


---

**A Unification-based Model of Perceptual Learning in Reading: Part 2. Computational Implementation**

**JOHN E. MCMENNEY**

Division of Education
Indiana University at South Bend, South Bend, IN 46634, USA

**Abstract**

This paper describes a two-stage, feature-based theory of letter and word perception and learning that is implemented as a Prolog computer simulation. The first stage of perception consists of the generation of a sensory representation or iconic. In the second stage, the iconic is matched with existing perceptual schemata that make up the perceptual repertoire of the system. Matching of icons and perceptual schemata is based on logical unification. Unification-based (UB) matching leads to two properties that make it especially useful in the present context. One property is that UB matching introduces an interactive element in a model that retains the theoretical simplicity of an essentially bottom-up mechanism. A second property is that UB matching provides a conceptual framework for understanding the acquisition and development of perceptual schemata. What is, perhaps, most unique about the proposed model is its emphasis on developmental and instructional issues that have traditionally been overlooked by theorists interested in explaining letter and word perception.

Context has long been known to influence the accuracy of letter and word recognition. As long ago as 1898, Eddmann and Dodge demonstrated that words, and the letters in those words, could be identified under viewing conditions that made identification of isolated letters impossible. Zeigler (1900) soon thereafter demonstrated that context effects were not only limited to words, but also appeared to lead to increased perceptual reports when syllable-like letter patterns (i.e., pseudowords) were used.

In the years since these early studies, a number of consistently observed context effects have been noted in the literature of letter and word recognition. It is the purpose of the present section of this paper to identify those consistently observed "basic findings," and in doing this, to establish a standard by which the proposed model can be evaluated.

The context-effect literature of letter and word recognition appears to support at least four basic findings. These four basic findings include: word superiority effects (WSE), the pseudoword effect (PSE), visual factor effects (VFE), and repetition effects (RE).

**Word Superiority Effects (WSE)**

WSE are among the oldest and most consistently observed of the basic findings in word and letter recognition. In its simplest form, a word superiority effect refers to the enhanced
perceptibility words seem to have over individual letters. Early studies (Cattell, 1885, 1886; Erdmann & Dodge, 1888; Zeiliger, 1900) were usually carried out with this simplest kind of word superiority effect in mind. Typically, subjects in these early studies viewed briefly presented words or letters and then attempted to name or pronounce these stimuli as quickly as possible. Dependent measures were usually of two kinds: correct or incorrect response latency measures and accuracy measures. In either case, however, comparisons were usually limited to differences between whole words and individual letter performance measures.

More recent examinations of WSE have focused on more specific aspects of word superiority. In one important and influential study, Reicher (1969) examined the relative perceptibility of letters in a number of different contexts that included individual letters, random letter pairs, four-letter words, and four-letter nonwords. His findings indicated significantly better recognition of the letters in words than letters either in isolation or in nonwords. This finding, replicated and extended in many other studies, has come to be known as the word-letter effect (Henderson, 1982) and represents one aspect of the general WSE which include the more global variant of the effect known to the researchers of the late 1800s.

It appears, therefore, that at least two kinds of word superiority effect must be accounted for by theories of letter and word recognition. One effect, hereafter referred to as the Cattell effect, is that words appear to be recognized as fast and as accurately as individual letters. The second effect, hereafter referred to as the word-letter effect (WLE), is that letters in words appear to be recognized both faster and more accurately than letters either in isolation or in orthographically irregular contexts. These two specific effects will hereafter be referred to collectively under the general heading of word superiority effects (WSE).

The Pseudoword Effect (PSE)

Almost as old as the Cattell effect, the pseudoword effect was first identified by Zeiliger (1900). Zeiliger showed that perceptual reports of briefly exposed nonword letter patterns were enhanced if vowels were interspersed between consonants, creating syllable-like organization within the letter patterns (e.g., Vogiter vs. VGTR). Nonword letter patterns such as those employed by Zeiliger have more recently become known as pseudowords, and, consequently, Zeiliger's finding as an expression of the pseudoword effect.

Like the word superiority effect of the late 1800s, the pseudoword effect has been consistently confirmed and extended by numerous more recent studies (Adams & Smith, 1971; Baron & Thurston, 1973; Cur, Davidson, & Hawkins, 1978; Massaro & Kistke, 1979; McClelland & Johnston, 1977; McClelland & Rumelhart, 1981; Sproehr & Smith, 1975). The pseudoword effect corresponds to the word-letter effect reported by Reicher (1969) since the dependent variable of interest in pseudoword studies is usually accuracy or speed of identification of individual letters in letter strings. The pseudoword effect, therefore, refers to the enhanced perceptibility of letters in pseudowords over letters either in isolation or in orthographically irregular contexts.

Visual Factor Effects (VFE)

Recent research, in an effort to separate visual and perceptual factors in recognition, has increasingly turned to the use of visual quality and visual masks as variables in the exploration of recognition processes. Erdmann and Dodge (1986) had pioneered the use of visual quality as an experimental variable in their demonstration of the superior perceptibility of words over letters. More recently, however, the visual quality of targets and the explicit manipulation of visual masking has revealed that letters and word recognition is, indeed, influenced by these factors (Johnston & McClelland, 1973; Juola, Leavitt, & Choe, 1974; Massaro & Kistke, 1979; McClelland & Rumelhart, 1981; Rumelhart & McClelland, 1982; Taylor & Chabot, 1978).

Although WSE are quite pronounced when a distinct target is followed by a distinct visual mask, the enhanced perceptibility of words over letters (the Cattell effect) disappears entirely, and may even be reversed, when the target is indistinct and followed by a blank, nonpatterned field rather than a mask. In addition, under these same circumstances, the perceptual advantage of letters in words over nonwords (the word-letter effect) is greatly reduced.

Repetition Effects (RE)

Repetition effects refer to the influence of multiple presentations of the same stimulus within a single experimental session or across different experimental sessions that occur within a limited period of time. Specifically, it appears that the speed and accuracy with which a word or pseudoword is recognized depends on whether that stimulus has been recently viewed (Carroll & Kirsner, 1982; Feustel, Shiffrin, & Salasoo, 1983; Jacoby, 1983; Murrell & Morton, 1974; Salasoo, Shiffrin, & Feustel, 1985). Moreover, it would appear that after relatively few exposures, words and pseudowords are recognized with almost equal facility, and this facility is retained throughout experimental sessions, declining only over extended periods of time (Salasoo et al., 1985), suggesting that the pseudowords achieve the memorial status of real words.

Repetition effects have also been related to a priming effect, based on orthographic similarity (Feustel et al., 1983), which appears to influence words and pseudowords differently. Apparently, orthographically similar primes facilitate speed of pseudoword recognition but do not appear to facilitate the speed of word recognition, although both words and pseudowords are subject to a greater proportion of recognition errors when orthographically similar primes are employed. This differential priming effect for words and pseudowords is especially interesting since one of the most widely accepted theoretical models (McClelland & Rumelhart, 1981) fails to account for, or even address, such effects.

Conceptual Foundations

The model of letter and word recognition and learning proposed in this paper, hereafter referred to as the PERceptual Learner (or PERL) model, is a feature-based theory. Letters are assumed to be composed of a limited set of basic features (i.e., line segments). Words are, in turn, composed of sequences of feature sets that represent letters. The set of basic features and the resulting font are depicted in Figure 1. This is the same font and feature set proposed by Rumelhart and Siple (1974) and subsequently employed in the interactive activation model of McClelland and Rumelhart (1981, 1988).

