C E N T E R   F O R   R E S E A R C H   I N   L A N G U A G E

November 1988 **Vol. 3, No. 2**

The monthly newsletter of the Center for Research in Language, University of California, San Diego, La Jolla CA 92093. *(619) 534-2536*; electronic mail: *crl@amos.ling.ucsd.edu*

• • •

CONTENTS

EDITOR'S NOTE

This newsletter is produced and distributed by the **CENTER FOR RESEARCH IN LANGUAGE**, a research center at the University of California, San Diego, which unites the efforts of researchers in such disciplines as Linguistics, Cognitive Science, Psychology, Computer Science, Communication, Sociology, and Philosophy, all of whom share an interest in language. We regularly feature papers related to language and cognition (1 - 10 pages, sent via email) and welcome response from friends and colleagues at UCSD as well as other institutions. Please forward correspondence to:

Teenie Matlock, Editor
Center for Research in Language, C-008
University of California, San Diego 92093
Telephone: (619) 534-2536
Email: *crl@amos.ling.ucsd.edu*

SUBSCRIPTION INFORMATION

If you are currently receiving a hardcopy of the newsletter only and have an email address to which the newsletter can be sent, please forward that information to CRL.

Can we send you the email version only to save printing and mailing costs? If you require a hardcopy in addition, please request it and we will be happy to send you one.

If you require the unformatted nroff version of the newsletter, please request it from CRL after you have received the current regular formatted version.

If you know of others who would be interested in receiving the newsletter, please forward the email or postal mailing address to CRL. Thank you.

BACK ISSUES

Back issues of this newsletter are available from **CRL** in hard copy as well as soft copy form. Papers featured in previous issues include the following:

The Cognitive Perspective
Ronald W. Langacker
Department of Linguistics, UCSD
vol. 1, no. 3, February 1987

Toward Connectionist Semantics
Garrison W. Cottrell
Institute for Cognitive Science, UCSD
vol. 1, no. 4, May 1987

Dimensions of Ambiguity
Peter Norvig
Computer Science, UC Berkeley
vol. 1, no. 6, July 1987

Where is Chomsky's Bottleneck?
S.-Y. Kuroda
Department of Linguistics, UCSD
vol. 1, no. 7, September 1987
(2nd printing of paper in no. 5, vol. 1)

Transitivity and the Lexicon
Sally Rice
Department of Linguistics, UCSD
vol. 2, no. 2, December 1987

Formal Semantics, Pragmatics, and Situated Meaning
Aaron Cicourel
Department of Sociology, UCSD
vol. 2, no. 3, January 1988

Rules and Regularities in the Acquisition of the English Past Tense
Virginia Marchman
Department of Psychology, UC Berkeley
vol. 2, no. 4, April 1988

A Geometric Conception of Grammar
S.-Y. Kuroda
Department of Linguistics, UCSD
vol. 2, no. 5, June 1988

Harris and the Reality of Language
S.-Y. Kuroda
Department of Linguistics, UCSD
vol. 3, no. 1, September 1988

* * * * * * * * * * * * * * * *

Our distribution list has grown markedly this past year. Researchers interested in cognition and language throughout the USA as well as abroad now subscribe to the *CRL Newsletter*. Represented are countries such as Spain, France, Canada, Ireland, Italy, Israel, Japan, South Korea, Denmark, the Netherlands, West Germany, Poland, Australia, and Singapore.

• RESEARCH AND TRAINING PROGRAM IN NEURAL MODELLING FOR DEVELOPMENTAL PSYCHOLOGISTS •

Center for Research in Language
University of California, San Diego
La Jolla, California 92093

Following new discoveries in neural modelling and parallel computation, the field of cognitive science is now at a major turning point, a paradigm shift equivalent in magnitude to the information processing revolution of the late 1950's. The first computer revolution had unfortunate results for developmental psychology. First, the focus shifted away from learning, development and/or maturation into a fascination with basic principles of representation and retrieval in static systems that really do not learn well at all. Second, cognitive psychologists embraced the view that there is little or no relationship between hardware and software; as a result, research on cognitive processes did not take the human brain seriously as a source of constraints on theory. The new emphasis on parallel computation has resulted in a revival of interest in realistic models of human learning, and a commitment to learning theories that are inspired and constrained by our knowledge of the developing human brain. There is now an extraordinary opportunity for the development of new theoretical frameworks (e.g. the long-awaited successor to the Piagetian paradigm), and there are exciting new methods that cry out for application by developmental scientists who know what real human learning and development look like.

The Center for Research in Language at UCSD has just obtained a pilot grant from the John D. and Catherine T. MacArthur Foundation, to provide 5 - 10 developmental psychologists at any level (dissertation students through senior investigators) with short-term training in neural computation. The program has two goals:

(1) To encourage developmental psychologists in target interest areas (speech, language, early visual-motor and cognitive development, future oriented processes) to begin making use of connectionist modelling as a tool for evaluating theories of learning and change;

(2) To encourage greater use of realistic developmental data in the connectionist enterprise.

Our experience at UCSD suggests that a well-prepared and computer literate developmental psychologist can learn to make productive use of neural modelling techniques in a relatively short period of time, i.e. 2 weeks to 3 months, depending on level of interest and prior experience. Applicants may request training periods in this range at any point from 9/89 through 8/90. Depending on the trainee's needs and resources, we will provide (1) lodging at UCSD, (2) travel (in some cases), (3) access to SUN and VAX workstations with all necessary software, and (4) hourly services of an individual programmer/tutor who will supervise the trainee's progress through self-paced learning materials while assisting in the implementation of the trainee's proposed developmental project. Trainees are also welcome to attend seminars and workshops, and to consult with the relatively large number of faculty involved in connectionist modelling at UCSD.

