Learning Morphophonemic Processes without Underlying Representations and Explicit Rules

Chan-Do Lee and Michael Gasser

Traditional phonology presupposes abstract underlying representations (*UR*) and a set of *rules* to explain the phonological phenomena. There are, however, a number of questions that have been raised regarding this approach : Where do URs come from? How are rules found and related to each other? In the current study, a connectionist network was trained without the benefit of any UR and *explicit* rules. We hypothesized that rules would emerge as the generalizations the network abstracts in the process of learning to associate forms (sequences of phonological segments comprising words) with meanings (of the words) and URs as a pattern on the hidden layer. Employing a simple recurrent network we ran a series of simulations on different types of morphophonemic processes. The results of the simulations show that this network is capable of learning morphophonemic processes without any URs and explicit rules.

1. Introduction

There are various theories on how and why some observed phonological phenomena occur in the way they do. However, most traditional phonology theories presuppose abstract underlying representations (*UR*) and a set of *rules* to explain the surface realization. Modern generative phonology is based on the notion of "deriving" forms through the application of rules, each of which takes a linguistic representation as input and yields one which is in some sense closer to the "surface." The idea is that behind surface forms are URs, abstractions within which each morpheme has an invariant form.

There are, however, a number of questions that have been raised regarding

this approach. How does knowledge about URs and rules relate to the psycholinguistic processes of production and perception, which relate form and meaning? The linguistic knowledge in URs and rules is meant to belong to "competence" and should thus be shared by both production and perception. Production might be to some extent parallel derivation, but perception would be the reverse process. Thus we have the familiar problem of using rules in one direction when they were designed for another.

A more serious problem, however, comes in imagining how knowledge about URs and rules might ever get learned. That is, given only surface input forms together with meanings inferrable from context, how is a learner to figure out how the form-meaning relation gets mediated by abstract URs? Where do URs come from? How are rules found and related to each other?

It is customary to assume that a language learner is helped by having certain predispositions wired in; however, we begin with an approach which is far more constrained. We assume that the basic building blocks of language acquisition and processing are the simple, neuron-like processing units that connectionist models start out with. What gives such a system its intelligence is its architecture.

Connectionism (Rumelhart and McClelland (1986), McClelland and Rumelhart (1986)) is an approach to cognitive modeling which assumes that knowledge is represented by weighted connections, spreading activations over large numbers of densely interconnected units. A network consists of input units, which respond to stimuli from the outside world, and output units, which represent the system's response to that input. There may be one or more "hidden" units. Each unit has an activation value, which is updated by multiplying each incoming signal by the connection weight along which it is received, summing these inputs, and passing them through some function, thus obtaining a new value. Processing involves activating input units; this activation spreads through the connections to produce a pattern of activations on the output level. This pattern is compared to "desired" output and the discrepancy is back propagated to adjust the weights.[1]

In the current study, a connectionist network is trained without any UR

[1] This is only one example of learning algorithms. To go further and introduce more of them is beyond the scope of the current paper.

or *explicit* rules. We expect that the rules come out in the form of the generalizations the network abstracts in the process of learning to associate form (sequences of phonological segments comprising words) with meanings (of the words). In the network the rules are determined by the connection weights between units which the network develops while trying to produce correct outputs. The weights are thus to be learned, not to be presupposed. As the network develops generalization over many exemplars, the weights act as constraints on future outputs. The prediction here is that these constraints will account for different allomorphs when novel inputs are given.

What we're trying to show in this paper is that our approach is "performance" phonology with different goals from generative phonology, which is "competence" phonology. We think our approach is more psychologically plausible and it might be that ours is the only way that's really needed : generative phonology might become superfluous once acquisition, production, and perception are understood. We don't think that each underlying segment goes through a derivation employing phonological rules to produce a "surface" segment. What counts is that for the speaker, meanings trigger the phonological production, whereas for listeners pholological material directly evokes word meanings. Here the rules are built into the associations between forms and meanings and URs are encoded as distributed representations somewhere on the associations.

