continuous for all intents and purposes.

Any simulation of the model involves making explicit assumptions about the appropriate featural analysis of the input font. We have, for simplicity, chosen the font and featural analysis employed by Rumelhart (1970) and by Rumelhart and Siple (1974), illustrated in Figure 4. Although the experiments we have simulated employed different type fonts, we assume that the basic results do not depend on the particular font used. The simplicity of the present analysis recommends it for the simulations, though it obviously skirts several fundamental issues about the lower levels of processing.

Finally, our simulations have been restricted to four-letter words. We have equipped our program with knowledge of 1,179 four-letter words occurring at least two times per million in the Kucera and Francis (1967) word count. Plurals, inflected forms, first names, proper names, acronyms, abbreviations, and occasional unfamiliar entries arising from apparent sampling flukes



Figure 4. The features used to construct the letters in the font assumed by the simulation program, and the letters themselves. (From "Process of Recognizing Tachistoscopically Presented Words" by David E. Rumelhart and Patricia Siple, *Psychological Review*, 1974, 81, 99-118. Copyright 1974 by the American Psychological Association. Reprinted by permission.)

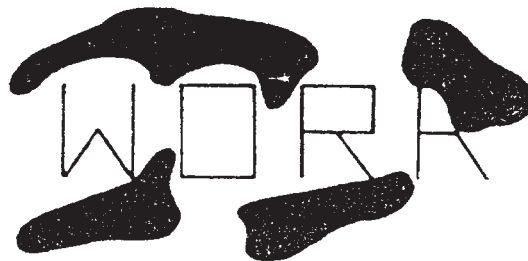


Figure 5. A hypothetical set of features that might be extracted on a trial in an experiment on word perception.

have been excluded. This sample appears to be sufficient to reflect the essential characteristics of the language and to show how the statistical properties of the language can affect the process of perceiving letters in words.

An Example

Let us now consider a sample run of our simulation model. The parameter values employed in the example are those used to simulate all the experiments discussed in the remainder of Part I. These values are described in detail in the following section. For the purposes of this example, imagine that the word *WORK* has been presented to the subject and that the subject has extracted those features shown in Figure 5. In the first three letter positions, the features of the letters *W*, *O*, and *R* have been completely extracted. In the final position a set of features consistent with the letters *K* and *R* have been extracted, with the features that would disambiguate the letter unavailable. We wish now to chart the activity of the system resulting from this presentation. Figure 6 shows the time course of the activations for selected nodes at the word and letter levels, respectively.

At the word level, we have charted the activity levels of the nodes for the words *work*, *word*, *wear*, and *weak*. Note first that *work* is the only word in the lexicon consistent with all the presented information. As a result, its activation level is the highest and reaches a value of .8 through the first 40 time cycles. The word *word* is consistent with the bulk of the information presented and therefore first rises and later is pushed back

down below its resting level, as a result of competition with *work*. The words *wear* and *weak* are consistent with the only letter active in the first letter position, and one of the two active in the fourth letter position. They are also inconsistent with the letters active in Positions 2 and 3. Thus, the activation they receive from the letter level is quite

weak, and they are easily driven down well below zero, as a result of competition from the other word units. The activations of these units do not drop quite as low, of course, as the activation level of words such as *gill*, which contain nothing in common with the presented information. Although not shown in Figure 6, these words attain near-mini-

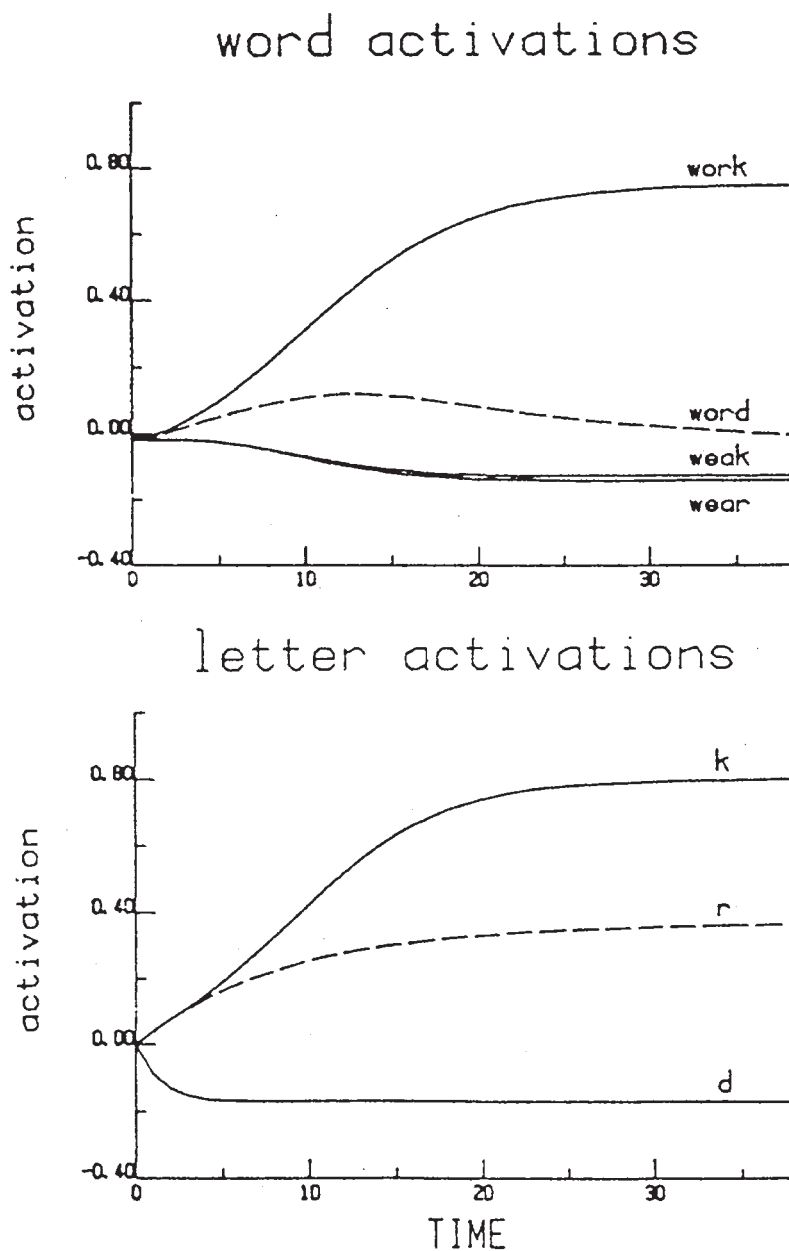


Figure 6. The time course of activations of selected nodes at the word and letter levels after extraction of the features shown in Figure 5.

mum activation levels of about $-.20$ and stay there as the stimulus stays on. Returning to *wear* and *weak*, we note that these words are equally consistent with the presented information and thus drop together for about the first 9 time units. At this point, however, the word *work* has clearly taken the upper hand at the word level, and produces feedback that reinforces the activation of the final *k* and not the final *r*. As a result, the word *weak* receives more activation from the letter level than the word *wear* and begins to gain a slight advantage over *wear*. The strengthened *k* continues to feed activation into the word level and strengthen consistent words. The words that contain an *R* continue to receive activation from the *r* node also, but they receive stronger inhibition from the words consistent with a *K* and are therefore ultimately weakened, as illustrated in the lower panel of Figure 6.