Applicants are asked to submit 5 - 10 page proposals outlining a specific modelling project in a well-defined domain of developmental psychology. Criteria for evaluating proposals will include (1) the scientific merit and feasibility of the project itself (2) the applicant's computer sophistication and probability of success with short term training, (3) the probability that the applicant can and will continue working at the interface between neural modelling and developmental psychology (including access to adequate computer facilities at the applicant's home site). Applicants should indicate the preferred duration and starting date for the training program.

Applications should be submitted to Jeff Elman, Director, Center for Research in Language, University of California, San Diego, La Jolla, Ca. 92093. For further information, contact Jeff Elman (619-534-1147) or Elizabeth Bates (619-534-3007). Email inquiries may be sent to elman@ amos.ling.ucsd.edu or bates@ amos.ling.ucsd.edu.

A Connectionist Perspective on Prosodic Structure

Mary Hare David Corina Garrison W. Cottrell
Linguistics Psychology Computer Science
University of California, San Diego

I. Introduction

One area of interest in current phonological research is the representation and analysis of a language's prosodic structure. This research is concerned not with the minimal contrastive units, or phonemes, in isolation, but with higher levels of organization which govern these units in context. This research relies on representations such as the mora, syllable, or foot, and attempts to characterize well-formedness conditions on these representations and rules which maintain well-formed structures. It is an open question whether generative theorists take these rules and representations to be psychologically real. However, it must be noted that these representations are generally taken as given, and little attention is paid to their development. Furthermore, it is a fact that prosodic constraints exhibit a large amount of cross-linguistic variation. To simply take the necessary representations and constraints as given offers little insight on how they might be acquired or why this variation occurs.

In addition, the classification system on which these representations are based is presumed to be composed of stable, discrete classes of elements. This assumption leads to problems when entities fail to undergo processes which should apply to elements of their class. Linguists are forced to treat such cases by arbitrarily marking the exceptional element as outside the scope of the relevant rule.

The current paper takes a somewhat different approach to prosodic data. Here, prosodic structure is not a set of pre-existing representations related by rules, but the sum of the generalizations a speaker abstracts in the process of learning to produce samples of a language. These generalizations serve as constraints on language production, making the abstraction process an interactive one. The paper describes two Parallel Distributed Processing (PDP) networks developed to model this view of prosodic structure as an interactive process. In the models, the generalizations abstracted by the model are determined by the set of weights which the network develops in the process of producing correct output patterns. These are weights which are not built in, but must be learned. These weights serve as constraints, requiring outputs to conform as far as possible to the generalizations that have been induced. These, however, are soft constraints, and subject to continual modification through their interaction with the data. In addition, the weights are developed in the course of producing examples of the language in question: if the language were different, the network of weights (and thus the generalizations) would also be different. What is constant is the initial structure of the network that learns these weights, and the algorithm by which they are learned. This is not to say that back propagation is *the* algorithm or that our network is *the* network. Rather, our view is that linguistic universals are the result of the structure of the linguistic processor and the learning rule. If this approach is successful in explaining the linguistic facts, we would view it as more explanatory than a theory that simply tries to describe the universals that derive, in our view, from the above constraints.

In addition we show that it is possible to give a connectionist account of phonological data without simply implementing a generative analysis. Here we rely on the definition of implementation given by Alan Prince and Steve Pinker. They make the valid point that a network whose wiring, input representations, and so on are motivated by a rule-based, theoretical account of the process being modeled simply implements that account. The explanation of the phenomena still rests with the rule-based theory.

The simulations discussed in this paper do adopt certain representational features from linguistic theory. However, these adopted features are held to the minimum necessary to describe the phonemes which are the output of our model. They are not ideal, and we are currently exploring ways to make these features less theory-laden and more naturalistic. At the same time, we do not feel that the use of these representational features necessarily implies that the models are implementational. Instead, we feel strongly that if a model with a learning algorithm and embodying constraints from the processing power of the brain develops the proper behavior in the domain in question, this constitutes a better explanation than one that simply posits the rules and constraints the system is to follow.

The structure of the paper is the following. Section II serves as a brief introduction to the PDP approach to cognitive modeling, while Section III discusses the notion of prosodic structure and how various phonological alternations are analyzed in certain current linguistic accounts. Sections IV through VI describe the analysis of prosodic structure assumed in this paper, and the models of language production which demonstrate that approach.

II. Parallel Distributed Processing

Parallel Distributed Processing (Rumelhart and McClelland 1986a, McClelland and Rumelhart 1986) assumes that knowledge is represented by weighted connections spreading patterns of activation over large numbers of densely interconnected units. That is, these units transmit activation to other units over connections, which are weighted to allow the spread of activation to have either an excitatory or an inhibitory effect on the rest of the system. A network consists of input units, which respond directly to stimuli from outside the system, and output units, whose activation patterns represent the system's response to that input. In addition, there may be one or more levels of "hidden" or intermediate units. Each unit has an activation value, and this value is updated by weighing each incoming signal by the strength of the connection along which it is received, summing these weighted inputs, and passing them through some function to attain a new activation level.

Processing involves activating a set of input units; this activation spreads via the weighted connections across any hidden units to produce a pattern of activation on the output units. Learning by back propagation of error involves comparing this actual response of the system with the desired response supplied by a separate "teacher" input, and adjusting the connection weights according to a mathematical algorithm in a way that reduces the discrepancy between the actual and desired outputs.

Before presenting the PDP models which make up the bulk of this paper, we will examine the notion of prosodic structure in more detail.

III. Prosodic Structure

According to many current theoretical approaches to phonology, prosodic phenomena result from processes sensitive to the syllable structure of the language in question. Cross-linguistically there is variation in what is accepted as a well-formed syllable, but given a particular language there appear to be strict constraints on well-formedness. Briefly, the relevant terminology is as follows. Words can be exhaustively decomposed into syllables, with the syllable consisting of an obligatory vowel[1] (or *peak*) optionally preceded and/or followed by one or more consonants. Consonants which precede the syllable peak are referred to as the *onset*; those which follow compose the *coda*. Thus the forms a syllable may take include V, CV, CVC, CCVC, VC, CVCC, etc. All languages contain syllables of the form CV. Languages differ, subject to certain constraints, on which of the other syllabic possibilities they exhibit.

