TECHNICAL REPORT NO. 307

Networks and Morphophonemic Rules Revisited

by

Michael Gasser and Chan-Do Lee

March 1990

COMPUTER SCIENCE DEPARTMENT
INDIANA UNIVERSITY

Bloomington, Indiana 47405-4101

Networks and Morphophonemic Rules Revisited

Michael Gasser
Chan-Do Lee
Computer Science Department
Indiana University
Bloomington, IN 47405

Abstract

In the debate over the power of connectionist models to handle linguistic phenomena, considerable attention has been focused on the learning of simple morphophonemic rules. Rumelhart and McClelland's celebrated model of the acquisition of the English past tense (1986), which used a simple pattern associator to learn mappings from stems to past tense forms, was advanced as evidence that networks could learn to emulate rule-like linguistic behavior. Pinker and Prince's equally celebrated critique of the past-tense model (1988) argued forcefully that the model was inadequate on several grounds. For our purposes, these are (1) the fact that the model is not constrained in ways that humans language learners clearly are and (2) the fact that, since the model cannot represent the notion "word", it cannot distinguish homophonous verbs. A further deficiency of the model, one not brought out by Pinker and Prince, is that it is not a processing account: the task that the network learns is that of associating forms with forms rather than that of producing forms given meanings or meanings given forms. This paper describes a connectionist model which addresses all three objections to the earlier work on morphophonemic rule acquisition. The model learns to generate forms in one or another "tense", given arbitrary patterns representing "meanings", and to output the appropriate tense given forms. The inclusion of meanings in the network means that homophonous forms are distinguished. In addition, this network experiences difficulty learning reversal processes which do not occur in human language.

1. Background

Several features of connectionism make it an attractive framework within which to model natural language processing (NLP) and other high-level cognitive processes. (1) Since a network works with patterns of features rather than unitary symbols, concepts may adjust themselves to particular contexts and even blend together. (2) Because connectionist networks are designed to find the best matching characterization in memory for a given set of input features, they are admirably suited to the problem of coping with errors or missing information in the input to comprehension and production. (3) Unlike the absolute constraints of symbolic models, knowledge in connectionist networks takes the form of tendencies which can be overridden when there is strong enough evidence to the contrary, so it is a straightforward matter for such a system to learn exceptions to rules. (4) Processing in connectionist networks involves the attempt to satisfy in parallel as many of the weak constraints embodied in the network as possible; this is precisely the way human language comprehension seems to work. (5) In connectionist networks long-term learning is just the adjustment of weights on connections, a simple idea which contrasts with the plethora of elaborate approaches to learning offered by symbolic models.

Thus there are good reasons for pursuing the goal of connectionist NLP. But there are also good reasons that this goal is far from achieved. Those aspects of NLP which symbolic approaches excel at, and which allow symbolic systems to function relatively efficiently within restricted domains, have to a large extent eluded connectionist approaches. It is a straightforward matter in a symbolic system to specify that a bound morpheme takes on particular forms depending on the shape of the stem it is bound to and to specify in what way the addition of that morpheme to a stem yields a meaning from the meanings of the stem and the bound morpheme. In a distributed connectionist system, however, where there may be no explicit stem, no explicit word, and no explicit rule, these matters are not straightforward at all. This paper considers these issues in the context of a model designed to learn the requisite knowledge in a form which is usable in both comprehension and production.

Connectionist models make use mainly of distributed representations. That is, concepts are represented across more than one memory location, and memory locations take part in the

representation of more than one concept. It is this style of representation which gives connectionist models some of their characteristic strong points, in particular, robustness, representationl fluidity, and efficient use of memory.

The fact that concepts do not appear in the system in the form of discrete objects means, however, that new forms of compositionality must also be developed if the system is to manipulate structured concepts. Fodor and Pylsyshyn (1988) base their critique of the connectionist enterprise largely on the perceived inability of networks to handle structured representations and structure-sensitive operations. In an extensive response to Fodor and Pylyshyn, van Gelder (forthcoming) has argued that connectionist models, in order to prove themselves, need not demonstrate constituency in the usual symbolic sense; they may offer alternatives which are just as good at structure-sensitive operations. Thus it remains to be seen whether networks are limited in this way.