The strong feature-letter inhibition ensures that when a feature inconsistent with a particular letter is detected, that letter will receive relatively strong net bottom-up inhibition. Thus in our example, the information extracted clearly disconfirms the possibility that the letter *D* has been presented in the fourth position, and thus the activation level of the *d* node decreases quickly to near its minimum value. However, the bottom-up information from the feature level supports either a *K* or an *R* in the fourth position. Thus, the activation of each of these nodes rises slowly. These activations, along with those for *W*, *O*, and *R*, push the activation of *work* above zero, and it begins to feed back; by about Time Cycle 4, it is beginning to push the *k* above the *r* (because *WORR* is not a word). Note that this separation occurs just before the words *weak* and *wear* separate. It is the strengthening of *k* due to feedback from *work* that causes them to separate.

Ultimately, the *r* reaches a level well below that of *k* where it remains, and the *k* pushes toward a $.8$ activation level. As discussed below, the word-to-letter inhibition and the letter-to-letter inhibition have both been set to 0. Thus, *k* and *r* both co-exist at moderately high levels, the *r* fed only from the bottom up, and the *k* fed from both bottom up and top down.

Finally, consider the output values for the

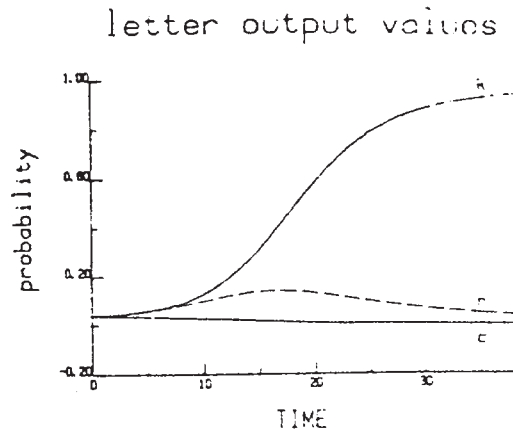


Figure 7. Output values for the letters *r*, *k*, and *d* after presentation of the display shown in Figure 5.

letter nodes *r*, *k*, and *d*. Figure 7 shows the output values for the simulation. The output value is the probability that if a response was selected at time *t*, the letter in question would be selected as the output or response from the system. As intended, these output values grow somewhat more slowly than the values of the letter activations themselves but eventually, as they reach and hold their asymptotic values, come to reflect the activations of the letter nodes. Since in the absence of masking subjects can afford to wait to read out a response until the output values have had a chance to stabilize, they would be highly likely to choose the letter *K* as the response.

Although this example is not very general in that we assumed that only partial information was available in the input for the fourth letter position, whereas full information was available at the other letter positions, it does illustrate many of the important characteristics of the model. It shows how ambiguous sensory information can be disambiguated by top-down processes. Here we have a very simple mechanism capable of applying knowledge of words in the perception of their component letters.

Parameter Selection

Once the basic simulation model was constructed, we began a lengthy process of attempting to simulate the results of several representative experiments in the literature.

Only two parameters of the model were allowed to vary from experiment to experiment: (a) the probability of feature extraction and (b) the timing of the presentation of the masking stimulus if one was used.

The probability of feature extraction is assumed to depend on the visual characteristics of the display. In most of the experiments we will consider, a bright, high-contrast target was used. Such a target would produce perfect performance if not followed by a patterned mask. In these cases probability of feature extraction was fixed at 1.0 and the timing of the target offset and coincident mask onset typically was adjusted to achieve 75% correct performance over the different experimental conditions of interest. In simulating the results of these experiments, we likewise varied the timing of the target offset/mask onset to achieve the right average correct performance from the model.

In some experiments no patterned mask was used, and performance was kept below perfect levels by using a dim or otherwise degraded target display. In these cases the probability of feature extraction was set to a value less than 1.0, which produces about the right overall performance level.

The process of exploring the behavior of the model amounted to an extended search for a set of values for all the other parameters that would permit the model to simulate, as closely as possible, the results of all of the experiments to be discussed later in Part 1, as well as those to be considered in Part 2 (Rumelhart & McClelland, in press). To constrain the search, we adopted various restrictive simplifications. First, we assumed that all nodes have the same maximum activation value. In fact, the maximum was set to 1.0, and served to scale all activations within the model. The minimum activation value for all nodes was set at $-.20$, a value that permits rapid reactivation of strongly inhibited nodes. The decay rate of all nodes was set to the value of $.07$. This parameter effectively serves as a scale factor that determines how quickly things are allowed to change in a single time slice. The $.07$ value was picked after some exploration, since it seemed to permit us to run our simulations with the minimum number of time slices per trial, at the same time as it minimized a kind

of reverberatory oscillation that sets in when things are allowed to change too much on any given time cycle. We also assigned the resting value of zero to all of the letter nodes. The resting value of nodes at the word level was set to a value between $-.05$ and 0 , depending on word frequency.

We have assumed that the weight parameters, α_{ij} and γ_{ij} depend only on the processing levels of nodes i and j and on no other characteristics of their identity. This means, among other things, that the excitatory connections between all letter nodes and all of the relevant word nodes are equally strong, independent of the identity of the words. Thus, for example, the degree to which the node for an initial t excites the node for the word *rock* is exactly the same as the degree to which it excites the node for the word *this*, in spite of a substantial difference in frequency of usage. To further simplify matters, the word-to-letter inhibition was also set to zero. This means that feedback from the word level can strengthen activations at the letter level but cannot weaken them.

The output from the detector network has essentially two parameters. The value $.05$ was used for the parameter r , which determines how quickly the output values change in response to changes in the activations of the underlying nodes. This value is small enough that the output values change relatively slowly, so that transient activations can come and go without much effect on the output. The value 10 was given to the parameter μ in Equation 6 above. The parameter is essentially a scale factor relating activations in the model to response strengths in the Luce formulation.