Through a series of experiments on morphophonemic processes,[2] we will show that (1) rules are determined by the connection weights between units which the network develops while trying to produce correct outputs, (2) URs are learned as the pattern on the hidden layer that mediates the relationship between form and meaning, and also (3) the network fails to learn the types of rules which are apparently difficult for human language learners.

2. System Structure

We use a relatively constrained three-layer network, one in which

[2] We will restrict ourselves only to morphophonemic alternations and hope to convince the reader of the plausibility of our approach by just showing only the tip of the iceberg.

feedforward connections are supplemented by limited feedback connections. Figure 1 shows the network architecture we have started with.

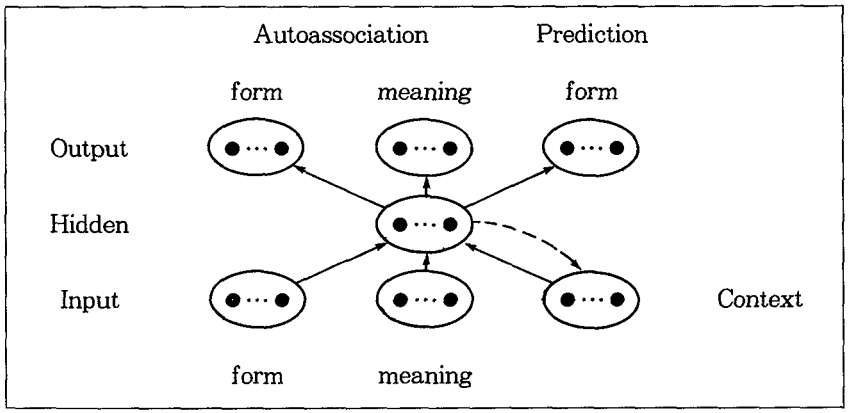

Figure 1: Architecture of the Network Used in the Morphophonemic Rules Study

The architecture shown in Figure 1 is a slight modification of the **simple recurrent network** (SRN) developed by Elman (1989, 1990). Since morphophonemic processes are temporal, we need to have some kind of short-term memory (STM) to store the previous events. The system cannot know how to behave on the basis of only the current input; the previous context is essential. The feedback connections from the hidden layer to the input layer serve this purpose. STM is held by Context units, which are the copies of the hidden layer from the previous time step. Thus at any given time step, the network has not only the current input but also the previous inputs. But because the previous input was also a function of the previous STM pattern, the network has information on many previous time steps.

The solid arrows denote the learnable one-to-many connections between units on the lower level and those on the higher level; for example, one unit on the input layer connects to all units on the hidden layer. The dashed arrow denotes fixed one-to-one connections, meaning no learning takes place on these weights and there is only one connection between any two units. There are no intra-level connections.

The standard back-propagation learning rule (Rumelhart, Hinton and Williams (1986)) is used to train the network.

Our model differs from other SRNs in that it is trained on auto-association as well as prediction. This is a way to force the network to distinguish the different input patterns on the hidden layer (Servan-Schreiber, Cleeremans and McClelland (1989)).

This network has the capacity to associate form with meaning as well as form with form and meaning with meaning. Thus it can perform the task of the production of a sequence of segments given a meaning, or of a meaning given a sequence of segments. It has the potential to make a generalization across phonologically related words.

3. Experiments

In a series of experiments, we have trained networks with the architecture described in the previous section on various morphophonemic processes. We roughly classified morphophonemic processes into 3 different categories: (1) insertion, (2) deletion, and (3) mutation and conducted experiments on different types of rules; the ones which are rarely found in human languages as well as the ones which are easily found.

Our results indicate that a network like this is capable of learning morphophonemic rules by encoding URs on the hidden layer and using them in perception and production processes. That is, given training on the singular, but not the plural of *lip*, we were later able to ask the network to generate the appropriate plural suffix following the stem or to tell us the number of /lɪps/, a form it had never seen.

For example, we trained the network on pairs like the following :

(1) CHIP + SINGULAR → /tʃɪp/
(2) CHIP + PLURAL → /tʃɪps/
(3) LIP + SINGULAR → /lɪp/

and then tested it on pairs like the following to see if it yielded correct morphophonemic realizations :

(4) LIP + PLURAL → /lɪp/ + ??

where the items in capitals represent meanings.