For ease of exposition we will speak of prosodic phenomena as falling into two major classes - those sensitive to syllable structure, and those sensitive to mora count, to be defined below. Naturally this is an oversimplification: the division is not clear-cut, and the two classes interact in complex ways.

A. Syllable Structure Processes

A variety of phonological alternations are analyzed as processes which exist to maintain the syllable structure of a language. For example, in Yawelmani (a dialect of the Yokuts language of California), no consonant cluster whatsoever is allowed to surface internal to a syllable. When such CC clusters arise at morpheme boundaries, as in 2(b), a vowel appears to break up the illicit cluster.[2] (data from Kuroda 1967)

```
(a) sent + al    ->  sen.tal
    smell  dubitive

(b) sent + hin   ->  se:.nit.hin
    smell  aorist
```

This phenomemon, where a vowel alternates with 0 in different forms of a word, is called *vowel epenthesis*, and is common cross-linguistically. However, this is only one means that a langauage might exploit to avoid CC clusters. Other possibilities include deletion of a consonant (as in French, Schane 1968), or reanalysis of a sonorant consonant as a vowel peak (Chichewa, a Bantu language).

B. Mora Count

Other prosodic phenomena are analyzed as processes sensitive to a notion of syllabic weight. The intuition being captured here is that for certain processes syllables of the form CV: and CVC often exhibit similar behavior, and this contrasts with the behavior of CV syllables.

The claim is that the syllable peak and the coda consonant contribute to what is referred to as the *weight* of a syllable. Phonological theory posits an abstract entity called the *mora*, a unit of syllabic weight. It is generally the case that the syllabic peak and the coda C count as one mora each, while a long vowel counts as two morae. The onset consonant contributes nothing to the mora count. Thus CVC and CV: syllables behave alike because both are *heavy*, or two-morae syllables. A CV syllable, on the other hand, contains only one mora, and is considered *light*.

A number of processes are viewed as conspiring to maintain the mora count of a syllable.[3] (Hyman 1984, McCarthy and Prince 1986). When an alternation in a language reduces mora count, the language will generally respond by compensating for that loss. Different languages choose different methods of maintaining mora count.

One common response to mora loss is compensatory lengthening of vowels (Hayes 1987). The data below is from the Bantu language Luganda. In this example, mora loss results from glide formation. When the V-initial suffix combines with the V-final stem, the stem vowel appears as a glide. Since this glide forms part of the syllable onset it no longer counts as a mora. The mora count of the word is maintained by the following vowel, which lengthens in this context.

Luganda (data from Clements 1986)

```
 m   m  m        m m m
 /|   | /|       /|\||\
mu + a.na   =>  mwa:.na
```

A variation on this response to mora loss involves consonant gemination. Ilocano is a language with no long vowels. It also exibits a process of glide formation. Here, the reaction to glide formation involves geminating (doubling) a consonant. In (A), the first half of the geminate consonant serves as coda to the preceding syllable, and so contributes a mora. In (B), on the other hand, gemination would lead to an unacceptable syllable structure, and so the process does not apply. The third mora in the representation is left without segmental content, and later deletes. This shows that although the language attempts to maintain a stable mora count it cannot always do so, given the resources at its disposal. This example illustrates two important of points. First, it demonstrates an alternate means of preserving mora count. Second, (B) demonstrates that although the pressure to maintain mora count exists, it functions as a soft constraint, and can be overridden by phonotactic constraints on syllable structure.

Ilocano (data from Hayes 1987)[4]

```
(A)
   m  m  m  m m                    m m m  m m
    | /| /|  \ |                 | || \/||
    a.do.bo + an =>  a.dob.bwan   (a.do  b.wan)
   meat dish

(B)
    m m m  m m     m mm m  m
    /|/ /|   |/     /|/    /|\ |
    lag. to + en =>  lag.    twe n    (*lagt.twen)
    jump
```

C. Linguistic Analysis

Theoretical accounts of the alternations outlined above rely on particular abstract representations: the CV template, the syllable, the mora. Prosodic processes are viewed as rules which maintain certain well-formedness conditions on these representations. For example, a formal treatment of the Turkish vowel epenthesis data involves associating a word of Turkish with a template corresponding to the well-formed syllables of that language. [Kornfilt 1986, Clements and Keyser 1983] If an abstract underlying representation of the word does not correspond to the form demanded by the template, that form is not allowed to surface. Instead, some process (here, vowel epenthesis) occurs to "save" the otherwise illicit form.

The example below demonstrates this approach. (4) shows the accusative form of a word, and the syllabification of the CV template with which it is associated[5]. Notice that since the accusative form is V-final, this form divides correctly into two acceptable syllables, and no change is required. (5) shows the same root, but in the nominative, or unaffixed, form. The first three segments of the word comprise a licit syllable, but the final *r* cannot combine with the *k* in the coda since Turkish exibits a constraint forbidding CS clusters in coda position. The *r* is left unattached. This is not a well-formed representation, and is unacceptable. The representation in (6), on the other hand, *is* well-formed. Here a second vowel has been epenthesized before the *r*, creating an acceptable syllable.[6] The form in (5) is assumed to be the underlying representation associated with this word, and serves as input to the vowel epenthesis rule which generates the output shown in (6).

```
(4)    $        $
      / |\    /|
     C V C   C V
     | | |    | |
     f  i  k    r i

(5)   $
     / | \
   * C V C C
     | | | |
     f  i  k r

(6)    $         $
      / \     / | \
     C V     C V C
     | |      | | |
     f  i     k  i  r
```

As the example above was intended to demonstrate, standard linguistic treatments of prosodic phenomena are based on abstract representations and well-formedness conditions. The bulk of current research in this area is devoted to determining what representations are correct, both cross-linguistically and internal to a language. The causal process in the examples above is a rule which is sensitive to syllabic representations, and acts to preserve or create well-formed structures.