The values of the remaining parameters were fixed at the values given in Table 1. It is worth noting the differences between the feature-letter influences and the letter-word influences. The feature-letter inhibition is 30 times as strong as the feature-letter excitation. This means that all of the features detected must be compatible with a particular letter before that letter will receive net excitation (since there are only 14 possible features, there can only be a maximum of 13 excitatory inputs whenever there is a single inhibitory input). The main reason for choosing this value was to permit the pre-

Table 1
Parameter Values Used in the Simulations

Parameter	Value
Feature-letter excitation	.005
Feature-letter inhibition	.15
Letter-word excitation	.07
Letter-word inhibition	.04
Word-word inhibition	.21
Letter-letter inhibition	0
Word-letter excitation	.30

sentation of a mask to clear the previous pattern of activation. On the other hand, the letter-word inhibition is actually somewhat less than the letter-word excitation. When only one letter is active in each letter position, this means that the letter level will produce net excitation of all words that share two or more letters with the target word. Because of these multiple activations, strong word-word inhibition is necessary to "sharpen" the response of the word level, as we will see. In contrast, no such inhibition is necessary at the letter level. For these reasons, the letter-letter inhibition has been set to 0, whereas the word-word inhibition has been set to .21.

Comments on Related Formulations

Before turning to the application of the model to the experimental literature, some comments on the relationship of this model to other models extant in the literature is in order. We have tried to be synthetic. We have taken ideas from our own previous work and from the work of others in the literature. In what follows, we attempt to identify the sources of most of the assumptions of the model and to show in what ways our model differs from the models we have drawn on.

First of all, we have adopted the approach of formulating the model in terms similar to the way in which such a process might actually be carried out in a neural or neural-like system. We do not mean to imply that the nodes in our system are necessarily related to the behavior of individual neurons. We will, however, argue that we have kept the kinds of processing involved well within the bounds of capability for simple neural

circuits. The approach of modeling information processing in a neural-like system has recently been advocated by Szentagothai and Arbib (1975) and is represented in many of the articles presented in the volume by Hinton and Anderson (1981) as well as many of the specific models mentioned below.

One case in point is the work of Levin (1976). He proposed a parallel computational system capable of interactive processing that employs only excitation and inhibition as its currency. Although our model could not be implemented exactly in the format of their system (called Proteus), it is clearly in the spirit of their model and could readily be implemented within a variant of the Proteus system.

In a recent article McClelland (1979) has proposed a cascade model of perceptual processing in which activations on each level of the system drive those at the next higher level. This model has the properties that partial outputs are continuously available for processing and that every level of the system processes the input simultaneously. The present model certainly adopts these assumptions. It also generalizes them, permitting information to flow in both directions simultaneously.

Hinton (Note 1) has developed a *relaxation* model for visual perception in which multiple constraints interact by means of incrementing and decrementing real numbered strengths associated with various interpretations of a portion of the visual scene in an attempt to attain a maximally consistent interpretation of the scene. Our model can be considered a relaxation system in which activation levels are manipulated to get an optimal interpretation of an input word.

James Anderson and his colleagues (Anderson, 1977; Anderson, Silverstein, Ritz, & Jones, 1977) and Kohonen and his colleagues (Kohonen, 1977) have developed a pattern recognition system which they call an *associative memory* system. Their system shares a number of commonalities with ours. One feature the models share is the scheme of adding and subtracting weighted excitation values to generate output patterns that represent cleaned-up versions of the input

patterns. In particular, our α_{ij} and γ_{ij} correspond to the matrix elements of the associative memory models. Our model differs in that it has multiple levels and employs a nonlinear cumulation function similar to one suggested by Grossberg (1978), as mentioned above.

Our model also draws on earlier work in the area of word perception. There is, of course, a strong similarity between this model and the logogen model of Morton (1969). What we have implemented might be called a hierarchical, nonlinear, logogen model with feedback between levels and inhibitory interactions among logogens at the same level. We have also added dynamic assumptions that are lacking from the logogen model.

The notion that word perception takes place in a hierarchical information-processing system has, of course, been advocated by several researchers interested in word perception (Adams, 1979; Estes, 1975; Johnston & McClelland, 1980; LaBerge & Samuels, 1974; McClelland, 1976). Our model differs from those proposed in many of these papers in that processing at different levels is explicitly assumed to take place in parallel. Many of the models are not terribly explicit on this topic, although the notion that partial information could be passed along from one level to the next so that processing could go on at the higher level while it was continuing at the lower level had been suggested by McClelland (1976). Our model also differs from all of these others, except that of Adams (1979), in assuming that there is feedback from the word level to the letter level. The general formulation suggested by Adams (1979) is quite similar to our own, although she postulates a different sort of mechanism for handling pseudowords (excitatory connections among letter nodes) and does not present a detailed account.

Our mechanism for accounting for the perceptual facilitation of pseudowords involves, as we will see below, the integration of feedback from partial activation of a number of different words. The idea that pseudoword perception could be accounted for in this way was inspired by Glushko (1979), who suggested that partial activation and synthesis of word pronunciations could ac-

count for the process of constructing a pronunciation for a novel pseudoword.

The feature-extraction assumptions and the bottom-up portion of the word recognition model are nearly the same as those employed by Rumelhart (1970, Note 2) and Rumelhart and Siple (1974). The interactive feedback portion of the model is clearly one of the class of models discussed by Rumelhart (1977) and could be considered a simplified control structure for expressing the model proposed in that paper.

Application of the Simulation Model to Several Basic Findings

We are finally ready to see how well our model fares in accounting for the findings of several representative experiments in the literature. In discussing each account, we will try to explain not only how well the simulation works but why it behaves as it does. As we proceed through the discussion, we will have occasion to describe several interesting synergistic properties of the model that we did not anticipate but discovered as we explored the behavior of the system. As mentioned previously, the actual parameters used in both the examples that we will discuss and in the simulation results we will report are those summarized in Table 1. We will consider the robustness of the model, and the effects of changes in these parameters, in the discussion section at the end of Part 1.

The Word Advantage and the Effects of Visual Conditions

As we noted previously, word perception has been studied under a variety of different visual conditions, and it is apparent that different conditions produce different results. The advantage of words over nonwords appears to be greatest under conditions in which a bright, high-contrast target is followed by a patterned mask with similar characteristics. The word advantage appears to be considerably less when the target presentation is dimmer or otherwise degraded and is followed by a blank white field.

Typical data demonstrating these points (from Johnston & McClelland, 1973) are

Table 2
Effect of Display Conditions on Proportion of Correct Forced Choices in Word and Letter Perception (From Johnston & McClelland, 1973)

Visual condition	Display type	
	Word	Letter with number signs
Bright target/patterned mask	.80	.65
Dim target/blank mask	.78	.73

presented in Table 2. Forced-choice performance on letters in words is compared to performance on letters embedded in a row of number signs (e.g., *READ* vs. *#E##*). The number signs serve as a control for lateral facilitation or inhibition. This factor appears to be important under dim-target/blank-mask conditions.

Target durations were adjusted separately for each condition, so that it is only the pattern of differences within display conditions that is meaningful. The data show that a 15% word advantage was obtained in the bright-target/patterned-mask condition and only a 5% word advantage in the dim-target/blank-mask condition. Massaro and Klitzke (1979) obtained about the same size effects. Various aspects of these results have also been corroborated in two other studies (Juola et al., 1974; Taylor & Chabot, 1978).