Consider the standard account of what is required in the lexicon of an NLP system in a human or a machine. Among other things, the portion of the lexicon devoted to verbs for a language like English needs to represent the form of the stem for each verb and the general rule specifying the form of the past tense of verbs. This rule might also include information specifying the phonologically conditioned variants of the past-tense morpheme, or this information might appear in a separate phonological component, where it would also apply to bound morphemes other than the past-tense morpheme. For verbs which are exceptions to the rule, the form of the past tense for those verbs also must be specified in the lexicon.

What changes when we try to represent and make use of knowledge of this type in a connectionist system? First, though words must in some sense be composed of the segments that make them up, the use of distributed representations precludes the usual symbolic constituency. Given a rule which simply prefixes or suffixes one allophone of the bound morpheme for the past tense, the stem is easy to find in the symbolic version of the past tense of the verb, but where do we find it in the distributed representation? The answer depends, of course, on what sort of distributed representation is being used, but in any case it is not a simple answer. Another way to view the problem is to ask what the past and present tense forms of the verb share. In the standard symbolic representation, the past form is the present form plus something. In the distributed case, the past and present forms are patterns over the same set of units.

How then can a system which makes use of distributed representations have a lexicon at all? That is, what form would the knowledge associating stems with meanings take in the system when there don't seem to be any stems at all, at least not in the usual sense?

One answer to a part of this question comes from the well-known past-tense model of Rumelhart and McClelland (1986). Their model takes distributed representations of English present-tense verbs as inputs and learns to associate them with distributed representations of the corresponding past tense forms. The main achievement of their model is the demonstration that such a network could learn to correctly generate the past-tense forms of both regular and irregular verbs and to generalize to some extent to novel forms. Much of the discussion of this model, and subsequent work by Plunkett and Marchman (1989), focused on the stages of learning, the effects of varying particular properties of the input, and the particular form of distributed phonological representation used (Pinker & Prince, 1988).

For our purposes, there are two problems with the Rumelhart and McClelland model. First, the representation of linguistic forms is clearly inadequate. Pinker and Prince (1988) recognized many of the inadequacies. Probably the most serious is that distinct words may be represented in an identical fashion. Another piece of evidence suggesting that the phonological representations are wrong is the lack of constraints on the kinds of processes that the network can learn. For example, as Pinker and Prince point out, there is nothing preventing Rumelhart and McClelland's network from learning a rule which reverses the segments of the stem. But this sort of process is unknown in natural language.

Second, the task the network was trained on is not one that is necessarily involved in actual language processing: speakers and hearers need not turn present-tense or stem forms of verbs into past-tense forms. What they need to do is to turn meanings into past-tense or present-tense forms and present-tense or past-tense forms into meanings. Thus the fact that the Rumelhart and

McClelland network has acquired an (implicit) rule does not imply that that rule is usable in perceiving or producing words. The information in a lexicon should in fact be usable in both perception and production.

A related problem, brought out by Pinker and Prince, is the lack of an notion of "word" in Rumelhart and McClelland's system. This presents many problems, but in particular it prevents the system from distinguishing homophonous forms. There is, for example, no way that their network could learn separate past-tense forms for the verbs *break* and *brake*. Including meaning in the system, however, required to accommodate perception and production, could allow homophonous forms to be distinguished.

The model proposed in this paper addresses all three problems. Recent work by Elman (1988; 1989) on **simple recurrent networks** (SRNs) offers a more reasonable way of representing linguistic sequences. Briefly, inputs to an SRN consist of single events, which feed into a recurrent hidden layer. This layer may learn to function as a kind of short-term memory (STM). Following the presentation of a sequence, for example, a string of phonemes making up a word, the pattern on the hidden layer is then a kind of distributed representation of the sequence. The present model makes use of an SRN to represent words. As in most other applications of SRNs, we train the network on a prediction task: given a segment within a word, the network is to predict the subsequent segment. Our distributed phonological representations are in a sense more constrained than Rumelhart and McClelland's, and one hypothesis is that these constraints will translate into behavior that is constrained in human-like ways.