In the models described below, the emphasis is somewhat different. Here the causal process involved in the phonological alternations is not an individual rule, but the interaction of the environment and the network through the learning rule. As stated in the Introduction, the learned weights function as constraints on possible outputs: they are the analog of the syllable templates and well-formedness conditions of generative phonology.

One might then raise the objection that since these weights are the functional analog to the CV template, there is no difference between the two approaches. In fact, there is an important difference. In these models there is an intimate connection between the generalizations and the data. The constraints do not function as independent abstractions. While some have argued that this context-dependency is an inherent problem in connectionist language models, we suggest that this is in fact a desirable outcome. It is the fact that these constraints are not independent which allows them to account for data that might not fall neatly into discrete classes.

As an additional point, we stress that these constraints are learned. That is, in the connectionist models the prosodic structure develops from generalization over instances of language.

Phonologists have made similar suggestions (see, for example McCarthy and Prince 1986) without being able to offer an account of how such a process can be instantiated. These models offer insight on this question.

The two simulations which follow demonstrate the validity of this approach. The first model verifies that prosodic structure can seen as a constraint which develops inductively from examples of the language in question. (It is for this reason that the prosody of each language is somewhat distinctive.) Second, such constraints will pressure outputs to conform to the legal structure of the language.

IV. First Simulation

This simulation was run with the aim of looking at the effect of a pre-existing syllable structure. The questions being asked were the following: given examples of well-formed syllables, will the model extract generalizations capable of functioning as templates? If so, how will the language treat forms which deviate from that prototypical structure?

The model developed to address these questions is a pattern associator whose task was an identity map; that is, the input was reproduced on the output layer. The hypothesis was that if a system had learned a set of weights that allowed it to faithfully reproduce good inputs on the output layer, those weights would alter unacceptable inputs, pressuring the output to conform to the general pattern of the language. In addition, we were interested in whether the results of this pressure would correspond to the phonological alternations known to exist in the language.

A. The Model

The model consisted of a three-layer network, with twelve input units, eight hidden units, and twelve output units. This was a fully-connected feedforward network run on the *rlearn* simulator, using the back propagation learning algorithm. The connectivity of the model is illustrated in the diagram below.

```
o o o o o o o o   Output Units
      ^^
      ||
    o o o        Hidden Units
      ^^
      ||
o o o o o o o o   Input Units
```

Input was designed to represent well-formed Turkish words, or rather to correspond to a CV skeleton for those words, with no segmental content.[7] For example, the Turkish words in (a) were represented in the model as the strings in (b).

(a)	(b)
dik	CVC
sirk	CVSC
boksit	CVCCVC

(where C=consonant, S=sonorant[8], V=vowel)

Each slot in the skeleton was represented by two units. One unit encoded information about segment type (C, V, or S) while the other gave information about position in the syllable. The coding scheme is given below.

syllable position	segment type
1 = peak	1 = vowel
0 = onset	0 = consonant
.5= coda or second half of long vowel	.75= sonorant
.25=sonorant in 2nd position in onset	

Thus the Turkish word "brut" would receive the following input representation:

b	0	0
r	.25	.75
u	1	1
t	.5	0

The input "brut" occupies eight of the twelve input units. The input was 12 units to allow for strings of up to six segments. When the string was shorter, as in this example, the remainder of the input space was padded with random entries.

There were 34 input patterns in all. The task of the model was to reproduce the input on the output layer. The system was trained for 1000 epochs, where an *epoch* equals one presentation of each pattern in the training set. At this point the model responded perfectly to the training set. The network was then tested for generalization.

This test data resembled the training data, but differed from it in various respects. The entries in the test file corresponded to novel "words" of the language, with different combinations of syllable types already seen. The entries of the test set could also differ from the training data in length. These changes have a definite effect on the output, even when most of the elements of a test patterns were shared with a training pattern. For example, consider the following pair:

training	*test*
CVCVC	CVCVCV

Here the test input overlaps the training input up to the final vowel. The result, however, is that the final C is a coda in the training data, but must be reproduced as an onset in the test.

B. Results of Test

The generalization test set was divided into three main classes. Ten items corresponded to prototypically acceptable strings in Turkish. A small number of test items were wildly unacceptable as examples of Turkish syllable structure, for example the string CCCV. The last three items were of intermediate acceptability, or have been posited to exist as underlying forms in the language.

In the first class, the model correctly reproduced 8 of the ten forms. The model responded in an interesting way to the other two items in this set. Both these items were patterns with complex onsets, as for example V.CSVC, where the second syllable begins with a consonant-sonorant cluster. The model made no clear decision on where to attach the first C, outputting a response ambiguous between V.CSVC and VC.SVC. This is striking in that although the model saw no examples of the second type, this in fact is also an acceptable syllabification. This sort of graded categorization judgement might be akin to what is referred to in the literature as ambisyllabicity, where a single segment serves both as the coda of one syllable and onset to the next (Kahn 1976).

As for the strongly ungrammatical examples, the model was unable to reproduce the input on the output layer. Instead, it produced near random output, which results in a high error rate. If error is taken as a measure of grammaticality, the response of the model is reasonable. One can contrast this with the cases described above, where prototypical items were reproduced with little error.

Perhaps the most illuminating was the response of the model to the third class of test items. Recall that these were either of intermediate acceptablity, or posited to be underlying forms in Turkish. In all three of these cases, the network edited the illicit forms and produced acceptable syllable structures. For example, given the input *VCCS, the model added a V to the end, and resyllabified to give the good form VC.CSV[8]. In the second case, the input string ?CV:C was modified to CV:.CV. This is interesting because long vowels in closed syllables appear to be only marginally acceptable at normal rates of speech in Turkish (Sezer 1986). This, then, is a reasonable response to this input string. The final form was the most interesting for us, since it corresponded to a bad surface form which is assumed in standard generative theories to be an existing underlying form in Turkish. This was the input string *CVCS. The model responded with the output CV.CVS. Note that the vowel here was not simply added to the end of the string, as seen in the two examples above, but inserted between the second C and the S. This is noteworthy because this is precisely the environment in which Turkish epenthesizes vowels.