Two different kinds of features are employed in the model. Presence features (f1, f2, ... f14) indicate the presence of one or more of the 14 line segments in the Rumelhart and Siple (RS) font at a given position in a sequence. Absence features (a1, a2, ... a14) indicate the absence of one or more of the 14 line segments in the RS font at a given sequence position. Presence and absence features with the same identifying integer (e.g., a3) and f2, and a3)
A Unification-Based Model of Perceptual Learning in Reading

of selection of each feature in the stimulus is 1, the resulting sensory icon is said to be complete since every feature in the visual stimulus is represented in the icon. When the probability of selection of one or more features is less than 1, the resulting sensory icon may be incomplete. When an icon is incomplete, one or more features present in the original visual stimulus are not represented in the sensory icon.

When a sensory representation has been generated, the first stage of the recognition process has been completed. The next stage of processing is to match the sensory icon that has been generated with cognitive representations resident in long-term memory. Matching of sensory representations with cognitive representations is based on a search through the perceptual repertoire (PR) of the system. A match occurs when a sensory representation will unify (Robinson, 1965) with a cognitive representation.

Two characteristics of unification make it especially useful for modeling letter and word recognition and acquisition. One characteristic is that pattern matching by unification allows a perceptual system to begin recognizing objects with relatively simple cognitive representations without constraining the development of more complex perceptual schemata at a later point. In fact, matching by unification provides a natural framework for conceptualizing perceptual schemata and their development. On the first exposure to a visual stimulus, a perceptual system may construct a cognitive representation that includes only a few salient features. As the system becomes more familiar with the stimulus through subsequent exposures, however, matching by unification allows missing features to be gradually filled in.

A second characteristic of unification that makes it useful in the present context is that incomplete sensory representations can be matched with complete cognitive units. Cognitive units function as perceptual schemata that can contribute new information to the recognition process. In other words, even though the PERL model takes a distinctly bottom-up approach, unification-based matching results in a distinctly interactive recognition process. The cognitive matching stage of the recognition process can introduce new information, and this information is, within the system, indistinguishable from the information provided by the original visual stimulus.

Computational Implementation of the Model

Recent developments in the field of artificial intelligence (AI) provide a number of important new tools and methods that can be applied in the development of reading theory. One of the recent innovations to come out of AI research is the logic-programming language Prolog (Arity Corporation, 1986; Clocksin & Mellish, 1987; Sterling & Shapiro, 1986), a language that is especially well-suited to the development of simulations of cognitive aspects of language acquisition and use. The model described in the present paper is one example of the application of logic-programming techniques (and Prolog in particular) in the development of theoretical models in reading research. The purpose of this section is to demonstrate the capabilities of the PERL model in a concrete manner by demonstrating the performance of the model with simulated experimental trials.

Trial 1: Basic Letter Learning/Recognition Mechanisms

Trial 1 demonstrates two fundamental kinds of learning simulated by the PERL system: random associative (RA) learning and elaborative (EL) learning. Table 1 depicts the development of a perceptual schema for the letter “A” over the course of seven successive exposures. The initial perceptual repertoire (PR0) for this trial is empty; the system starts

---

Figure 1.

are said to be incompatible since they represent mutually exclusive feature detection outcomes for any single position in a sequence.

The theory proposed here is what philosophers refer to as a sense-data theory of perception (Audi, 1988). Sense-data theories are characterized by a sensory mechanism that mediates between material objects and perceptual experience. The sense-data approach adopted here distinguishes three perceptual domains. The material domain corresponds to the notion of the “external world” and is defined by material objects (i.e., physical characters and spatial arrangements of characters). The sensory domain corresponds to the world of sense-data and is defined by the sensory objects created by a sensory system, usually in the transformation of a material input into a sensory representation of that input. The cognitive domain corresponds to the knowledge a perceptual system has that allows it to identify a sense-data with some past sensory experience(s). Perception, according to the account proposed here, is a process that relates objects across these three domains.

Letter and word learning, according to the proposed model, occurs in two different ways. One kind of learning consists of adding new cognitive representations to the perceptual repertoire, where a perceptual repertoire is simply the set of recognizable patterns known to a perceptual system. The second kind of learning is a consequence of the elaboration of an existing, but incomplete, cognitive representation. Letter and word recognition is conceived of as a matching of sensory representations or “icons” with cognitive representations residing in long-term memory.

Letter and word recognition is a two-stage process. The first stage consists of the generation of a sensory icon. A sensory icon corresponds to the representation of the stimulus in short-term memory. The second stage of recognition consists of the matching of the sensory icon with representations of stimuli that have been acquired and stored in long-term memory.

Generation of sensory icons in the PERL model is assumed to be governed by a stimulus sampling model similar to that proposed by Estes (1959). According to the stimulus sampling model, features present in a stimulus have a certain probability (less than or equal to 1) of being included in the sensory representation of that stimulus. When the probability
A Unification-based Model of Perceptual Learning in Reading

not take place. The system does not need to be told what the stimulus is because the system has a match. Exposures 2-7, therefore, each represent a successful recognition of the stimulus "A" and, in addition, demonstrate the progressive elaboration of the "A" schema, culminating in the fully specified schema following the seventh exposure.

Elaborative learning occurs because pattern matching by unification allows perceptual schemata to be modified simply by virtue of their being used. With the second exposure of the letter "A," PERL again randomly generates a sensory representation of the stimulus. It is highly likely, however, that the second sensory representation generated for the stimulus "A" will include features other than those included in the first representation. When the second representation is matched with the existing schema, however, the new features sampled in the second sensory representation are substituted for the variable placeholders in the schema that represents the letter "A." According to the concept of elaborative learning developed in the PERL model, recognition is not necessarily a passive process that leaves perceptual schemata unchanged. The concept of elaborative learning suggests that the traditional distinction between recognition and learning is inappropriate, especially for subjects whose knowledge is incomplete (e.g., beginning readers). According to the PERL model, letter and word acquisition is a gradual phenomenon that begins with a relatively simple perceptual schema that must be elaborated through repeated exposures before learning is complete. Random associative learning provides a base, but elaborative learning will usually be required to complete a subject's schema for a stimulus. Perceptual learning is not stable and cannot be said to be "complete" until every feature present in a stimulus is represented in the cognitive code that represents that stimulus.

**Table 1**

Acquisition of a Single Letter (A) Over Seven Successive Exposures

<table>
<thead>
<tr>
<th>Number of exposures &amp; (features known)</th>
<th>Perceptual Schema</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 (5)</td>
<td>[f0, f1, f2, A, B, C, D, E, F, G, H, a11, i, a13]</td>
</tr>
<tr>
<td>2 (6)</td>
<td>[f0, f1, f2, A, f4, B, C, D, E, a9, F, a11, a12, a13]</td>
</tr>
<tr>
<td>3 (9)</td>
<td>[f0, f1, f2, f3, f4, A, B, C, D, E, a9, F, a11, a12, a13]</td>
</tr>
<tr>
<td>4 (10)</td>
<td>[f0, f1, f2, f3, f4, A, B, C, D, E, a9, F, a11, a12, a13]</td>
</tr>
<tr>
<td>5 (12)</td>
<td>[f0, f1, f2, f3, f4, A, f6, a7, B, e9, a10, a11, a12, a13]</td>
</tr>
<tr>
<td>6 (13)</td>
<td>[f0, f1, f2, f3, f4, A, f5, f6, a7, B, e9, a10, a11, a12, a13]</td>
</tr>
<tr>
<td>7 (14)</td>
<td>[f0, f1, f2, f3, f4, a5, f6, a7, f8, a9, a10, a11, a12, a13]</td>
</tr>
</tbody>
</table>

*Note:* According to the adopted font the letter A is represented by the feature set {f0, f1, f2, f3, f4, a5, f6, a7, f8, a9, a10, a11, a12, a13}. Features not represented in the schemata above are indicated with the use of logical variables (upper-case letters).