In addition, the proposed approach models language processing rather than the artificial task of generating forms from forms. Given a word, the system is to provide a meaning; given a meaning, it provides a word. This approach also has the potential to distinguish homophonous words.

The next section discusses the architecture of the proposed model. Following that, there is a description of a set of experiments designed to test it. Finally, some extensions of the model are discussed.

2. The model

Figure 1 shows the basic architecture of the proposed model. Because the network is designed to perform both production and perception, there are form and meaning units in both input and output layers. In a perception task, the network is given a sequence of phonological segments on the form input units and expected to generate an appropriate pattern on the meaning output units. In a production task, the meaning input units are clamped to a constant value while the network produces a sequence of patterns on the next segment output units representing a word.

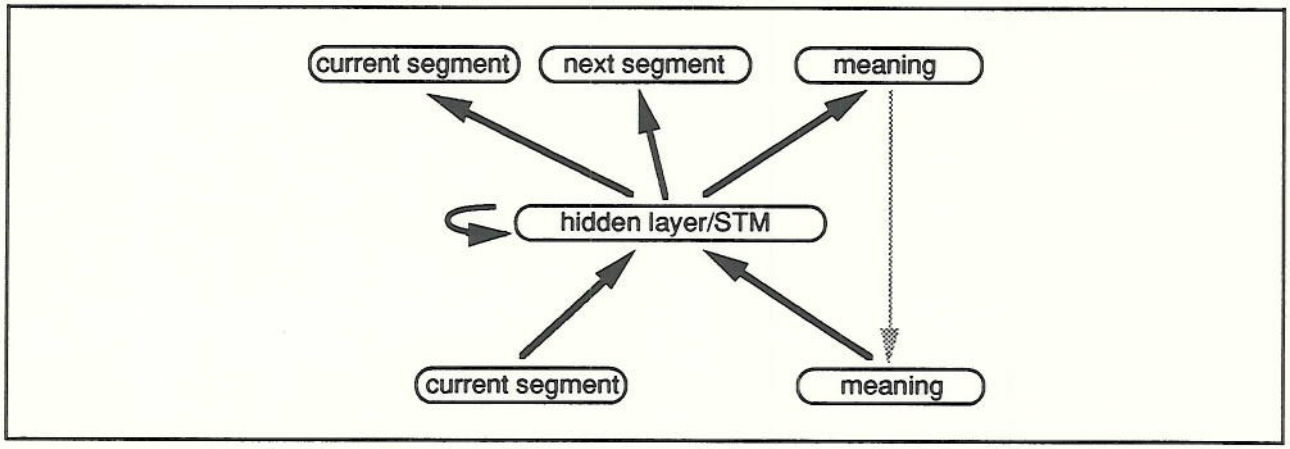

Figure 1: Network for Perception and Production of Words

The hidden layer has recurrent connections, as in other SRNs (Elman, 1988; Elman, 1989). That is, the pattern appearing on this layer at a given time step is part of the input to the network on the next time step. This permits the network to keep track of where it is within a word.

The network is trained using the back-propagation algorithm (Rumelhart, Hinton, &Williams, 1986). On most training trials, the network receives both form and meaning inputs and is provided with form and meaning targets. Such trials are potentially useful for both perception and production tasks. We have found, however, that for perception, the network also requires trials in which only the input form is provided. On these trials the pattern appearing on the output meaning units is copied to the input meaning units and appears as input on the next trial. Thus in these trials the network is actually being trained to perform a perception task (though there are targets for both form and meaning output units). For both types of trials, the network is trained to predict the next segment on the form output units. There is also a separate set of form output units which are trained to copy the current input form. This has been found to aid the network in learning to use the hidden layer as a short-term memory.

We are interested in the ability of such a system to learn simple morphophonemic rules. The simplest should be suffixing rules for which the form of the suffix depends on features of the previous segment because the relevant features are available on the current segment input. More challenging should be prefixing rules for which the form of the prefix depends on features of the following segment. While these features are not initially available in the STM because they have not appeared yet, they are implicit in the meaning of the word, which is available during production. One question we will want to address is whether this information is usable, that is, whether the network can be made to appear as though it has learned such a "right-to-left" rule. Finally, we would expect rules which do not involve a fixed stem to be most difficult. This is the case, for example, if the present and past-tense forms are mirror images of one another. Here we hypothesize that the network, like human learners, will find the rule especially challenging, if not impossible, to acquire.