However, this solves the problem in only the production direction. Our

model should also predict meanings given forms. We trained the model on (5), (6) and (7) and tested on (8) to see if it was able to get the correct grammatical number.

(5) /tʃɪp/ → CHIP + SINGULAR
(6) /tʃɪps/ → CHIP + PLURAL
(7) /lɪp/ → LIP + SINGULAR
(8) /lɪps/ → LIP + ??

3.1. Method

Input words were composed of sequences of segments. Each segment consisted of a binary vector which represents modified Chomsky-Halle phonetic features (Chomsky and Halle (1968)). There were 20 words for each simulation. Ten sets of randomly generated artificial words were used for each experiment. Twelve of these were designated "training" words, 8 "test" words. For each of these basic words, there was an associated inflected form. For convenience, we will refer to the uninflected form as the "singular" and the inflected form as the "plural" of the word in question. The network was trained on both the singular and plural forms of the training words and only on the singular forms of the test words. Words were presented one segment at a time. Each word ended in a word boundary pattern consisting of all zeroes.

Each "meaning" consisted of an arbitrary pattern across a set of 6 "stem" units, representing the meaning of the "stem" (hereafter referred to as *s-meaning*) of one of the 20 input words, plus a single bit representing the grammatical number (hereafter *g-number*) of the input word (0 for singular, 1 for plural).

The network was trained on the auto-association and prediction task. On 4 out of every 5 words, the network was given complete words and meanings. On 1 out of every 5 words, the input g-number was treated as unknown. That is, the g-number unit was set to an intermediate value of 0.5 word-initially and in the subsequent segments to the value that it took on the previous time step. This was necessary to help the network learn the perception task.

The network was trained until the model responded perfectly to the training data. The network was then tested for generalization.

To test the network's performance on the production task, we gave the network the appropriate segments for the stem successively, along with the meaning of that stem and the g-number unit on for plural. We then examined the prediction output units at the point where the plural morpheme should appear. Based on Euclidian distance, we converted each output pattern to the nearest phoneme.

To test the perception performance, we gave the network the sequence of input segments of a word, set the stem meaning units to the appropriate pattern, and set the g-number unit initially to 0.5. At the presentation of each new segment, the g-number unit was copied from the output on the previous time step. We then examined the output g-number unit after the appearance of either the appropriate plural form or word boundary.

3.2. Experiment 1: Insertion

3.2.1. Task

Two separate experiments were conducted to test the network's ability to acquire morphophonemic processes which add a phoneme to the stem. CVC words from an artificial language were used to test the "suffix" and "prefix" rules which added an /s/ or /z/ to singular words to form plural words. The affixes agreed on the voice feature with the following or previous segment. The experiments are described in more detail in Gasser and Lee (1991). There were 10 separate simulations for each of the two artificial inflectional rules.

3.2.2. Results

The network predicted the correct segments for all of the training words. When test words were presented, most of test words were correct.

For the training words, the output g-number unit fluctuated around 0.5 until the relevant information was given. Then it correctly turned on or off according to whether the word boundary or plural ending appeared. For the test words, the network consistently output 0 before the appearance of the relevant information. This is not surprising since the network only saw singular forms during training. When the word boundary appeared in the

input, it correctly turned off. When the plural morpheme appeared, the output of the g-number unit was correct for most test items. Results for the test words are summarized in Table 1.

Table 1: Results of Insertion Experiments

	Production	Perception	n
	% Affixes Correct	% Plurality Correct	
Suffix	82.5	79.0	80
Prefix	76.3	76.0	80

3.3. Experiment 2: Deletion

3.3.1. Task

In this set of experiments, one of three rules was used to generate the plural forms in which a segment was deleted from the singular words: from the beginning of a word (CVC→VC), from the middle (VCCV→VCV), and from the end (CVC→CV). To the authors' knowledge, there exists no human language which undergoes the first type of rule for inflection. The second type of rule can be found in some American Indian languages, especially in Muskogean languages (Hardy and Montler (1988a, 1988b)). The third rule is analogous to the French rule for masculine adjectives. There were 10 separate simulations for each of the three inflectional rules.