The first exposure to the letter “A” corresponds to an instruction phase in the trial. Since the system starts out without any perceptual schemata, it is apparent that the system cannot possibly recognize the stimulus on the first exposure; the system is simply "told" that the stimulus is the letter “A.” Having been told that the stimulus is the letter “A,” the system then associates the label “A” with the features that have been randomly sampled from the original input. The resulting perceptual schema for the letter “A” is depicted in Table 1 at the top of the column on the right. The learning that results from this first exposure is referred to as random associative (RA) learning since it represents the acquisition of an association between the label “A” and a randomly sampled set of features.

It is important to note that the schema for “A” after the first exposure is not complete. Of the 14 features that make up the letter “A,” only 5 are represented in the initial schema. Those features are the ones that have been randomly selected by the system in the course of generating the sensory representation. Features not represented in the schema have their slots held by variables (upper-case letters).

Once a schema for the letter “A” has been added to the PR of the system, subsequent exposures to “A” generate sensory representations that can be matched with the existing schema. When the system can find a match for an input, random associative learning does not take place. The system does not need to be told what the stimulus is because the system has a match. Exposures 2-7, therefore, each represent a successful recognition of the stimulus “A” and, in addition, demonstrate the progressive elaboration of the “A” schema, culminating in the fully specified schema following the seventh exposure. Elaborative learning occurs because pattern matching by unification allows perceptual schemata to be modified simply by virtue of their being used. With the second exposure of the letter “A,” PERL again randomly generates a sensory representation of the stimulus. It is highly likely, however, that the second sensory representation generated for the stimulus “A” will include features other than those included in the first representation. When the second representation is matched with the existing schema, however, the new features sampled in the second sensory representation are substituted for the variable placeholders in the schema that represents the letter “A.”

**Trial 2: The Effect of Misidentifications on Letter Learning**

Trial 2 demonstrates how misidentifications influence the basic perceptual and learning mechanisms of the PERL system. For the sake of both simplicity and clarity, the initial perceptual repertoire (PR) of the system is again empty; i.e., the system starts out without any letter knowledge whatsoever.

Target letters in the trial include “A,” “E,” “I,” “O,” and “L.” The instruction phase of trial 2 consists of a single exposure to each of the five target stimuli. The outcome of the instruction phase is a random associatively generated perceptual schema (or code unit) for each of the five letters. Following the instruction phase, the system is provided with a practice phase of randomly counterbalanced exposures to the stimulus set. Selected output is presented in Table 2. The feature sampling rate of the model in Trial 2 is 40%.

As before, schemata acquired by the system are represented by Prolog code units. Note also that the resulting perceptual schemata are still largely incomplete after the first exposure, although they are complete since no errors of identification have occurred. All 5 code units, therefore, satisfy the definition for a code: code that is incomplete and correct, and are reported as such in the output.

Obviously, the average rate of feature sampling specified for a trial exerts a powerful influence on the performance of the PERL system. In the context of the instruction phase, this influence is restricted to the relative completeness of learning outcomes since errors cannot occur in this phase. When the trial moves into the practice phase where errors can occur, however, the influence of the average feature sampling rate is evidenced not only in the relative completeness of learning but in the correctness of learning outcomes as well.
### Table 2
Selected Output from Trial 2

<table>
<thead>
<tr>
<th>Output following 1 instructional exposure</th>
</tr>
</thead>
<tbody>
<tr>
<td>ic_code(u,[[f0,f1,a2,f3,a,B,a7,D,E,F,a11,0,G,H]],0).</td>
</tr>
<tr>
<td>ic_code(o,[[A,B,C,F3,F4,D,E,F,G,a9,H,a12,a13]],1).</td>
</tr>
<tr>
<td>ic_code(a,[[A,a,B,C,a4,f5,D,E,F,G,a11,a12,a13]],1).</td>
</tr>
<tr>
<td>ic_code(e,[[f0,f1,a2,f3,a,B,a7,D,E,F,G,a11,a12,a13]],1).</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Output following 4 randomly counterbalanced practice exposures</th>
</tr>
</thead>
<tbody>
<tr>
<td>cc_code(u,[[f0,f1,a2,f3,f4,f5,a6,a7,a8,a9,a10,a11,a12,a13]],1).</td>
</tr>
<tr>
<td>cc_code(o,[[f0,f1,f2,f3,f4,f5,a6,a7,a8,a9,a10,a11,a12,a13]],1).</td>
</tr>
<tr>
<td>ic_code(a,[[f0,f1,a2,f3,f4,f5,a6,a7,a8,a9,a10,a11,a12,a13]],1).</td>
</tr>
<tr>
<td>error_match(u,o,1).</td>
</tr>
<tr>
<td>error_match(a,o,1).</td>
</tr>
<tr>
<td>error_match(e,o,1).</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Output following 19 randomly counterbalanced practice exposures</th>
</tr>
</thead>
<tbody>
<tr>
<td>cc_code(u,[[f0,f1,a2,f3,f4,f5,a6,a7,a8,a9,a10,a11,a12,a13]],1).</td>
</tr>
<tr>
<td>cc_code(o,[[f0,f1,f2,f3,f4,f5,a6,a7,a8,a9,a10,a11,a12,a13]],1).</td>
</tr>
<tr>
<td>cc_code(a,[[f0,f1,a2,f3,f4,f5,a6,a7,a8,a9,a10,a11,a12,a13]],1).</td>
</tr>
<tr>
<td>ci_code(i,[[a0,a1,a2,a3,a4,a5,f5,f6,a7,a8,a9,a10,a11,a12,a13]],1).</td>
</tr>
<tr>
<td>error_match(o,u,4).</td>
</tr>
<tr>
<td>error_match(i,o,1).</td>
</tr>
</tbody>
</table>

**Note**: error_match(X,Y,n) indicates the number of times n that letter X is misidentified as letter Y.

---

A Unification-based Model of Perceptual Learning in Reading

Output following the first 4 practice exposures (exposures 2-5) is depicted in Table 2B. Practice exposures differ from instructional exposures in one important way: In practice exposures, recognition errors can occur. In Trial 2, the error recognition parameter in the system is adjusted so that only 40% of all naturally occurring errors will be detected and corrected. In other words, for every 10 possible errors, the system will, on average, only recognize and self-correct four of them. Undetected errors, however, influence subsequent learning and can lead to inappropriate (i.e., incorrect) learning, as is demonstrated in the output listed in Table 2.

Three differences between the instruction and practice exposure outputs are worthy of note. One difference is the presence of multiple perceptual schemata for letters following practice exposures. A second difference is the presence of error data (error_match) in the output. The third difference is the presence of incorrect code units as represented by ic_code (incomplete and incorrect) or ci_code (complete and incorrect) units.

According to the PERL model, a sensory representation of a stimulus is generated according to the average sampling rate specified by one or more sampling parameters. When a sensory representation has been generated, the PERL system begins a sequential search through the PT of the system in an attempt to match the sensory representation with an existing perceptual schema. If a sensory representation is less than completely specified, however, the system may match that sensory representation with an incorrect perceptual schema. In fact, a sensory representation can match any schema with which it is featurally compatible. An icon and a schema are featurally compatible as long as the feature sets occupying corresponding positions do not include incompatible features (where for any n ≤ 14, fo and an are incompatible features, and variables match either presence or absence features or other variables).