C. Discussion

The results of the preliminary simulation were encouraging for a number of reasons. The model was able to reproduce well-formed strings and did not reproduce those that were ill-formed. In addition, it learned to do this while receiving only positive data. Third, it performed as predicted when faced with data that required vowel epenthesis, and inserted a vowel in exactly the position where a vowel is epenthesized in the actual language. Vowel epenthesis is arguably a response to deviations from the prototypical prosodic structure of the language, and the behavior of this network suggests that the conditions controlling vowel epenthesis may be the result of statistical properties of the language in question.

Although this first model showed a number of nice results, the form of the model led to certain problems. Most obviously, the static, fixed-length input-output representations are of questionable validity in looking at language data. This model cannot deal with strings of arbitrary length, since the length of the input must be frozen into the architecture of the model. The "filler" in the input layer is a highly artificial method of accounting for variable length inputs.

In addition, this type of representation seriously restricts the model's ability to generalize what it learns. The problem involves representing sequentially ordered information in a static pattern. Here the input uses a spatial metaphor of time: the temporal order of segments is represented in their order of occurrence. As an example consider the input string *CVSSVS*. In any representation of this string, one wants the second occurrence of *VS* to be seen as in some way the same pattern as the first, occuring later in time. Without this, the network is unable to generalize what it has learned across positions.

This recognition of an invariant pattern across temporal translation is not an easy problem, given the model described above. In the first place, the input to the system is not the string of symbols given in the preceding paragraph. Instead it is a pattern of activation corresponding to binary bit codes at each position. This pattern of activation can be represented as a vector of length twelve. A crucial fact about a vector is that it is an *ordered* list of numbers. [2,7,10] and [10,2,7] are made up of the same numbers, but are entirely different vectors. This has consequences for our problem of temporal translation. Here translation in time is treated as translation in space. Assume as an illustration that *VS* is represented as the pattern *101*, while the rest of the sentence consists of 0's. The problem is that

[0 0 *1 0 1* 0 0 ... 0]

and

[0 0 0 0 0 ... *1 0 1*]

include the same pattern, but the spatial translation has resulted in very different vectors. Since what the system receives as input is vectors like these, it is very difficult to interpret the two patterns as the same.

In fact, there is no evidence that the model does develop any notion of a higher-order structure despite the addition of syllable placement information in the input. Clustering of the hidden unit activation patterns shows no interesting groupings at the level of the syllable. Instead, the network appears to be referring only to linear order in the input string. The hierarchical cluster of the hidden unit activations shows that the patterns are divided into strings that are V-initial and those that are C-initial. These groups are subdivided into strings that contain consonant clusters and those that do not. It is true that the consonant-cluster groupings are further subdivided in a way consistent with syllable structure, but this grouping appears to reflect segment type (e.g. CS versus CC) more than syllable type.

It appears that the identity-map problem and the problem of vowel epenthesis can be solved by referring to string adjacency alone. In order to address further questions on prosodic structure, it is necessary to model other phonological processes, those which seem to rely on such notions as syllable weight and mora preservation. In the current system, however, it is difficult if not impossible to do so. Consider the following data. In this example (from Turkish) the second vowel appears as short in a closed syllable, but is long if the coda consonant is syllabified with a following vowel (data from Clements and Sezer 1982).

nominative	*accusative*	
em.lak	em.la:.ki	"real estate"
u.sul	u.su:.lu	"system"

This is an interesting problem for our account, but cannot be modeled with the current system. Since we currently have no means of designating the notion of a lexical item, this system can recognize inputs of

VS.SVC VS.SV:.CV

But it has no way of relating the two, hence no basis for forming generalizations across the two inputs.

Finally, the input-output system described above is heavily influenced by certain theoretical approaches to these data. Most obviously, information about syllable type is included in the input. This notion is borrowed wholesale from theoretical linguistic analyses, and its inclusion in the input representation betrays two implicit assumptions: first, that the domain in question is the syllable precisely as defined in linguistic theory. Second, the syllabic information is somehow already available in pre-existing underlying representations. Even more importantly, this model unquestioningly accepts the notion of an abstract underlying representation. The training set consists only of good strings, but the test set contains a mixture of both good and bad. When we claim that the model takes an unacceptable input and modifies it into a correct output, it is hard to take that input to be anything but an underlying representation which cannot exist on the surface.

V. Second Simulation

This simulation takes a different approach in an attempt to eliminate the problems outlined above. This section will begin with a description of the model, then show in what ways this architecture responds to the issues raised in the preceding section.

The network used in these simulations is based on work by Michael Jordan (1986b). In these models output is a sequence of actions (here, a sequence of phonemes). There is no intuitive relationship between the input and output; rather, the input (or that part of the input referred to as the *plan*) is an arbitrary vector which functions as the trigger to a particular output sequence. A concrete example will help clarify the input-output relationship. Suppose that the desired output is the Turkish word *oul*, 'son'. In this case the input (that is, the plan) is an arbitrary vector of numbers, for example 11011. This plan is presented and held constant throughout the production of the output sequence "o-u-l". Although there is no phonological information present in the input, this particular plan serves to cue the correct phonological output because the two are associated in this fashion during the learning process. In the discussion of the model, we refer to the arbitrary plan vector as the "name" of the word to be produced.

In addition to the plan, the input layer involves a bank of units which encodes information about the temporal nature of the output. This is referred to as the *state* vector. As will be discussed below, the state vector provides a context in which new inputs are processed.