To understand the difference between these two conditions it is important to note that in order to get about 75% correct performance in the no-mask condition, the stimulus must be highly degraded. Since there is no patterned mask, the iconic trace presumably persists considerably beyond the offset of the target. It is our assumption that the effect of the blank mask is simply to reduce the contrast of the icon by summing with it. Thus, the limit on performance is not so much the amount of time available in which to process the information as it is the quality of the information made available to the system. In contrast, when a patterned mask is employed, the mask produces spurious inputs, which can interfere with the processing of the target. Thus, in the bright-target/patterned-mask conditions, the pri-

mary limitation on performance is the amount of time that the information is available to the system in relatively legible form rather than the quality of the information presented. This distinction between the way in which blank masks and patterned masks interfere with performance has previously been made by a number of investigators, including Rumelhart (1970) and Turvey (1973). We now consider each of these sorts of conditions in turn.

Word perception under patterned-mask conditions. When a high-quality display is followed by a patterned mask, we assume that the bottleneck in performance does not come in the extraction of feature information from the target display. Thus, in our simulation of these conditions, we assume that all of the features presented can be extracted on every trial. The limitation on performance comes from the fact that the activations produced by the target are subject to disruption and replacement by the mask before they can be translated into a permanent form suitable for overt report. This general idea was suggested by Johnston and McClelland (1973) and considered by a number of other investigators, including Carr et al. (1978), Massaro and Klitzke (1979), and others. On the basis of this idea, a number of possible reasons for the advantage for letters in words have been suggested. One is that letters in words are for some reason translated more quickly into a non-maskable form (Johnston & McClelland, 1973; Massaro & Klitzke, 1979). Another is that words activate representations removed from the direct effects of visual patterned masking (Carr et al., 1978; Johnston & McClelland, 1973, 1980; McClelland, 1976). In the interactive activation model, the reason letters in words fare better than letters in nonwords is that they benefit from feedback that can drive them to higher activation levels. As a result, the probability that the activated letter representation will be correctly encoded is increased.

To understand in detail how this account works, consider the following example. Figure 8 shows the operation of our model for the letter *E* both in an unrelated (*#*) context and in the context of the word *READ* for a visual display of moderately high quality.

We assume that display conditions are sufficient for complete feature extraction, so that only the letters actually contained in the target receive net excitatory input on the basis of feature information. After some number of cycles have gone by, the mask is presented with the same parameters as the target. The mask simply replaces the target display at the feature level, resulting in a completely new input to the letter level. This input, because it contains features incompatible with the letter shown in all four positions, immediately begins to drive down the activations at the letter level. After only a

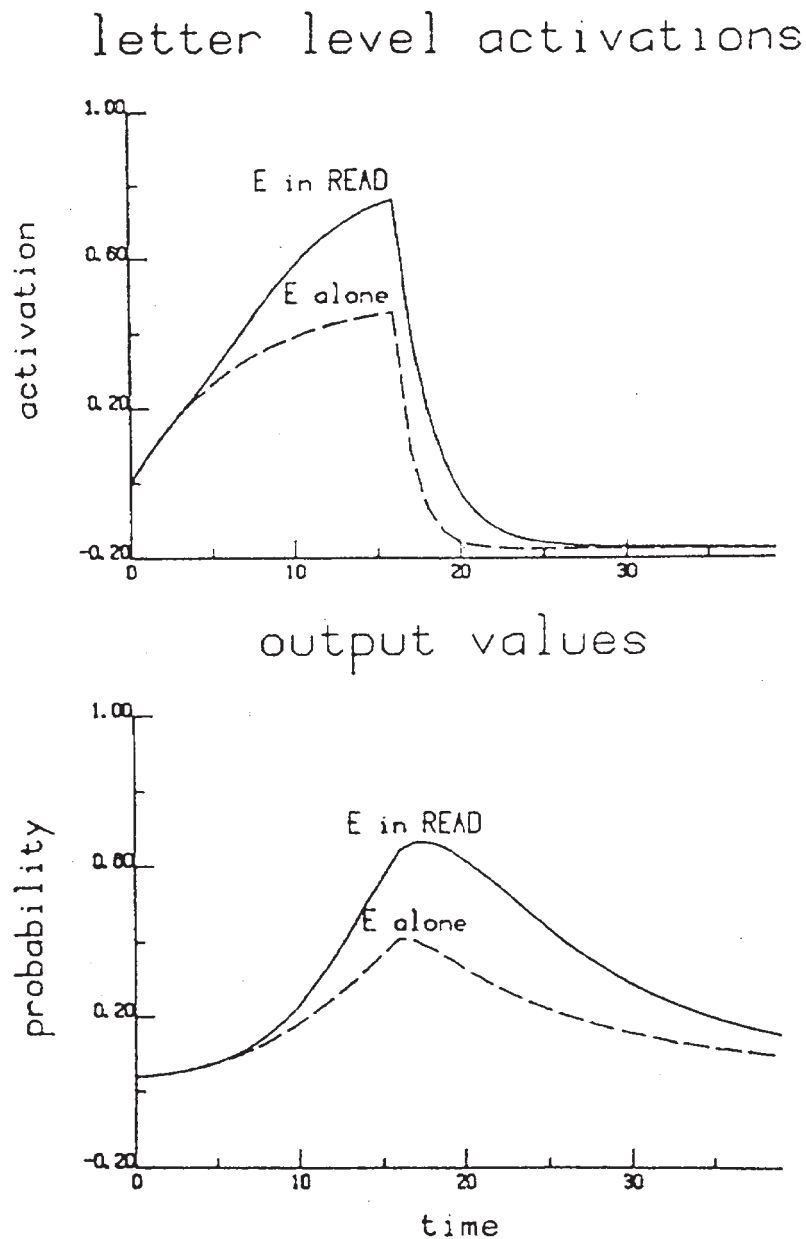


Figure 8. Activation functions (top) and output values (bottom) for the letter *E*, in unrelated context and in the context of the word *READ*.

few more cycles, these activations drop below resting level in both cases. Note that the correct letter was activated briefly, and no competing letter was activated. However, because of the sluggishness of the output process, these activations do not necessarily result in a high probability of correct report. As shown in the top half of Figure 8, the probability of correct report reaches a maximum after 16 cycles at a performance level far below the ceiling.

When the letter is part of the word (in this case, *READ*), the activation of the letters results in rapid activation of one or more words. These words, in turn, feed back to the letter level. This results in a higher net activation level for the letter embedded in the word.

Our simulation of the word advantage under patterned-mask conditions used the stimulus list that was used for simulating the blank-mask results. Since the internal workings of the model are completely deterministic as long as probability of feature extraction is 1.0, it was only necessary to run each item through the model once to obtain the expected probability that the critical letter would be encoded correctly for each item under each variation of parameters tried.