The errors introduced to the PERL system by misidentifications, however, are not limited to being expressed as recognition errors alone. The reason is that when an incorrect pair are matched, elaborative learning may pass on features to the cognitive unit that are not correct, thus leading to the inappropriate elaboration of perceptual schemata. Recognition errors can, therefore, have a destabilizing influence on a perceptual repertoire.

The events that lead to misidentifications provide important clues to understanding how the practice exposure output depicted in Table 2 differs from the output of the instructional exposure. In fact, all three of the noted differences are consequences of misidentifications and PERL's attempts to recover from the errors introduced to the system by those misidentifications.

Multiple perceptual schemata for both "e" and "i" demonstrate PERL's strategy in attempting to recover from inappropriate elaboration. At some point during exposures 2-5, both "e" and "i" were involved in misidentifications that resulted in inappropriate elaborative learning and incorrect schemata. Note, however, that not every misidentification results in inappropriate elaborative learning. Although every letter is involved in at least three misidentifications, only "i" and "e" result in incorrect code units (i.e., ic_code units).

At the conclusion of exposures 2-5 (Table 2B), two letters (uo) have been learned completely and correctly (represented by cc_code), two letters (ao) have been learned incompletely and correctly (represented by ic_code), and one letter (i) is both incomplete and incorrect (represented by ci_code). From this point onward, even though misidentifications of "u" and "v" can still occur (since feature sampling < 1), there is no longer the danger that the "u" or "v" units will be inappropriately elaborated since both of these units are complete and, therefore, stable. The incomplete and correct "a" and "e"
units can still result in misidentifications and are also still subject to inappropriate elaboration (these units represent untestable perceptual schemata). Lastly, the incomplete and incorrect "t" and "c" units may continue to be elaborated or may be forgotten by assuming an inaccessible position in the PR (which the incorrect "e" unit has already done).

Selected output following 20 exposures (1 instructional and 19 practice) are presented in Table 2C. Note that the PR has stabilized (i.e., all perceptual schemata are complete) but that the letter "t" is still not correctly represented even after the 20th exposure.

It may seem surprising that the PERL system is capable of any learning at all when highly confusable stimuli are used, the rate of feature sampling is set as low as .04, and misidentifications are allowed to occur. In fact, PERL's tendency to converge on correct representations of stimulus over the course of as few as 15 or 20 practice exposures is quite consistent, even under these conditions. The reason is that PERL's misidentifications usually result in less useful code than correct identifications, especially as PERL's PR becomes more complete. With each incorrect elaboration of a code unit, the likelihood that the schema will be compatible with future sensory representations decreases. Moreover, when none of the code units in the PR are compatible with an input, the system returns to a random associative strategy. In effect, when it does not have a reasonable guess, it simply "asks" what the stimulus is and it is "told." When it is told, it generates a new schema using the same random associative strategy applied in the instructional phase of the trial. Since the new unit is at least correct, it is more likely to be matched than its incorrectly elaborated predecessor. Code units that are incorrectly elaborated, therefore, tend to "sink" to the bottom of the PR and ultimately are "forgotten." The old schemata are forgotten in a manner similar to the forgetting implemented in the Fiegenbaum and Simon (1984) EPAM model, with forgotten units subsuming to positions in the PR that are no longer accessible to the PERL system, despite their continued presence in the PR.

**Trial 3: Word Learning/Recognition and the Parsing Problem**

Trial 3 demonstrates some of the basic word learning and recognition mechanisms of the PERL system under experimental conditions that, like Trial 2, allow misidentifications during learning. A second goal of Trial 3 is to demonstrate how the PERL model solves a major problem in the theory of word recognition commonly referred to as the "parsing problem" (Adams, 1979).

Trial 3 simulates the learning of five words: "cat," "tat," "sin," "saw," and "cat." For the sake of simplicity and clarity, the initial PR of the system is empty. The instruction phase of Trial 3 consists of a single exposure to each of the five target stimuli. The outcome of the instruction phase is a random asymmetrically generated schema for each of the five words. Following the instructional phase, the system is provided with a series of randomly counterbalanced practice exposures to the stimulus set. Selected output from the instructional phase of Trial 3 is presented in Table 3. Feature lists of word code units consist of letter-string feature lists within a higher level word feature list. For example, the word "CAT" could be represented as:

```plaintext
code('CAT', [f1, f2, f3, a1, a2, a3, a4, a5, a6, a7, a8, a9, a10, a11, a12, a13], [80, a1, a2, f1, f2, a3, a4, a5, a6, a7, a8, a9, a10, a11, a12, a13], [0, f1, f2, f3, f4, a5, a6, a7, a8, a9, a10, a11, a12, a13], [80, a1, a2, f1, f2, a3, a4, a5, a6, f7, a8, a9, a10, a11, a12, a13], [80, a1, a2, f1, f2, a3, a4, a5, a6, a7, a8, a9, a10, a11, a12, a13]], 1).
```

where each letter position is represented by a specific feature list. Feature list output depicted in Table 3 is abbreviated ([ ] ) in order to avoid crowding the table. As before, prefixes of [i0] [i1] [i2] [i3] [i4] and it identify code units that are, respectively, incomplete and incorrect, incomplete and correct, and complete and incorrect, and complete and correct. Feature sampling in Trial 3 is position dependent; the sampling rate depends on the position of the letter in the word. Sampling rates at initial, medial, and final letter positions conform to the relation SP1 ≤ SF1 ≤ SP2, where SF1, SF2, and SP1 specify sampling rates at initial, final, and medial character positions, respectively. Position dependent feature sampling has been adopted because numerous empirical studies (Bruner, O'Dowd, 1958; Eriksen & Robbaugh, 1970; Estes, Allmeyer, & Reder, 1976; Humphreys, Quinlan, & Evett, 1983; McCusker, Gough, & Brias, 1983; Rumelhart & McClelland, 1982) have suggested that position in a word does, in fact, influence the perceptibility of letters. Initial letters tend to be most perceptible, and medial letters tend to be least perceptible, resulting in the perceptibility relation adopted (SP1 ≤ SF1 ≤ SP2).

One of the most obvious differences between the output from Trial 3 and that of previous trials is the presence of the new output measure "parse code" that indicates how words are segmented or parsed prior to recognition by the PERL system. The output depicted in Table 3A indicates that the instructional exposures to "saw," "bin," and "cat" result in whole-word pairings ([ [saw]], [[bin]], and [[cat]]). In other words, "saw," "bin," and "cat" are learned as unsegmented visual patterns or sight words.

Table 3A also indicates, however, that PERL's first exposure to "ore" and "eat" results in segmented orthographic patterns. Both of these words are acquired as schemata that are subdivided into two units: a first letter and two remaining letters (for "ore") or a first letter and two remaining letters (for "eat"). How PERL manages to do this and why the parsing problem arises in the first place are two important questions that will be addressed below. Before considering these questions, however, it will be useful to review the parsing problem.

Adams (1979) has noted that the problem of parsing arises for any theory that posits units of analysis between letters and whole words. Letter-based and whole-word theories of word recognition avoid the parsing problem by assuming that there is one, and only one, possible parsing for a word. According to letter-based theory, words are parsed into strings of individual letters. Thus, by letter-based accounts, the only admissible parsing of a word is the one ordered letter string that corresponds to that word. Whole-word accounts of word recognition, on the other hand, make the claim that words are not parsed at all, but are recognized as whole units rather than as segmented strings.