3.3.2. Results

The network learned the set of training words for all three rules quite successfully. Segments were produced correctly more than 99% of the time and the network predicted plurality more than 99% of the time, too.

The results for the test words are shown in Table 2. In the table, "% Segments Correct" refers to the percentage of the segments which the network predicted correctly after a segment was deleted. The network predicted the word boundary more than half of the time when it was tested on the test words after being trained to delete the final consonant ("post-del"). It was quite bad at deleting a segment in the middle ("mid-del"), and even worse

for the case it had to delete a segment from the word beginning ("pre-del"). Note that the network failed to learn the types of rules which are rarely found in human languages. For the perception task, the network performed little better than chance. The figures shown in the table are average numbers over the 10 separate runs. In some of the runs the network did very poorly, yet in other runs it performed very well.[3] In fact, in one of "post-del" runs, the network produced correct segments for 7 out of 8 test words, and predicted the plurality for all 8 words. The best run for the "mid-del" case produced 4 segments correct, while predicting all 8 test words as plural. The "pre-del" runs performed consistently bad over 10 separate runs. Note that some aspects of the rules had been learned. Thus in about 80% of the cases the network produced the correct syllable structure (77% VC for "pre-del", 80% CVC for "mid-del"; in "post-del" cases it always predicted CV.).

Table 2: Results of Deletion Experiments

	Production	Perception	n
	% Segments Correct	% Plurality Correct	
pre-del	12.5	60.0	80
mid-del	23.8	73.8	80
post-del	57.5	67.5	80

3.4. Experiment 3: Mutation

3.4.1. Task

Two different kinds of experiments were done to test the mutation rules. In one experiment, the network was trained to change a feature of the singular word in all segments to generate plural word. This is analogous to a tone rule where singular words are in low tones, while plural tones are in high tones. The network was to test if it can make use of the analogue of a tone tier (cf. Goldsmith (1990)). In another experiment, the "reversal" rule was tested: the plural words were generated by reversing the segments of

[3] It is not unusual to have variation in the results on different runs; it has been shown that the back propagation learning algorithm is sensitive to initial weights (Kolen and Pollack (1990)). This is one of the reasons why I ran 10 separate simulations.

the singular words. As with the previous experiments 10 separate simulations were run (see Gasser and Lee (1991) for more detail on "reversal" experiments.)

3.4.2. Results

The network was able to learn very well the "H-L Tones" task. Also it was very good at generating "high" tones for novel plural words and perceiving a novel word with a "high" tone as a plural word.

Yet the network failed to generate the reversed form, even though it learned the training words more than 99% of the time. For the perception task, the network is performing at a level considerably worse than chance(50 %). This is apparently due to the fact that during training the network was exposed to singular and plural forms of training words but only singular forms of test words. Thus it saw more singular forms overall and, given no evidence one way or the other, responds with an activation less than 0.5 on the g-number unit. The network finds it much harder to learn a reversal rule of a type which is apparently difficult for human language learners. Results are summarized in Table 3. In the table, "% Segments Correct" refers to the percentage of all the segments which the network predicted correct.

Table 3: Results of Mutation Experiments

	Production	Perception	n
	% Segments Correct	% Plurality Correct	
H-L Tones	97.5	99.1	80
Reversal	22.5	13.0	80

3.5. Analysis of Hidden Layers

So far we have reported that the network was able to learn apparent rule-governed morphophonemic processes in a manner that makes use of associations between forms and meanings. The next question we should be able to answer is: where are the URs? It is the pattern on the hidden layer that mediates the relation beween form and meaning. Analysis of hidden layers of the network indicates that certain units there are dedicated to rep-

resenting the plural morpheme, independent of its surface form. Thus it appears that our networks have the capacity to learn distributed URs.

We analyzed hidden unit representations which the network developed during a successful run of "post-del" task.

Figure 2 shows a box plot of hidden unit activations of 10 randomly-picked words (test words as well as training words) at the segment before it recognizes the plurality (perception task): after the second consonant was input for singular or after the word boundary for plural words. Only selected 5 units are shown in the figure. Big boxes are activation value close to 1.0 and dots denote value close to 0.0.