As described previously, we have assumed that readout of the results of processing occurs in parallel for all four letter positions and that the subject learns through practice to choose a time to read out in order to optimize performance. We have assumed that readout time may be set at a different point in different conditions, as long as they are blocked so that the subject knows in advance what type of material will be presented on each trial in the experiment. Thus, in simulating the Johnston and McClelland (1973) results, we allowed for different readout times for letters in words and letters in unrelated contexts, with the different times selected on the basis of practice to optimize performance on each type of material.

A final feature of the simulation is the duration of the target display. This was varied to produce an average performance on both letters embedded in number signs and letters in words that was as close as possible to the average performance on these two conditions in the 1973 experiment of John-

ston and McClelland. The value used for the run reported below was 15 cycles. As in the Johnston and McClelland study, the mask followed the target immediately.

The simulation replicated the experimental data shown in Table 2 quite closely. Accuracy on the forced choice was 81% correct for the letters embedded in words and 66% correct for letters in an unrelated (#) context.

It turns out that it is not necessary to allow for different readout times for different material types. A repetition of the simulation produced a 15% word advantage when the same readout time was chosen for both single letters and letters in words, based on optimal performance averaged over the two material types. Thus, the model is consistent with the fact that the word advantage does not depend on separating the different stimulus types into separate blocks (Massaro & Klitzke, 1979).

Perception of letters in words under conditions of degraded input. In conditions of degraded (but not abbreviated) input, the role of the word level is to selectively reinforce possible letters that are consistent with the visual information extracted and that are also consistent with the words in the subject's vocabulary. Recall that the task requires the subject to choose between two letters, both of which (on word trials) make a word with the rest of the context. There are two distinct cases to consider. Either the featural information extracted from the to-be-probed letter is sufficient to distinguish between the alternatives, or it is not. Whenever the featural information is consistent with both of the forced-choice alternatives, any feedback will selectively enhance both alternatives and will not permit the subject to distinguish between them. When the information extracted is inconsistent with one of the alternatives, the model produces a word advantage. The reason is that we assume forced-choice responses are based not on the feature information itself but on the subject's best guess about what letter was actually shown. Feedback from the word level increases the probability of correct choice in those cases where the subject extracts information that is inconsistent with the incorrect alternative but consistent with the correct alternative

and a number of others. Thus, feedback would have the effect of helping the subject select the actual letter shown from several possibilities consistent with the set of extracted features. Consider again, for example, the case of the presentation of *WORD* discussed above. In this case, the subject extracted incomplete information about the final letter consistent with both *R* and *K*. Assume that the forced choice the subject was to face on this trial was between a *D* and a *K*. The account supposes that the subject encodes a single letter for each letter position before facing the forced choice. Thus, if the features of the final letter had been extracted in the absence of any context, the subject would encode *R* or *K* equally often, since both are equally compatible with the features extracted. This would leave the subject with the correct response some of the time. But if *R* were chosen instead, the subject would enter the forced choice between *D* and *K* without knowing the correct answer directly. When the whole word display is shown, the feedback generated by the processing of all of the letters greatly strengthens the *K*, increasing the probability that it will be chosen over the *R* and thus increasing the probability that the subject will proceed to the forced choice with the correct response in mind.

Our interpretation of the small word advantage in blank-mask conditions is a specific version of the early accounts of the word advantage offered by Wheeler (1970) and Thompson and Massaro (1973) before it was known that the effect depends on masking. Johnston (1978) has argued that this type of account does not apply under patterned-mask conditions. We are suggesting that it does apply to the small word advantage obtained under blank-mask conditions like those of the Johnston and McClelland (1973) experiment. We will see below that the model offers a different account of performance under patterned-mask conditions.

We simulated our interpretation of the small word advantage obtained in blank-mask conditions in the following way. A set of 40 pairs of four-letter words that differed by a single letter was prepared. The differing letters occurred in each position equally often. From these words corresponding con-

trol pairs were generated in which the critical letters from the word pairs were presented in nonletter contexts (#s). Because they were presented in nonletter contexts, we assumed that these letters did not engage the word processing system at all.

Each member of each pair of items was presented to the model four times, yielding a total of 320 stimulus presentations of word stimuli and 320 presentations of single letters. On each presentation, the simulation sampled a random subset of the possible features to be detected by the system. The probability of detection of each feature was set at .45. As noted previously, these values are in a ratio of 1 to 30, so that if any one of the 14 features extracted is inconsistent with a particular letter, that letter receives net inhibition from the features and is rapidly driven into an inactive state.

For simplicity, the features were treated as a constant input, which remained on while letter and word activations (if any) were allowed to take place. At the end of 50 processing cycles, which is virtually asymptotic, output was sampled. Sampling results in the selection of one letter to fill each position; the selected letter is assumed to be all the subject takes away from the target display. As described previously, the forced choice is assumed to be based only on this letter identity information. The subject compares the letter selected for the appropriate position against the forced-choice alternatives. If the letter selected is one of the alternatives, then that alternative is selected. If it is not one of the alternatives, then one of the two alternatives is simply picked at random.

The simulation produced a 10% advantage for letters in words over letters embedded in number signs. Probability-correct forced choice for letters embedded in words was 78% correct, whereas for letters in number signs, performance was 68% correct.

The simulated results for the no-mask condition clearly show a smaller word advantage than for the patterned-mask case. However, the model produces a larger word advantage, which is observed in the experiment (Table 2). As Johnston (1978) has pointed out, there are a number of reasons why an account such as the one we have offered would overestimate the size of the

word advantage. First, subjects may occasionally be able to retain an impression of the actual visual information they have been able to extract. On such occasions, feedback from the word level will be of no further benefit. Second, even if subjects only retain a letter identity code, they may tend to choose the forced-choice alternative that is most similar to the letter encoded—instead of simply guessing—when the letter encoded is not one of the two choices. This would tend to result in a greater probability of correct choices and less of a chance for feedback to increase accuracy of performance. It is hard to know exactly how much these factors should be expected to reduce the size of the word advantage under these conditions, but they would certainly bring it more closely in line with the results.

Perception of Letters in Regular Nonwords

One of the most important findings in the literature on word perception is that an item need not be a word in order to produce facilitation with respect to unrelated letter or single letter stimuli. The advantage for pseudowords over unrelated letters has been obtained in a very large number of studies (Aderman & Smith, 1971; Baron & Thurston, 1973; Carr et al., 1978; McClelland, 1976; Spoehr & Smith, 1975). The pseudoword advantage over single letters has been obtained in three studies (Carr et al., 1978; Massaro & Klitzke, 1979; McClelland & Johnston, 1977).

Our model produces the facilitation for pseudowords by allowing them to activate nodes for words that share more than one letter in common with the display. When they occur, these activations produce feedback which strengthens the letters that gave rise to them just as in the case of words. These activations occur in the model if the strength of letter-to-word inhibition is reasonably small compared to the strength of letter-to-word excitation.