The position adopted in the PERL model is one of great flexibility, accommodating both the letter-based and whole-word positions. According to the PERL model, words may be recognized either as whole units or as segmented strings, depending on a number of circumstances. Since the PERL model admits units of analysis between letters and words, however, it is apparent that the parsing problem noted by Adams must be addressed. Essentially, the problem is this: If a word can be identified either as a single unit or as any one of a number of strings of sub-word units, how does the system decide which parsing to apply?

For example, consider the target "cat." According to the PERL model, this three-letter word has four possible pairings: [[cat]], [[c][at]], [[ca][t]], and [[c][a][t]]. In principle, any one of these four pairings could be applied in the recognition of this stimulus. The parsing problem is specifying how one parsing is selected from the set of possible pairings.
Table 3
Selected output from Trial 3

A. Output following 1 Instructional exposure

code(saw\_[\_\_1],
parse(saw[1].

B. Output following 24 randomly counterbalanced practice exposures

code(one\_[\_\_1],
parse(one[1].

code(eat\_[\_\_1],
parse(eat[1].

code(bin\_[\_\_1],
parse(bin[1].

code(eat\_[\_\_1],
parse(eat[1].

code(bin\_[\_\_1],
parse(bin[1].

C. Output following 64 randomly counterbalanced practice exposures

code(saw\_[\_\_1],
parse(saw[1].

code(one\_[\_\_1],
parse(one[1].

code(eat\_[\_\_1],
parse(eat[1].

code(bin\_[\_\_1],
parse(bin[1].

Note: parse and error_match counts are based on one exposure in 3A and on the last five exposures of each stimulus for 3B and 3C.

A Unification-based Model of Perceptual Learning in Reading

The solution to the parsing problem is based on a simple principle that turns out to have interesting results when applied to a system with a sensory apparatus like PERL's. The principle is a variation of Zipf's (1949) principle of least effort. In the present circumstances, the principle of least effort means that PERL's perceptual system attempts to recognize words in the simplest parsing available, where the simplicity of a parsing is determined by the number of pieces that make up the parsing. According to the principle of least effort, the set of possible parsings of the word "eat" can be divided into three categories composed of one of the (eat), two of the (eat.[]), and three of the (eat [[]]). The first attempt at recognition will be based on the simplest whole-word parsing (eat).

Obviously, however, the system's attempt to recognize "eat" and "one" according to the whole-word parsing has failed, and it raises two important questions. One question is why the whole-word parsing failed. The second question is why the system selected between the two possible parsings composed of two units. The principle of least effort alone cannot answer either of these two questions. In order to understand the performance of the model, some further details of PERL's sensory system must be considered.

One parameter in the model that has not been referred to in the trials discussed to this point is the chunk parameter (CP). CP defines the maximum number of features that can make up a sensory chunk, where the term "chunk" is used as the sensory equivalent to the concept of a chunk as proposed by Miller (1950) in Explaining short-term memory. The value assigned to CP (i.e., 9) is also deliberately selected to be consistent with Miller's estimation (7 ± 2) of the number of chunks that can be accommodated by human subjects.

By defining a maximum number of features that can make up a sensory chunk, the system is provided with a basis for parsing. Recall that the generation of sensory representations is assumed to be a random process, within the constraints imposed by sampling parameters. In the present trial, the sampling parameters applied (SPI = 0.5, SPM = 0.5, & SPI = 0.2) result in an overall sampling rate of 0.2 across all three letter positions. Since the total number of features in a three-letter word is 24, the average number of features sampled will be 8.4. As long as the total number of features sampled is less than CP (which has been set to the value 9), the stimulus can be treated as a single sensory chunk. When the stimulus is treated as a single sensory chunk, it may be recognized according to a whole-word parsing.

It is apparent, however, that there will be occasions when the number of features sampled will exceed the chunk parameter. When the number of features sampled exceeds the chunk parameter, the system cannot treat the input as a single sensory unit. Rather, the system must parse the sensory representation into two or more units. This is why the whole-word parsing for "eat" failed; the sensory sampling of this stimulus exceeded the chunk parameter.

But the second of the two questions remains: How does the system ultimately select a parsing? The whole-word parsing has failed. What now? To begin with, recall that the principle of least effort makes the claim that parsings are considered according to their relative simplicity. The simplest parsings remaining are those composed of two units (i.e., (eat[])) and (eat [[]]). The second question now becomes, how does the system decide which of these two parsings to try first?

The answer to the second question lies in the sampling characteristics of the human visual system. People do not appear to sample the visual information in words uniformly across letter positions. Numerous perceptual studies have been noted above have shown that more visual information is sampled from the beginnings and endings of words than from medial positions. These findings and the concept of lateral masking provide,
respectively, the empirical and theoretical rationales for position-dependent sampling parameters (SPI, SPn, & SPj) satisfying the relation SPn ≤ SPI ≤ SPj. An example will make clear how position-dependent sampling influences the probability that recognition will occur based on a given parsing.

Suppose, as is true of the “eat” example used above, that the whole-word parsing for a stimulus fails and the distribution of features sampled adheres to the position-dependent relation assumed by the PERL model. According to the principle of least effort, the failure of the whole-word parsing leads the system to attempt recognition based on a two-unit parsing. But when a three-letter word like “eats” is employed as a stimulus, there are two different two-unit parses that are possible (i.e., [[e][a]t], & [[ea][t]]). Since non-uniform sampling characteristics have been adopted, however, some parses are more likely to satisfy the chunking requirements of the system than are other parses, and the most likely parsing to satisfy the chunking requirements is the one that most uniformly distributes sampling across the parse unit.

For example, consider the “eat” example above. One parsing divides the word between the first and the second letters: [[e][a]t]. The absolute rate of feature sampling in the first parse unit is simply the sampling probability of the first letter position, SPI = 0.3. The absolute rate of feature sampling in the second parse unit, however, will be the sum of the sampling probabilities of both the second and third letter positions since both of these letters are included in the second parse unit. Thus, the absolute rate of feature sampling of the second parse unit is SPn(0.1) + SPI(0.2) = 0.3. Note that for the first parsing, the sampling distribution across parse units is uniform, (i.e., both are equal to 0.3).

Now consider the second possible parsing that divides the word “eat” between the second and third letters: [[ea][t]]. The absolute rate of sampling of the first parse unit is the sum of the sampling probabilities for the first two letter positions, given by SPI + SPn = 0.4. The absolute rate of sampling in the second parse unit is simply the sampling probability of the third letter position (0.2). Note that the sampling distribution across parse units in the second parsing is not uniform. Since sampling is not uniform, some units are likely to sample more features than other units, and the probability that the unit will fail to satisfy the chunking requirement is increased.

The parsing most likely to succeed is the parsing that most uniformly distributes absolute feature sampling across parse units, and that is how the PERL model proposes. The idea is that the PERL model proposes parsings are selected between two or more are possible. The principle of least effort provides a global criterion for ordering possible parsings into plausible least effort categories, and the interaction of position-dependent sampling and the chunking requirement leads to a natural probabilistic mechanism for selecting parsings on a more local level, within least-effort categories.