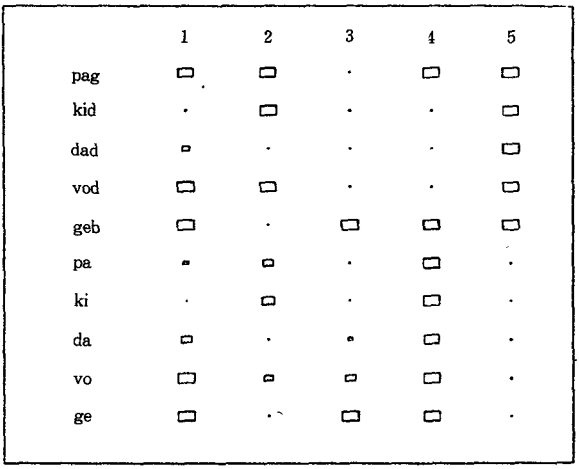

Figure 2: Box Plot of Hidden Layer Activations before Recognizing the Plurality during the Perception Task in a "Post-del" Task Run.

From the figure we can easily see that unit 5 distinguishes plurality. Further analysis shows that the unit isn't turned on until it has enough information to decide the number of the word in question.

Now what happens when the network is given the production task? In this case we found the same unit is responsible for producing plural words; the unit is on for the singular words and off for the plural words.

Figure 3 shows another plot of hidden unit activations for production task, that is, after the second segment (vowels) was presented. One might argue that since number is presented in the input layer the unit can easily represent it by simply copying the value. But this is not the case. The unit turns on only after the second segment for singular words.

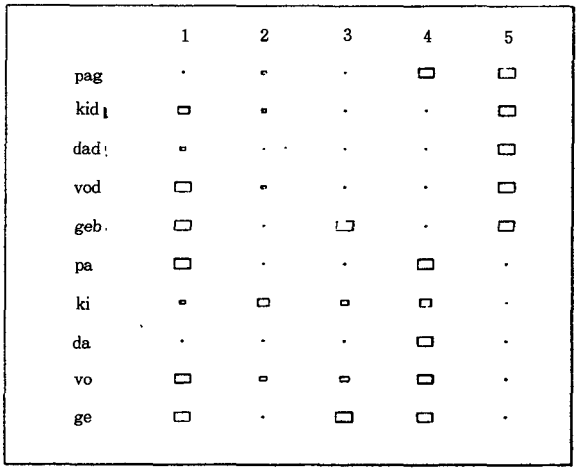

Figure 3: Box Plot of Hidden Layer Activations after the Second Segment was Input during the Production Task in a "Post-del" Task Run.

In this subsection, we have shown only two analyses for a "post-del" experiment. More extensive analyses for other experiments are reported in Lee and Gasser (1992).

4. Discussion and Extensions

This model successfully learns morphophonemic "rules"(of the kinds which can be found in human languages) by abstracting the generalizations from the exemplars on the connection weights in the process of learning. The set of weights the model developed in the process of producing desired plural morphemes constraints the model's outputs to follow the desired patterns, and what looks like a "rule" is in fact the generalization embodied in these weights. The results indicate that it has the ability to learn apparent rule-governed morphophonemic processes in a manner that makes use of associations between forms and meanings. Note that it failed on the types of rules which are rarely found in human languages.

The analogues of underlying representations are encoded as a distributed representation on the hidden layer which the network develops. An important question for future investigation concerns what happens in cases where the traditional analysis posits a sequence of rules operating on inter-

mediate representations at different levels of abstraction.

Is our network as it is powerful enough to produce correct outputs when there are some rule interactions? Do we need to make some modifications to our model to incorporate rule interactions? If so, how much? Is changing some of the parameters affecting the behavior of networks enough to accommodate the seemingly more difficult problems? Do we have to make some drastic changes? These questions are yet to be answered.