A. The Model

The model used a 4-layer feedforward network. The pattern of connectivity was as diagramed below.

```
       o o o .......o o  o  o    Output
        | | |  | | | |
        | | |  | | | |
      o  o  o ....o  o o o  o    Hidden 1
     / | | |      | | |
    /  | | |      | | |
   o..o   o..o o       | | |    Hidden 2
    |      ||         | | |
    |      ||         | | |
  o...o  o..o o       o....o o    Input

   (a)    (b)          (c)

      plan            state
```

The input layer consisted of three banks of units (to be described in more detail below.) Each of the first two banks connected directly to its own bank of hidden units, as shown in the diagram. A second hidden unit layer received activation from these intermediate hidden units, and from the third group of input units, which are referred to here as the *state* or *context* units. The output units received activation from both the input and final hidden unit layer. In addition, the target of the output layer was fed back to the state units. This was a fixed connection, meaning that no learning took place on these weights. The state units are self-connected to produce an exponentially weighted average of the output history.[10]

The input layer comprised two major subdivisions, *plan* and *state*. The *state* units give a representation of the output of the system at the previous point in time. The 26 units of the *plan* pool were subdivided into two parts - 22 units over which the name of a word was represented, and four units which represented the morphological form that word was to take on the output layer. There were 22 distinct words in the training data. The first part of the *plan* (labeled (a) in the diagram above) consisted of 22 units to allow each of these names to be represented locally. That is, a separate unit was dedicated to each of the 22. In this simulation each word could appear in one of two morphological cases - nominative or accusative. These morphological variants also received a localist encoding[11] over the units in the bank labeled (b).

B. Advantages of this Architecture

Before discussing the results of the simulation, we will point out the ways in which this architecture avoids some of the problems of the previous model. In the first place, the Jordan model avoids the difficulty of a fixed-length representation of variable-length patterns, since the length of input patterns is no longer frozen into the architecture.

In addition, this type of input representation does involve the notion of a "lexical item". This is exactly what the first part of the plan corresponds to. What this means is that the system can represent relationships among different output patterns. In this model, outputs like *emlak* and *emla:ki* are not unrelated, but are two versions of the same entity. This should allow the network to form generalizations across these two related items.

Finally, this model is able to pose problems involving phonological alternations without having to take a stand on the psychological reality of underlying representations. The input form that is represented in the model is not a phonological form, but a plan that can be equated with either the "meaning" of the word, or simply the intention to produce it. A form which never appears on the surface need not be represented as an input.

It is for these reasons that this architecture is of interest in developing a system which correctly models the language data. The simulation under consideration is an attempt at accomplishing this.

C. Simulation

The purpose of this simulation was to reproduce the results of the first simulation with a Jordan network. The technique used was to teach this network to correctly produce a number of words in a variety of inflected forms. The data was taken from Turkish, which both is very rich morphologically, and shows a number of interesting phonological processes.

In this simulation, there were fourteen stems and two morphological variants of each, for a total of 28 possible forms. The input was given as described in the previous section. The output was processed dynamically. That is, the phonemes of the word were represented sequentially, one per cycle, over the eleven bits of the output layer. These eleven bits encoded syllable location (onset, peak, and coda) and a ten-bit modified distinctive feature matrix, shown below.

1. syllable placement[12]
2. vocalic
3. consonantal
4. front/back
5. voiced
6. nasal
7. high
8. low

9. stop
10. strident
11. round

Inputs were fixed arbitrary patterns but the outputs were of varying duration. A plan was presented and held constant (with changing state unit activations) for as many presentations as there were phonemes in the correct response. For example, in receiving the instruction (ie the plan) to produce the word *emla:ki* (in the accusative case) on the first iteration this input triggers the output /e/. On the second iteration, the plan is repeated and the network is expected to produce /m/. This process iterates through the entire word. At the end of one word, a new plan was presented, and the system was reset by presenting 0's on the state vector.

The network was trained on a subset of 24 of the 28 possible forms. These involved the 14 different lexical items, with one or both morphological variants of each. Since there was one presentation of the plan for each phoneme in the 24 forms, the training set contained 100 pattern lines.

D. Results

As stated above, the task of the current model was to take in an arbitrary code for a word and produce the correct surface form. The simulation ran for 10000 epochs, at which time the total summed squared error (*tss*) was 7.73. This did not represent perfect learning, but as it equals the total error on eleven output units for 100 patterns it is actually quite low.[13] The actual errors in the training set were limited to a small subset of the patterns. Of the 7.74 tss error, 5.05 came from only four of the 100 input patterns.[14]

The network was then tested on a set of four novel patterns. Testing involved giving an input pattern corresponding to a stem plan which the network had seen, combined with a novel morpheme plan. For example, if the training set had included word A only in the nominative, the test set asked for the accusative form. If the network had seen only the accusative form of the word during training, it was tested on the nominative.

The task the network had been given was to produce the phonological form of a word, given an arbitrary plan corresponding to that word. However, the hypotheses being tested were the same as in the original simulation. That is, the model develops a set of weights in the process of learning to produce the correct output patterns. It is these weights that permit the network to correctly output the forms it has learned, and the generalizations encoded in these weights are the equivalent of "well-formedness conditions" which impose themselves on the outputs. As the network develops a set of weights which allow it to produce the correct phonological forms, those weights act as constraints on future outputs. The prediction here, as in the first simulation, is that those constraints will result in phonological alternations that correspond to real processes occuring in the language. The results of the test were very encouraging in that they confirmed this prediction, even though the overall generalization was not perfect.

The test set consisted of the plans corresponding to the four following output forms:

bakɨr	'copper' (nom)
cɨɨrɨ	'era' (acc)
garipi	'strange' (acc)
fikir	'idea' (nom)

The first, *bakɨr*, was reproduced perfectly. This was the nominative, or unaffixed, form. The next two entries in the test set were in the accusative case, and these were output as vowel-final, as required. In one case (cɨɨrɨ) this additional vowel harmonized with those of the stem. Vowel harmony is a phonological process evident in Turkish. Once again the model extracted this generalization even though this was not an original intent of the simulation. The third entry, *garipi*, was produced with the final high vowel which marks the accusative case. However in this case the vowel did not harmonize with the stem vowel, so that the actual output was closer to an /u/ than to an /i/.

fikir, the last entry in the test set, is the most interesting. The network was trained on the accusative form of this word, which is *fikri*. Notice that if the network creates a nominative form simply by eliminating the accusative affix, the expected output is **fikr*. As was shown in Section II, *fikr* is not an acceptable phonological form in Turkish. Although this model was given no information on syllable-structure constraints, it correctly "epenthesized" a high vowel between the stop and the sonorant.