The PERL system thus solves the parsing problem with a single general principle (the principle of least effort) and the application of a limited number of simple sensory mechanisms that can plausibly be conjectured to be a part of the human visual system. It is also worthwhile noting that the solution to the parsing problem proposed above has a number of interesting implications for our understanding of syllabic structure in language. Among other things, it suggests that syllabification is at least as much a visual as it is an aural phenomenon, and it may provide a basis for explaining the relative frequency of syllabic structures on the basis of the relative uniformity of featural sampling across syllabic units in words. This suggestion is consistent with findings indicating that experience with the segmentation of written language may contribute to, or even be a requirement for, successful phonetic (i.e., aural) segmentation (Elfr & Wilce, 1980; Mann, 1986; Morris et al., 1986).

Returning to the output generated in Trial 3, it is apparent that many of the learning characteristics displayed in previous trials are also in evidence in this trial. As in Trial 2, misidentifications and elaborative learning result in incorrect and multiple schemata. Complete and correct schemata for two of the target stimuli (“orc” and “cat”) emerged after 25 exposures; a third (“saw”) appeared after 35 exposures; a fourth (“eat”) appeared after 40 exposures. As indicated by the data in Table 3C, one target (“bin”) is still not represented by correct code after 65 exposures.

Adequacy of the Model in Accounting for Observed Findings

The purpose of the previous section has been to demonstrate how the PERL model simulates various aspects of letter and word perception and learning. The purpose of this section is to demonstrate that the proposed model does, in fact, accommodate the existing empirical findings that have been identified.

Word Superiority Effects (WSE) and the Pseudoword Effect (PSE)

The first set of findings to be considered include both word superiority and pseudoword effects. These findings will be treated together since the PERL model proposes a single processing mechanism for words and pseudowords (and nonwords as well). Two kinds of WSE have traditionally been noted: the “Cattell effect,” and the word-letter effect. The Cattell effect refers to the finding that words are recognized (i.e., named) as fast as or faster than individual letters. The word-letter effect refers to the enhanced perceptibility of letters in words over letters either in isolation or in orthographically irregular contexts. The pseudoword effect refers to the enhanced perceptibility of letters in pseudowords over letters either in isolation or in orthographically irregular contexts.

Two features of the PERL model are central to understanding word and pseudoword effects. One feature is the orthographic parsing mechanism that has been adopted. The second feature is that, according to the PERL model, individual letters in multi-letter sequences are usually identified post-perceptually on the basis of the larger parse units (whole words or syllable-like units) employed in identification. A reaction-time trial will make the influence of both of these features more apparent.

Table 4 depicts selected output from a reaction-time experimental trial. Reaction-time trials differ from simple accuracy trials by providing measures of the processing time required by the system to identify each input. The error recognition parameter in Trial 3 has been set to preclude errors from occurring during this trial in order to avoid complications. The data generated during this trial should, therefore, be considered to simulate correct-response latency data. Another important feature of this trial is that the PERL system's ability to learn has been turned off in order to keep reaction-time output as simple as possible.

Processing times are indicated by Prolog predicates that take the form process time(X,Y), where X represents both the input and the parsing employed in recognizing that input, and Y is a list of two elements representing processing time in seconds and hundreds of seconds. Processing times are provided by a hardware-driven system clock.
Table 4
Selected reaction-time trial output

<table>
<thead>
<tr>
<th>Type</th>
<th>Reaction-Time Trials</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single letter data</td>
<td>process_time([t],[0.33])</td>
</tr>
<tr>
<td></td>
<td>process_time([x],[0.38])</td>
</tr>
<tr>
<td></td>
<td>process_time([r],[0.33])</td>
</tr>
<tr>
<td>Word data</td>
<td>process_time([fed],[0.22])</td>
</tr>
<tr>
<td></td>
<td>process_time([jab],[0.29])</td>
</tr>
<tr>
<td></td>
<td>process_time([thin],[0.23])</td>
</tr>
<tr>
<td>Pseudoword data</td>
<td>process_time([bei],[0.60])</td>
</tr>
<tr>
<td></td>
<td>process_time([abi],[0.83])</td>
</tr>
<tr>
<td></td>
<td>process_time([fin],[1.15])</td>
</tr>
<tr>
<td>Nonword data</td>
<td>process_time([bip],[1.26])</td>
</tr>
<tr>
<td></td>
<td>process_time([bix],[1.65])</td>
</tr>
<tr>
<td></td>
<td>process_time([fin],[2.25])</td>
</tr>
</tbody>
</table>

Stimuli employed in the trial include individual letters, words, pseudowords, and nonwords. Exposure to targets was randomly counterbalanced. Reaction-time output is presented in letter, word, pseudoword, and nonword groupings to make differences between processing times more obvious. Processing times for individual letters range from 0.33 to 0.38 of a second. Processing times for words range from 0.22 to 0.39 of a second and are of the same magnitude as the times for individual letters. Note, however, that although the number of letters in the letter and word stimuli differ, the number of elements in the parsing employed do not. This is the key to understanding word and pseudoword effects according to the PERL model: processing time for an input is determined, in large part, by the level of parsing required for identification of that input to take place.

Parsing is a determinant of processing time in two ways. One way is by varying influences processing time that parsing itself requires time. Recall that the first parsing attempted is a whole-word parsing. If the whole-word parsing fails, however, the system must generate a two-unit parsing. If every two-unit parsing fails, then the system generates three-unit parsings. The more parse units in the parsing ultimately matched, therefore, the more time is taken up in generating parsings, thus resulting in higher processing times.

A second way the number of parse units influences processing time is in the number of searches through the PR required in order for every parse unit to be matched. A parsing made up of three units requires three searches through the PR of the system. A successfully matched parsing made up of a single unit will never require more than one.

According to the PERL model, words can be recognized as fast as individual letters because both words and letters can be recognized as single parse units. Word frequency effects have also been observed (Howes & Solomon, 1951; Johnston, 1978; McClelland & Johnston, 1978; Solomon & Postman, 1952) that can be accounted for by the contribution search time makes to the overall rate of processing. Every word or letter requires at least the time $T_p$ to carry out a single unit parsing. In addition to $T_p$, there is also a time requirement $T_s$ for the search through the PR. Code units that have been recently applied are near the top of the PR since schemata are moved to the top when they are used. Schemata near the top of the stack are, naturally, encountered sooner than code units at the bottom of the PR, since the search through the PR is top-down and sequential. The search time requirement $T_s$ will, accordingly, vary depending upon the position of the code in the PR. Since high frequency words are encountered more often, those words are more likely to be found near the top of the PR and are recognized faster than low frequency words. Another possible explanation for frequency effects compatible with the PERL model is that if cognitive code is conjectured to decay over time, if cognitive code is subject to memorial decay, it is possible that high frequency whole-word code units are forgotten between exposures, thus requiring subsequent recognition to rely on the slower processing of sub-word cognitive units in order to achieve recognition.

The enhanced perceptibility of letters in words over nonwords (i.e., the word-letter effect) is attributable to the assumption that letter readout occurs post-perceptually. Post-perceptual letter readout means that the identification of letters in known words involves an inferential component. What is recognized is the word, not the letters in it. The presence or absence of letters in the recognized input is inferred by the subject on the basis of the knowledge he or she has about the orthography of the recognized word. In the case of a nonword input, however, there is no means of correctly recognizing the input except on the basis of features detected. The subject is unable to apply any orthographic knowledge in making the decision about the presence or absence of letters in a nonword target. The enhanced perceptibility of letters in words, therefore, due to the knowledge subjects apply by making use of perceptual schemata in a recognition task.