Another question we hope to answer is: how easy will it be to retrain the network to predict another form which is phonologically very similar after it is trained on one type of rule? For example, third person singular present verb suffix also has forms /s/, /z/, /ɪz/ which can be found in English plural suffixes. There is a distinction between a morphological category, such as 'plural', and realization of it in phonological substance. For example, the relation can be many-one: the same phonological entity can mark several categories. Human beings can easily transfer knowledge from one environment to another. For the model to account for psychological data, it should handle this case rather easily. That is, it should be easier for this model to learn a second or third morphophonemic task after the first.

5. Conclusion

In this paper, we have presented the problem of learning phonological processes and showed how a recurrent connectionist network can learn many different types of morphophonemic processes without explicit rules and URs. The model exhibits the capability of summing up the generalizations abstracted from the exemplars, thus eliminating the need of presupposing the abstract underlying representations and rules, which constitute the major part of the generative phonology.

The study of phonology itself has not attracted much interest in the cognitive science community. When it is studied in conjunction with another subfield of linguistics, however, it is useful to do computational phonology. When the subfields of linguistics, such as phonetics, phonology, morphology, semantics, and syntax, are studied together as a whole, not unrelated separate parts, and when we take representation scheme together with models of processing, we may get more insights into how phonological phenomena

are related to cognitive processing and hypothesize the forms of mental representations.

We hope our study of morphophonemics has shed some light on how phonological phenomena are related to cognitive processing and on what mental representations are like. The current study has the potential to provide some clues as to how human cognition works.

References

Chomsky, N. and M. Halle (1968) *The Sound Pattern of English,* New York: Harper and Row.

Elman, J. (1989) *Representation and Structure in Connectionist Models* (Technical Report 8903), Center for Research in Language, University of California, San Diego.

Elman, J. (1990) 'Finding Structure in Time,' *Cognitive Science* 14, 179-211.

Gasser, M. and C.-D. Lee (1991) 'A Short-term Memory Architecture for the Learning of Morphophonemic Rules,' In R. Lippmann, J. Moody, and D. Touretzky (Eds.), *Advances in Neural Information Processing Systems* 3 (pp. 605-611), San Mateo: Morgan Kaufmann.

Goldsmith, J. (1990) *Autosegmental and Metrical Phonology,* Cambridge: Basil Blackwell.

Hardy, H. and T. Montler (1988a) 'Alabama Radical Morphology: h-infix and Disfixation,' In W. Shipley (Ed.), *In Honor of Mary Haas* (pp. 377-409), Berlin: Mouton de Gruyter.

Hardy, H. and T. Montler (1988b) 'Imperfective Gemination in Alabama,' *International Journal of American Lingusitics* 54:4, 399-415.

Kolen, J. and J. Pollack (1990) 'Back Propagation is Sensitive to Initial Conditions,' *Complex Systems* 4, 269-280.

Lee, C.-D. and M. Gasser (1992) 'Where Do Underlying Representations Come from?: A Connectionist Approach to the Acquisition of Phonological Rules,' To appear in J. Dinsmore (Ed.), *Closing the Gap: Symbolicism vs. Connectionism,* Hillsdale: Lawrence Erlbaum Associates.

McClelland, J. and D. Rumelhart (1986) *Parallel Distributed Processing,* Vol. 2, Cambridge: MIT Press.

Rumelhart, D., G. Hinton and R. Williams (1986) 'Learning Internal Repre-

sentations by Error Propagation,' In D. Rumelhart and J. McClelland (Eds.), *Parallel Distributed Processing*, Volume 1 (pp. 319-362), Cambridge: MIT Press.

Rumelhart, D. and J. McClelland (1986) *Parallel Distributed Processing*, Vol. 1, Cambridge: MIT Press.

Servan-Schreiber, D., A. Cleeremans and J. McClelland (1989) 'Learning Sequential Structure in Simple Recurrent Retworks,' In D. Touretzky (Ed.), *Advances in Neural Information Processing Systems* 1 (pp. 643-652), San Mateo: Morgan Kaufmann.

Prof. Chan-Do Lee
Department of Information Engineering
Taejon University
Taejon 300-716
Korea

Prof. Michael Gasser
Department of Computer Science
Indiana University
Bloomington, Indiana 47405
U. S. A.