To see how this takes place in detail, consider a brief presentation of the pseudoword *MAVE* followed by a patterned mask. (The pseudoword is one used by Glushko, 1979, in developing the idea that partial activa-

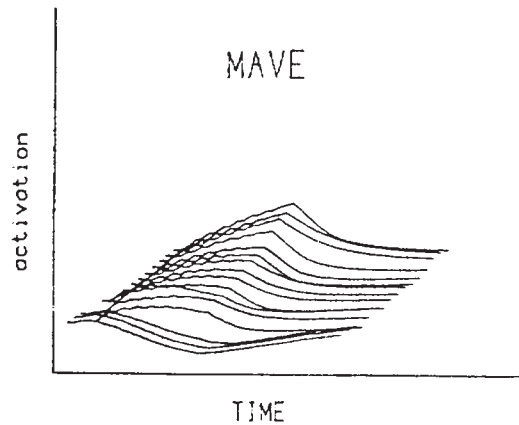


Figure 9. Activation at the word level upon presentation of the nonword *MAVE*.

tions of words are combined to derive pronunciations of pseudowords.) As illustrated in Figure 9, presentation of *MAVE* results in the initial activation of 16 different words. Most of these words, like *have* and *gave*, share three letters with *MAVE*. By and large, these words steadily gain in strength while the target is on and produce feedback to the letter level, sustaining the letters that supported them.

Some of the words are weakly activated for a brief period of time before they fall back below zero. These typically are words like *more* and *many*, which share only two letters with the target but are very high in frequency, so they need little excitation before they exceed threshold. But soon after they exceed threshold, the total activation at the word level becomes strong enough to overcome the weak excitatory input, causing them to drop down just after they begin to rise. Less frequent words sharing two letters with the word displayed have a worse fate still. Since they start out initially at a lower value, they generally fail to receive enough excitation to reach threshold. Thus, when there are several words that have three letters in common with the target, words that share only two letters with the target tend to exert little or no influence. In general then, with pronounceable pseudoword stimuli, the amount of feedback—and hence the amount of facilitation—depends primarily on the activation of nodes for words that share three letters with a displayed pseudoword. It is the

nodes for these words that primarily interact with the activations generated by the presentation of the actual target display. In what follows we will call the words that have three letters in common with the target letter string the neighbors of that string.

The amount of feedback a particular letter in a nonword receives depends, in the model, on two primary factors and two secondary factors. The two primary factors are the number of words in the neighborhood that contain the target letter and the number of words that do not. In the case of the *M* in *MAVE*, for example, there are seven words in the neighborhood of *MAVE* that begin with *M*, so the *m* node gets excitatory feedback from all of these. These words are called the "friends" of the *m* node in this case. Because of competition at the word level, the amount of activation that these words receive depends on the total number of words that have three letters in common with the target. Those that share three letters with the target but are inconsistent with the *m* node (e.g., *have*) produce inhibition that tends to limit the activation of the friends of the *m* node, and can thus be considered its "enemies." These words also produce feedback that tends to activate letters that were not actually presented. For example, activation from *have* produces excitatory input to the *h* node, thereby producing some competition with the *m* node. These activations, however, are usually not terribly strong. No one word gets very active, and so letters not in the actual display tend to get fairly weak excitatory feedback. This weak excitation is usually insufficient to overcome the bottom-up inhibition acting on nonpresented letters. Thus, in most cases, the harm done by top-down activation of letters that were not shown is minimal.

A part of the effect we have been describing is illustrated in Figure 10. Here, we compare the activations of the nodes for the letters in *MAVE*. Without feedback, the four curves would be identical to the one single-letter curve included for comparison. So although there is facilitation for all four letters, there are definitely differences in the amount, depending on the number of friends and enemies of each letter. Note that within a given pseudoword, the total number of friends and enemies (i.e., the total number

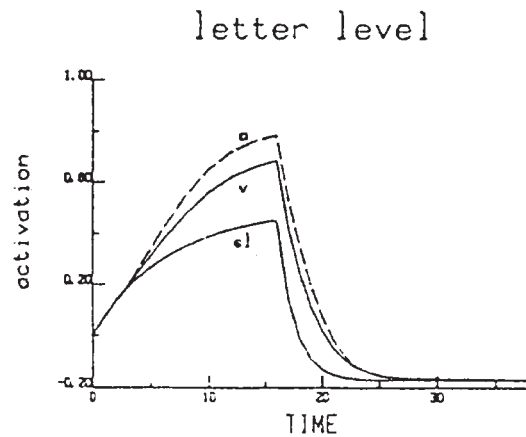


Figure 10. Activation functions for the letters *a* and *v* on presentation of *MAVE*. (Activation function for *e* is indistinguishable from function for *a*, and that for *m* is similar to that for *v*. The activation function for a single letter (*s*), or a letter in an unrelated context is included for comparison.)

of words with three letters in common) is the same for all the letters.

There are two other factors that affect the extent to which a particular word will become active at the word level when a particular pseudoword is shown. Although the effects of these factors are only weakly reflected in the activations at the letter level, they are nevertheless interesting to note, since they indicate some synergistic effects that emerge from the interplay of simple excitatory and inhibitory influences in the neighborhood. These are the *rich-get-richer effect* and the *gang effect*. The rich-get-richer effect is illustrated in Figure 11, which compares the activation curves for the nodes for *have*, *gave*, and *save* under presentation of *MAVE*. The words differ in frequency, which gives the words slight differences in baseline activation. What is interesting is that the difference gets magnified; so that at the point of peak activation, there is a much larger difference. The reason for the amplification can be seen by considering a system containing only two nodes, *a* and *b*, starting at different initial positive activation levels, *a* and *b* at time *t*. Let us suppose that *a* is stronger than *b* at *t*. Then at *t* + 1, *a* will exert more of an inhibitory influence on *b*, since inhibition of a given node is determined by the sum of the activations of all nodes other than itself. This

the "rich get richer" effect.

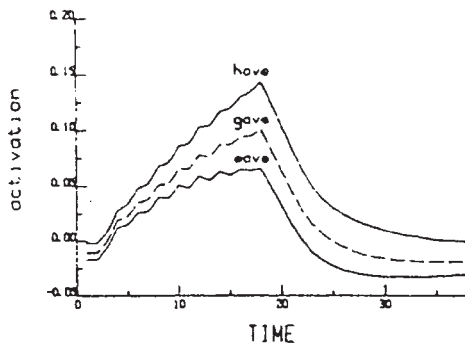


Figure 11. The rich-get-richer effect. (Activation functions for the nodes for *have*, *gave*, and *save* under presentation of *MAVE*.)

advantage for the initially more active nodes is compounded further in the case of the effect of word frequency by the fact that more frequent words creep above threshold first, thereby exerting an inhibitory effect on the lower frequency words when the latter are still too weak to fight back at all.