The enhanced perceptibility of letters in pseudowords is accounted for in the same way. Although pseudowords require more complex parsing than nonwords, the parsing required by a pseudoword is simpler than that required by an orthographically irregular letter sequence. This means that although a subject might be unable to apply word knowledge, general orthographic knowledge can still be applied. General orthographic knowledge is represented by recognition units in a perceptual repertoire that are between whole words and individual letters. When the search through the PR turns up multi-letter units that can be applied in the recognition of the pseudoword, these multi-letter units provide a basis for inferential letter identification that augments the perceptual information provided by the sensory system. The pseudoword-letter effect can, therefore, be accounted for by the same kind of perceptual schemata proposed to explain the word-letter effect.

Visual Factor Effects (VFE)

Another set of findings to be considered in the empirical evaluation of the PERL model are visual factor effects. Specifically, it appears that when a target is indistinct and followed by a blank nonpatterned field, the observed word advantage over nonwords is greatly reduced, and the observed word advantage over individual letters is similarly reduced, or even reversed. The PERL model offers two different explanations for what are proposed to be two different sources of VFE outcomes. One explanation is based on a theoretically predicted interaction effect between the rate of feature sampling of the stimulus and word length that influences the overall rate of recognition errors occurring in the system. The second explanation is proposed to account for the preservation of the word advantage under the distinct target/pattern mask condition and is based on the time required by the system for parsing prior to cognitive matching and recognition.
A Unification-based Model of Perceptual Learning in Reading

The first explanation to be considered, hereafter referred to as the sampling/word length interaction effect, begins with the assumption that experimental conditions observed to lead to the elimination of the word advantage do so because they result in a lower rate of feature extraction for the target. An indistinct target, for example, is indistinct precisely because the features that compose it are degraded in some fashion. Since the overall rate of feature extraction is dependent upon a feature recognition mechanism, it is apparent that as the quality of the target declines, the proportion of features actually available to the recognition system likewise declines. Degrading a stimulus, therefore, leads to a lowering of the overall rate of feature extraction.

The effect of the blank mask is similarly assumed to result in a decline of overall rate of feature extraction. It seems reasonable to suppose, as do McClelland and Rumelhart (1981), that the effect of the blank mask condition is to reduce the contrast of the visual icon generated by the exposure of the target. The reduction of contrast, however, can reasonably be assumed to lead to a reduction in the detection of presence features by the feature detection system, and thus result in an overall decline in feature extraction.

The sampling/word length interaction effect is also, in part, a consequence of the fact that there are far more words than there are letters in the English language. As the rate of feature extraction approaches zero, the probability that the PERL system will correctly identify a word or letter approaches a lower limiting value of 1/x, where x represents the number of code units in the current PR with the same length as the item to be identified. For individual letters, it is apparent that the probability that a letter will be correctly identified never drops below a constant value equal to 1/26, or approximately 0.038. For a word of n letters, however, the number of possible matches is an exponential function $a^n$ where $a$ represents the number of letters in the alphabet.

In principle at least, the number of possible two-, three-, and four-letter words in English are 676, 17,576, and 455,976 respectively. Obviously, the number of possible words is more than the number of actual words, but the important point is that the actual number of two-, three-, and four-letter words increases with increasing word length, and therefore, as feature extraction rates decrease, the lower limiting probability that an n-letter word will be correctly identified likewise decreases. As a consequence, rate of probability of making a correct guess based on incomplete featural information should be observed to interact with word length, resulting in the gradual elimination of the word advantage that is typical of data reported in VIE studies.

The interaction effect that has been proposed, while it accounts for differences between bright target/patterned mask and degraded target/nonpatterned mask outcomes, does not help explain the persistence of the accuracy advantage for letters in words under the bright target/patterned mask condition. Three different explanations have been proposed to account for the word advantage under the bright target/patterned mask condition. One alternative is that letters in words are translated into nonmaskable form faster than are individual letters (Johnston & McClelland, 1973; Massaro & Klitzke, 1979). A second alternative is that words are transformed into representations that are less affected or completely removed from the influence of masking (Carr et al., 1978; Johnston & McClelland, 1973, 1982; McClelland, 1976). A third possibility is that some kind of top-down influence contributes to the recognition of letters in words that does not contribute to the identification of letters in nonwords (McClelland & Rumelhart, 1981).

According to the PERL model, the key to understanding the persistence of the word advantage is to note that the bright target/patterned mask condition requires a very brief interval between the onset of the stimulus and that of the mask in order to generate performances comparable in accuracy to those of the degraded target/nonpatterned mask condition. The implication is that the amount of time between stimulus onset and the onset of the mask is an important factor in the accuracy of performance between the bright target/patterned mask and the degraded target/nonpatterned mask conditions. The bright target/patterned mask condition seems to represent an "interrupt" condition that limits the rate of features extracted if a cognitive match has not been established before the mask appears. The degraded target/nonpatterned mask condition, however, seems to represent a condition where rate of feature extraction does not depend on an interrupt effect but, rather, depends on the average rate of feature extraction as determined by the extent of degradation of the stimulus and the related "decay" imposed by the blank nonpatterned field.

This interpretation of the persistence of the word advantage under the bright target/patterned mask condition is, in effect, a version of the thesis that letters in words are translated more quickly into nonmaskable form than are letters in nonwords (Johnston & McClelland, 1973; Massaro & Klitzke, 1979). This interpretation is, of course, consistent with the theory of sensory processing proposed in the present research. According to the PERL model, masking effects can be expected as long as processing is dependent upon sensory information. Once the sensory representation has been successfully matched with a cognitive representation, however, the performance is no longer subject to the influence of masking. The reason letters in words are less subject to interference by the bright patterned mask is that letters in words are translated into a nonmaskable cognitive form faster than are letters in nonwords. The reason words are transformed faster is simply because a word usually requires only a single cycle through the parsing routine in order to generate a nonmaskable cognitive representation. Nonwords, on the other hand, require multiple cycles through the parsing routine before being represented in a nonmaskable cognitive form. The difference. The difference in the number of cycles through the parsing routine is assumed to be directly related to the amount of time required to represent a stimulus in a nonmaskable cognitive form. The source of the word advantage under the bright target/patterned mask condition is, therefore, an artifact of orthographic parsing differences between words and nonwords.

Repetition Effects (RE)

Repetition effects refer to the enhancement of perceptibility that seems to occur when a stimulus is repeatedly presented. Some repetition effects are observed to apply to both words and pseudowords. For example, the speed and accuracy of identification of both words and pseudowords increases with repeated exposures (Feustel et al., 1983; Salasoo et al., 1985). Other repetition effects seem to affect words and pseudowords differently. One difference is the extent to which facilitation is "generalized" to orthographically similar stimuli. Pseudowords that have not been exposed before but are orthographically similar to previously exposed pseudowords show enhanced perceptibility. A word, however, that is orthographically similar to a word previously exposed does not show enhanced perceptibility (Feustel et al., 1983). Both words and pseudowords are subject to a greater proportion of identification errors when orthographically similar stimuli are employed (Feustel et al., 1983).

Another interesting word/pseudoword difference concerns the influence of repeated exposures on the recognition advantage words usually show over pseudowords. Fewer than five repeated exposures typically preserves the word recognition advantage over pseudowords, but by the fifth or sixth exposure, the word advantage usually disappears after
A Unification-based Model of Perceptual Learning in Reading

J.E. McEneaney

this point, the pseudowords repeatedly exposed are recognized as fast and as accurately as words (Feustel et al., 1983; Salasco et al., 1985). Moreover, the enhanced perceptibility of pseudowords exposed five or more times has been shown to persist over periods of time as long as a year (Feustel et al., 1983; Salasco et al., 1985), whereas the enhancement of word perceptibility that results from repetition is observed to decline over time such that after a year the repeatedly exposed pseudowords and words are recognized with the same speed and accuracy (Salasco et al., 1985).