VI. Discussion and Future Research

A. Discussion of Current Model

This model successfully duplicates (in more detail) the results in the first simulation, and does so in a way which appears to confirm the hypothesis being tested. That is, the set of weights which a model develops in the process of producing examples of good forms in a language will constrain the model's outputs to follow certain patterns, and what is referred to as prosodic structure is in fact the generalizations embodied in those weights. The imposition of these patterns on outputs will result in just those alternations attested in the language being modeled. Such an alternation is demonstrated in this model by the pair *fikri* and *fikir*.

The current model not only duplicates the results of the first simulation, but also eliminates many of the problems raised in Section III (C). The use of the Jordan network eliminates the necessity of fixing the length of input patterns in advance. In addition, the use of the plan vector to represent inputs allows the model to associate two different inputs as variations on the same entity. The use of this input representation also eliminates the need to assume an underlying representation for an item.

It is true that the representational system used here continues to borrow from pre-existing linguistic analyses. The real question, however, is not whether the model adopts features of phonological theory, but whether this unduly influences the behavior of the model. Another, related, question is the Pinker and Prince objection raised earlier: whether or not the use of these representational entities is motivated by considerations having more to do with linguistic analyses than with principles of PDP modeling. We consider each of these in turn.

Both parts of the output representation, the distinctive feature matrix and the syllable information, are elements of generative phonological theory. The fact that the segmental output is described as a distinctive feature matrix seems unproblematic for the current approach. We share the theoretical assumption that in some way these features reflect articulatory gestures.

The use of the syllable placement information is perhaps more questionable. The syllable as it is used here is more a theoretical construct than a physical entity; it could be argued that the presence of this information unduly influences the behavior of the model.

However, it may be argued that this coding of syllable information simply duplicates information that is available from facts of sonorancy. In this model we take this representation as an abbreviation for the encoding of complex acoustic events which correlate highly but not perfectly with the syllable placement of the segments. We are currently exploring more acoustically-based feature representations.

B. Future Research

It does appear, however, that this ability to divide consonants into classes was crucial to the working of the model, and that it was the presence of this information in the output representation that caused the model to learn the distinction. A version of this simulation without syllable placement features learned much less readily, and the results of generalization were much less clear. It is of interest, then, to devise a model that learns to make the relevant distinctions without a "teacher" who is already aware of syllable structure.

There are good reasons for this. As discussed in Smolensky (1988), adopting representational features of other theoretical analyses leaves the PDP approach dependent on those other research paradigms. One is then faced with the need to insure that the principles adapting these features to PDP models reflect principles of connectionist computation. Furthermore, adopting the generative phonology classification of consonants as syllable onsets and codas also means adopting the view that these are in fact discrete classes, and that it is possible to divide consonants neatly between them. A look at the data shows that this is overly simple. In many cases syllables have well-defined boundaries, but there exist a number of phenomena which can be accounted for only under the supposition that certain segments are ambisyllabic, behaving simultaneously as both the coda of one syllable and the onset of the next. Likewise, generative treatments of prosodic data often find it necessary to treat word-final consonants as "invisible" to syllabification, or "extrametrical". This is simply another way of saying that an element may meet the structural description of a coda consonant without actually behaving as a member of that class.

A goal of future research is to develop a model which makes these categorizing judgements based on the data itself, rather than on a theoretically biased teaching input. One simulation which is currently being run involves modeling data in which consonants in similar positions in a word exhibit similar behavior. It seems likely that it is behavioral differences of this sort that contribute to the speaker's ability to categorize consonants and make decisions about syllable structure. The assumption underlying this model is that some categorization is necessary to successfully complete the task. However, if the network is left free to develop the necessary distinctions, those distinctions will not strictly follow the division presumed in generative accounts.

VII. Conclusion

As stated in the Introduction, generative approaches to the analysis of prosodic phenomena involve representations and constraints on their well-formedness. Phonological processes are viewed as rules sensitive to representations such as the syllable or the mora, and which act to create or maintain well-formed structures. Although this type of analysis gives a nice account of a variety of phonological phenomena, it raises questions that it makes no attempt to answer.

In the first place, although these constraints are crucial elements in the account, they are taken as given. No interest is paid to what else in the language necessitates that these conditions be as they are. Further, prosodic constraints show a great deal of cross-linguistic variation. This is generally treated as a variation in the representations and/or constraints pertinent to a given language. However, in general no explanation is given for this variation. Third, the classification system on which these representations are based is presumed to be composed of stable, discrete classes of elements.

The account offered in this paper questions these assumptions. First, the claim is that prosodic structure is not simply given, but is learned from statistical properties of the target language. In learning to produce valid samples of a language, the models must develop a set of weights which allow them to associate input patterns with correct outputs. It is these weights which function as "well-formedness conditions" governing the prosodic structure of a language. In associating input with output, the network also develops an activation pattern on the hidden layer, in which the inputs are restructured in a way that allows the network to correctly associate them with output patterns. This internal representation, which again is learned, also reflects the generalizations which the system extracts from the input.

This learning of constraints and representations is important for a second reason. The weights develop to allow particular strings to be produced on the output layer. If the output (i.e., the phonlogical properties of the language being modeled) were different, a different set of weights would develop. This difference in the weights corresponds to a difference in the constraints developed by the network. Thus the model suggests an account of cross-linguistic variation.