Even more interesting is the gang effect, which depends on the coordinated action of a related set of word nodes. This effect is depicted in Figure 12. Here, the activation curves for the *move*, *male*, and *save* nodes are compared. In the language, *move* and *make* are of approximately equal frequency, so their activations start out at about the same level. But they soon pull apart. Similarly, *save* starts out below *move* but soon reaches a higher activation. The reason for these effects is that *male* and *save* are both members of gangs with several members, whereas *move* is not. Consider first the difference between *male* and *move*. The reason for the difference is that there are several words that share the same three letters with *MAVE* as *male* does. In the list of words used in our simulations, there are six. These words all work together to reinforce the *m*, *a*, and the *e* nodes, thereby producing much stronger reinforcement for themselves. Thus, these words make up a gang called the *ma_e* gang. In this example, there is also a *_ave* gang consisting of 6 other words, of which *save* is one. All of these work together to reinforce the *a*, *v*, and *e*. Thus, the *a* and *e* are reinforced by two gangs, whereas the

letters *v* and *m* are reinforced by only one each. Now consider the word *move*. This word is a loner; there are no other words in its gang, the *m_ve* gang. Although two of the letters in *move* receive support from one gang each, and one receives support from both other gangs, the letters of *move* are less strongly enhanced by feedback than the letters of the members of the other two gangs. Since continued activation of one word in the face of the competition generated by all of the other partially activated words depends on the activations of the component letter nodes, the words in the other two gangs eventually gain the upper hand and drive *move* back below the activation threshold.

As our study of the *MAVE* example illustrates, the pattern of activation produced by a particular pseudoword is complex and idiosyncratic. In addition to the basic friends and enemies effects, there are also the rich-get-richer and the gang effects. These effects are primarily reflected in the pattern of activation at the word level, but they also exert subtle influences on the activations at the letter level. In general though, the main result is that when the letter-to-word inhibition is low, all four letters in the pseudoword receive some feedback reinforcement. The result, of course, is greater accuracy of reporting letters in pseudowords compared to single letters.

Comparison of performance on words and pseudowords. Let us now consider the fact that the word advantage over pseudowords

the "gang" effect

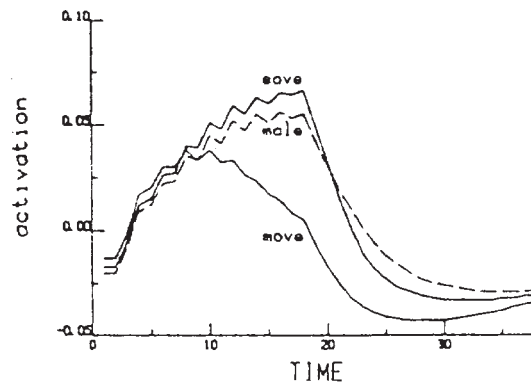


Figure 12. The gang effect. (Activation functions for *move*, *male*, and *save* under presentation of *MAVE*.)

Table 3
Actual and Simulated Results of the
McClelland & Johnston (1977) Experiments
(Proportion of Correct Forced Choice)

Result class	Target type		
	Word	Pseudoword	Single letter
Actual data			
High BF	.81	.79	.67
Low BF	.78	.77	.64
Average	.80	.78	.66
Simulation			
High BF	.81	.79	.67
Low BF	.79	.77	.67
Average	.80	.78	.67

Note. BF = bigram frequency.

is generally rather small in experiments where the subject knows that the stimuli include pseudowords. Some fairly representative results, from the study of McClelland and Johnston (1977), are illustrated in Table 3. The visual conditions of the study were the same as those used in the patterned-mask condition in Johnston and McClelland (1973). Trials were blocked, so subjects could adopt the optimum strategy for each type of material. The slight word-pseudoword difference, though representative, is not actually statistically reliable in this study.

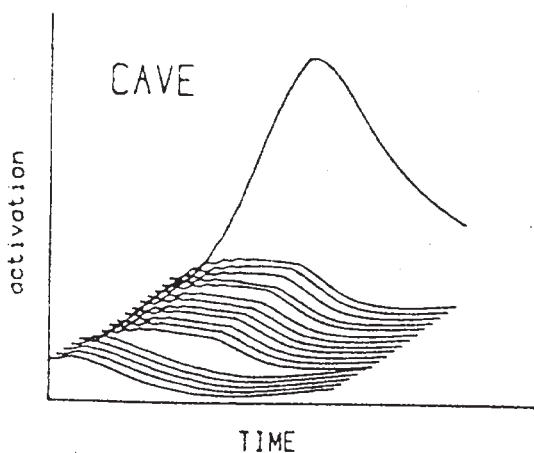


Figure 13. Activity at the word level upon presentation of *CAVE*, with weak letter-to-word inhibition.

a in different contexts

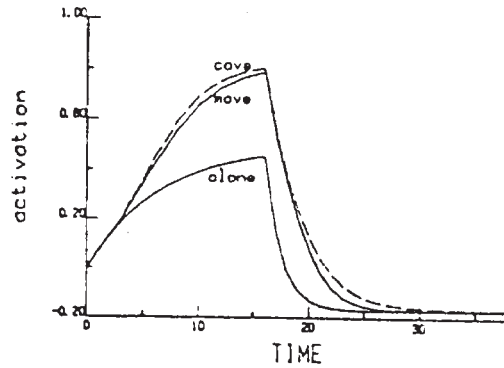


Figure 14. Activation functions for the letter *a*, under presentation of *CAVE* and *MAVE* and alone.

Words differ from pseudowords in that a word strongly activates one node at the word level, whereas a pseudoword does not. While we would tend to think of this as increasing the amount of feedback for words as opposed to pseudowords, there is the word-level inhibition that must be taken into account. This inhibition tends to equalize the total amount of activation at the word level between words and pseudowords. With words, the word shown tends to dominate the pattern of activity, thereby keeping all the words that have three letters in common with it from achieving the activation level they would reach in the absence of a node activated by all four letters. This situation is illustrated for the word *CAVE* in Figure 13. The result is that the sum of the activations of all the active units at the word level is not much different between the two cases. Thus, *CAVE* produces only slightly more facilitation for its constituent letters than *MAVE*, as illustrated in Figure 14.

In addition to the leveling effect of competition at the word level, it turned out that in our model, one of the common design features of studies comparing performance on words and pseudowords would operate to keep performance relatively good on pseudowords. In general, the stimulus materials used in most of these studies are designed by beginning with a list of pairs of words that differ by one letter (e.g., *PEEL-PEEP*). From each pair of words, a pair of non-words is generated, differing from the orig-

inal word pair by just one of the context letters and thereby keeping the actual target letters—and as much of the context as possible—the same between word and pseudoword items (e.g., *TEEL-TEEP*). A previously unnoticed side effect of this matching procedure is that it ensures that the critical letter in each pseudoword has at least one friend, namely the word from the matching pair that differs from it by one context letter. In fact, most of the critical letters in the pseudowords used by McClelland and Johnston (1977) tended to have relatively few enemies, compared to the number of friends. In general, a particular letter should be expected to have three times as many friends as enemies. In the McClelland and Johnston stimuli, the great majority of the stimuli had much larger differentials. Indeed, more than half of the critical letters had no enemies at all.