Feustel, Shiffrin, and Salasco (1985) and Salasco, Shiffrin, and Feustel (1985) have attempted to account for the various repetition effects by suggesting that episodic memory plays a role in the recognition of words. According to the episodic account, the recognition performance of subjects is enhanced by the availability of episodic recall for particular exposures of the word or pseudoword repeatedly exposed. The Salasco et al. episodic recall theory, however, suffers from a number of limitations including a relatively large number of free parameters and a theoretical apparatus that seems rather data-specific, and in certain respects, only loosely tied to the basic concepts and mechanisms typical of episodic memory theory (Johnston, van Santen, & Hale, 1985).

According to the PERL model, however, there is no need to postulate episodic mechanisms in order to account for the observed repetition effect findings; the same processes that account for recognition generally can be shown to lead to repetition effects as well. The explanation adopted is quite similar to the temporary strengthening of permanent semantic memory codes proposed in several earlier studies (Forbach, Stanislaw, & Hochhaus, 1974; Morton, 1969, 1979). The thesis proposed in the PERL model is that repetition effects are a levels-of-activation phenomenon that is implemented in the PERL system by the sequential order of the schemata that make up the PR employed by the system.

When a word is activated, the activation level for the schema representing that word increases. This increase in activation level is simulated in the PERL model by the ordinal reorganization of the PR. When a perceptual schema has been matched, it is moved from wherever it is to the top of current PR. It is apparent that following the first exposure, subsequent attempts to recognize the same stimulus will require less search time through the PR since the needed unit is now already somewhere near the top. Repetition effects that lead to enhanced perceptibility are, therefore, explained in a manner quite similar to that employed with frequency effects. The difference between repetition and frequency effects is simply the scale of the phenomenon. Repetition effects reflect more local influences (i.e., recent exposures), whereas frequency effects are associated with more global linguistic influences.

When a pseudoword is recognized, however, at least two (and possibly more) schemata are activated. This is what distinguishes word recognition from pseudoword recognition in the PERL model: the number of elements that make up the parse-unit. With the activation of the multiple elements that make up a pseudoword parse-unit, it is apparent that more than one schema will be rewritten to the top of the PR. In fact, every unit that is an element in the pseudoword parse will be moved. If, for example, the pseudoword “PRENATURE” was successfully recognized according to the three-element parse unit [[PRE][NA][TRE]], then all three schemata would be rewritten to the top of the PR.

Orthographically similar words fail to result in any facilitation effect because, at the word level, schemata are independent of one another. Repeated exposure to the word “BONE” fails to lead to a decrease in correct-reaction latency of the word “CON”; because the only word unit that can lead to successful recognition is the specific word schema that represents “CONE”, the schema that represents the word “BONE” may lead to fast misidentification (i.e., “BON” for “CON”), but it will not be correct. Orthographically similar words, thus, tend to increase the proportion of recognition errors that occur but fail to lead to any facilitation effect because word recognition is usually based on the recognition of single units.

Orthographically similar pseudowords, on the other hand, may allow one or more of the previously applied schemata that are now at or near the top of the PR to be applied again. For example, suppose the pseudoword “PRENATURE” were exposed following repeated exposures of the pseudoword “PRENATIVE.” If the parse employed in recognizing the pseudoword “PRENATIVE” is the three-element one described above ([[PRE][NA][TRE]]), two parse elements are available ([PRE] and [NA]) that may be applied in the recognition of the new orthographically similar pseudoword “PRENATURE” via the parsing [[PRE][NA][TRE]]. Orthographic similarity enhances the perceptibility of pseudowords because pseudoword recognition employs multi-element parse units. With the use of multi-element parse units, orthographic dissimilarity (which leads to recognition errors) can be isolated, and previously applied units now at or near the top of the PR can be applied again, with a concomitant decline in overall search time. The successful recognition of a pseudoword will, according to the PERL model, require little more search time than that required to find the dissimilar units still buried in the PR, since one or more of the needed parse units will probably be near the top of the PR.

Accounting for the elimination of the word advantage, likewise, does not require any additional mechanisms in the PERL model. The elimination of the word advantage can be attributed to the model’s inherent preference for whole-word recognition. If a stimulus is presented frequently enough, that stimulus will eventually be acquired as a single whole-word unit. The elimination offered by the PERL model of the disappearance of the word advantage is simply that the number of repetitions required for the advantage to disappear is the same number required in order for the pseudoword stimulus to be acquired as “word” units. According to the PERL model, the word advantage does not disappear; rather, it is the pseudoword disadvantage that disappears with repeated exposures as a result of the acquisition of whole-word representation for pseudowords.

The explanation offered above is also consistent with the finding that the enhancement of perceptibility of pseudowords does not decline below the word level, even after periods of time as long as a year (Salasco et al., 1985). Apparently, the number of exposures required in order to eliminate the word advantage is also enough to result in single-element pseudoword codes that are, for all practical purposes, equivalent to permanent real-word units.

The PERL model, thus, seems capable of explaining a variety of repetition findings that have been reported in the literature. The general increase of speed and accuracy of recognition with repetitions and the decline in accuracy of both words and pseudowords with the use of orthographically similar stimuli are both attributable to general processing mechanisms in the PERL model. Observed word/pseudoword repetition effect differences are explained by basic processing differences between words and pseudowords and by general learning mechanisms that result in pseudowords achieving “word” status as a consequence of repeated exposures.

Conclusions and Summary

The model proposed in this paper has a number of distinctive characteristics that recommend it as a conceptual framework for research in letter and word perception. One
characteristic is the degree to which the model has been formalized, allowing it to be implemented as a computer simulation. A second characteristic is the wide variety of empirically observed effects for which it seems to be able to account, including word superiority effects, pseudoword effects, masking effects, and repetition effects.

A third distinctive characteristic of the PERL model is the emphasis it places on developmental and instructional issues that theories of letter and word perception have traditionally failed to address. The model proposed in this paper makes specific claims about how perceptual schemata are acquired and elaborated. Moreover, since the model functions as a simulation, instructional issues can be explored through controlled trials; the model can, in effect, function like an instructional laboratory where methods and materials can be tried out. A series of simulations not described in this paper, for example, suggest that learning can be enhanced by controlling the confusability of the words to be learned during instructional sessions. When highly confusable words are taught in the same session, the perceptual schemata learned by the system tend to be destabilized by misidentification and attendant mis-elaboration. According to the model, optimal learning occurs when words to be learned are least confusable and enough practice trials are provided to stabilize the schemata that are acquired. Other theories of letter and word perception have little, if anything, to say about the developmental and instructional issues that are of major importance in reading education.

In summary, the PERL model is a feature-based theory of letter and word perception. The model has been developed as a Prolog computer simulation based on the formal foundations described in Part 1 of this paper. The model demonstrates the application of powerful new theoretical tools and methods recently developed by researchers in the field of artificial intelligence and appears to satisfy basic theoretical and empirical standards. The PERL model also makes model-specific predictions about letter and word learning that may be of interest to reading researchers and practitioners.

References


A Unification-based Model of Perceptual Learning in Reading