Finally, the models offer a chance to study the representations that develop in the process of producing correct phonological forms. Further analysis of the hidden unit representations devised by the models can shed light on the problems in classification alluded to above. The activation patterns developed on the internal layer are highly context-dependent representations of inputs. These patterns exhibit a similarity of structure which contains information about how inputs are categorized, but this structure is graded, and the resulting categories are not discrete. The model discussed in Section VI exhibits just this type of graded categorization in its response to the ambiguous input V.CSVC. Importantly, the model can then use these context-dependent representations in the production of output, without the need for any further theoretical apparatus.

[The font size used in the production of this paper was requested by the authors.]

NOTES

We thank Jeff Elman for guidance and Steve Poteet for profitable discussions on this project. Comments by Sally Rice and Farrell Ackerman on an earlier draft were especially helpful.

1. This is an oversimplification. Although the syllabic nucleus is prototypically a vowel, in many languages sonorant consonants also play this role.

2. Syllable division is indicated by the period. The symbol V: represents a long vowel.

   Note that in the analysis quoted here, *sent* is not a surface form, but is posited as the abstract underlying form of the root morpheme. It would be judged ill-formed on the surface.

3. The domain of mora preservation is often larger than the syllable, for example the foot or the word.

4. The suffix *an/en* carries a number of meanings.

5. Elements dominated by $ form one syllable.

6. The *k* joins the second syllable for reasons which are not relevant to the present analysis.

7. Where "no segmental content" means no information at the level of the individual phoneme. The input specifies "consonant", for example, without describing any particular consonant.

8. For the purposes of this simulation, "sonorant" specified *r*, *l*, or a nasal consonant.

9. In the input each of the [+cons] segments was represented as a syllable onset. Syllabification here was .C.C.S.

   Given the way input is represented in this model, it is impossible to input a string which has *not* been syllabified in some way. This is a potential problem with the network, since this predetermined syllabification can have an effect on the behavior of the system. This problem of underlying representations is discussed in the next section, and is avoided in the second simulation.

10. *mu*, the multiplier on this connection, was .5 .

11. This part of the *plan* consisted of 4 units to allow for expansion in the number of morphological variations learned by the model.

12. Information about segment type was distributed over this and the next unit, with C = 0 1, V = 1 0, S = 1 1.

13. Chance error on the 11 output units would equal 5.5; squared this is approximately 30 tss for one output pattern. There were 100 output patterns.

14. The largest error was centered on the following four segments. (The upper-case letter indicates the offending segment, while the lower-case letters show the context.)

target	*output error*
prenS -	onset (should be a coda) +stop, -strident
taSdiki -	onset

+ stop

endRaaki - syllable peak
- consonantal

endraAki - -low (i.e., did not agree in height with preceding vowel)

In three of the four errors, the segment forms part of a CC or CS cluster. *endra:ki*, in particular, is problematic since this is the only form involving a string of three [+cons] segments.

BIBLIOGRAPHY

Chomsky, Noam (1980). "Rules and Representations" *Behavioral and Brain Sciences*, 3, pp. 1-61.

Clements, G.N. and S.J. Keyser (1983). *CV Phonology: a Generative Theory of the Syllable*; Linguistic Inquiry Monograph # 9, MIT Press, Cambridge, Mass.

Clements, G.N. and E. Sezer (1982). "Vowel and Consonant Disharmony in Turkish" in H. van der Hulst and N. Smith (eds): *The Structure of Phonological Representations*, Part II; Foris Publications, Dordrecht, Holland and Cinnaminson, New Jersey.

Hayes, Bruce (1987). "Compensatory Lengthening in Moraic Phonology". UCSD Colloquium.

Jordan, Michael (1986a). "Introduction to Linear Algebra in Parallel Distributed Processing." in Rumelhart, D.E., J.L. McClelland, and the PDP Research Group.

Jordan, Michael (1986b). *Serial Order: A Parallel Distributed Processing Approach*. ICS Report # 8604. UC San Diego.

Kahn, D. (1976). *Syllable-based generalizations in English Phonology.* Bloomington: Indiana University Linguistics Club.

Kornfilt, J. (1986). "Stem Penultimate Empty C's, Compensatory Lengthening and Vowel Epenthesis in Turkish" in L. Wetzels and E. Sezer (eds): *Studies in Compensatory Lengthening*; Foris Publications, Dordrecht, Holland, and Riverton, New Jersey.

Kuroda, S-Y. (1967). *Yawelmani Phonology*. Cambridge, Mass. MIT Press.

McCarthy, A. and S. Prince (1986). "Prosodic Morphology" ms.

McClelland, J.L., Rumelhart, D.E., and the PDP Research Group (1986). *Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Volume 2: Psychological and Biological Models.* Cambridge, MA. MIT Press/Bradford Books.

Piatelli-Palmarini, M. (ed) (1980). *Language & Learning*. Harvard University Press, Cambridge, Mass.

Pinker, S. and A. Prince (1987). "On Language and Connectionism: An Analysis of a Parallel Distributed Processing Model of Language Acquisition." *Cognition*, 28, pp. 73-193.

Rumelhart, D.E., G. Hinton, and R.J. Williams (1986). "Learning Internal Representations by Error Propagation." in Rumelhart, D.E., J.L. McClelland, and the PDP Research Group.

Rumelhart, D.E., J.L. McClelland, and the PDP Research Group. (1986a). *Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Volume 1: Foundations.* Cambridge MA. MIT Press/Bradford Books.

Schane, Sanford. (1968). *French Phonology and Morphology*. Cambridge, Mass. MIT Press.

Sezer, E.F. (1986). "An Autosegmental Analysis of Compensatory Lengthening in Turkish" in L. Wetzels and E. Sezer (eds): *Studies in Compensatory Lengthening*; Foris Publications, Dordrecht, Holland and Riverton, New Jersey.

Smolensky, Paul (1988a). "The Proper Treatment of Connectionism" in *Behavioral and Brain Sciences*, 11, pp 1-74.