The puzzling absence of cluster frequency effects. In the account we have just described, facilitation of performance on letters in pseudowords was explained by the fact that pseudowords tend to activate a large number of words, and these words tend to work together to reinforce the activations of letters. This account might seem to suggest that pseudowords that have common letter clusters, and therefore have several letters in common with many words, would tend to produce the greatest facilitation. However, this factor has been manipulated in a number of studies, and little has been found in the way of an effect. The McClelland and Johnston (1977) study is one case in point. As Table 3 illustrates, there is only a slight tendency for superior performance on high cluster frequency words. This slight tendency is also observed in single letter control stimuli, suggesting that the difference may be due to differences in perceptibility of the target letters in the different positions, rather than cluster frequency per se. In any case, the effect is very small. Other studies have likewise failed to find any effect of cluster frequency (Spoehr & Smith, 1975; Manelis, 1974). The lack of an effect is most striking in the McClelland and Johnston study, since the high and low cluster frequency items differed widely in cluster frequency as measured in a number of ways.

In our model, the lack of a cluster frequency effect is due to the effect of mutual inhibition at the word level. As we have seen, this mutual inhibition tends to keep the total activity at the word level roughly constant over a variety of different input patterns, thereby greatly reducing the advantage for high cluster frequency items. Items containing infrequent clusters tend to activate few words, but there is less competition at the word level, so that the words that do become active reach higher activation levels.

The situation is illustrated for the nonwords *TEEL* and *HOET* in Figure 15. Although *TEEL* activates many more words, the total activation is not much different in the two cases.

The total activation is not, of course, the whole story. The ratio of friends to enemies is also important. And it turns out that this ratio is working against the high cluster items more than the low cluster items. In McClelland and Johnston's stimuli, only one of the low cluster frequency nonword pairs had critical letters with any enemies at all! For 23 out of 24 pairs, there was at least one friend (by virtue of the method of stimulus construction) and no enemies. In contrast, for the high cluster frequency pairs, there was a wide range, with some items having several more enemies than friends.

To simulate the McClelland and Johnston (1977) results, we had to select a subset of their stimuli, since some of the words they used were not in our word list. The stimuli had been constructed in sets containing a word pair, a pseudoword pair, and a single letter pair that differed by the same letters in the same position (e.g., *PEEL-PEEP*, *TEEL-TEEP*, *___L___P*). We simply selected all those sets in which both words in the pair appeared in our list. This resulted in a sample of 10 high cluster frequency sets and 10 low cluster frequency sets. The single letter stimuli derived from the high and low cluster frequency pairs were also run through the simulation. Both members of each pair were tested.

Since the stimuli were presented in the actual experiment blocked by material type, we separately selected an optimal time for readout for words, pseudowords, and single letters. Readout time was the same for high

and low cluster frequency items of the same type, since these were presented in a mixed list in the actual experiment. As in the simulation of the Johnston and McClelland (1973) results, the display was presented for a duration of 15 cycles.

The simulation results, shown in Table 3, reveal the same general pattern as the actual

data. The magnitude of the pseudoword advantage over single letters is just slightly smaller than the word advantage, and the effect of cluster frequency is very slight.

We have yet to consider how the model deals with unrelated letter strings. This depends a little on the exact characteristics of the strings. First let us consider truly ran-

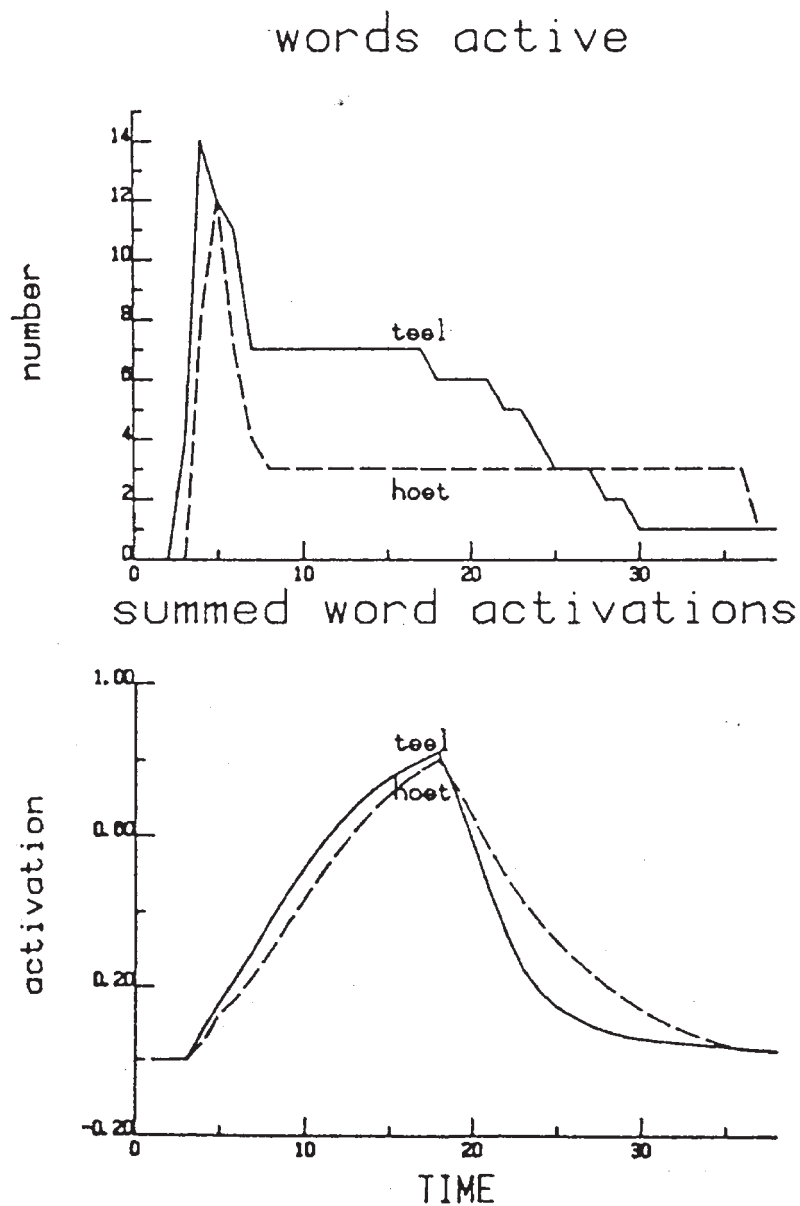


Figure 15. The number of words activated (top) and the total activation at the word level (bottom) upon presentation of the nonwords *TEEL* and *HOET*.