3. Simulations

3.1. Stimuli

A set of experiments was conducted to test the effectiveness of this approach for the learning of morphological rules. Input words were composed of sequences of segments in an artificial language. Each segment was represented by a pattern over a set of 8 phonetic features. There were 15 possible phonemes. For each simulation, a set of 20 CVC syllables was generated randomly from the set of possible syllables. Twelve of these were designated "training" words, 8 "test" words.

For each of these basic words, there was an associated inflected form. For each simulation, one of three rules was used to generate the inflected forms. The "suffix" rule suffixed an /s/ or /z/ to the uninflected form; the suffix agreed on the voice feature with the previous segment. The "prefix" rule prefixed and /s/ or /z/ to the uninflected form; the prefix agreed on the voice feature with the following segment. The "reversal" rule reversed the segments of the uninflected form. For convenience, we will refer to the uninflected form as the "present" and the inflected form as the "past tense" of the word in question.

Each input word consisted of a present or past tense form preceded and followed by a word boundary pattern composed of zeroes.

Each "meaning" consisted of an arbitrary pattern across a set of 6 "stem" units, representing the meaning of the "stem" of one of the 20 input words, plus a single bit representing the "tense" of the input word (1 for past, 0 for present).

3.2. Training

During training each of the training words was presented in both present and past forms, while the test words appeared only in the present form. Thus there were 32 separate words in all.

80% of training trials ("perception/production trials") gave both form and meaning inputs and 20% ("perception trials") provided only forms.

For perception/production trials, the words were presented, one segment at a time, beginning with a word boundary, on the segment input units, and the appropriate meaning was provided on the stem and tense units. Targets specified the current segment, next segment, and complete meaning. When the last segment of a word appeared on the form input units, the network was trained to predict the word boundary output pattern.

For perception training, words were again provided one segment at a time, and correct stem meanings appeared on the stem input units. Input on the tense input unit was word-initially .5 and later the current activation on the tense output unit. Targets again specifed current segment, next segment, and complete meaning. Since the network was given stem meanings as input, it was actually being trained to produce only the tense portion of the meaning for each word.

There were 10 separate simulations for each of the three inflectional rules. Pilot runs were conducted to estimate optimum hidden layer size. This was 16 for the suffix rule, 24 for the prefix rule, and 21 for the reversal rule.

Training continued on each run until the mean sum-of-squares error per pattern was less than 0.05. This normally required between 50 and 100 training epochs.

To test whether the network had acquired the morphophonemic rules, the connection weights were frozen, and the network was tested in both the perception and production directions on the past tense forms of the test words. Production tests were conducted like the perception/production training trials. The network's output was taken to be that segment of the 16 possible (15 phonemes plus word boundary) which was closest to the pattern on the next segment output units. Perception tests were conducted like the perception training trials. Of interest was only whether the network could yield the correct tense following the presentation of the past-tense of a test word. The network's response was taken to be the rounded activation of the tense output unit following presentation of the final segment in the word.

3.3. Results

The network learned the set of training patterns for all three rules quite successfully. For production segments were produced correctly more than 99% of the time. For perception the correct tense was generated for 96% of the suffixed forms, 95% of prefixes forms, and 87% of the reversed forms.

Results for the production of past tense forms of test words are shown in Table 1. The network was very good at generating the suffixed forms, somewhat less so at generating the prefixed forms, and quite bad at generating the reversed forms.

	% Segments Correct	% Affixes Correct
Suffix	82.3	82.5
Prefix	62.0	76.3
Reversal	22.5	N/A

Table 1: Performance on Production of Test Items

Table 2 shows results for the recognition of tense in past tense forms of test words. Again performance was much better for the affixed forms than the reversed forms. Note that in the reversal case, 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 present and past forms of training words but only present forms of test words. Thus it saw more present forms overall and, given no evidence one way or the other, responds with an activation less than .5 on the tense output unit.

	% Tenses Correct
Suffix	79
Prefix	76
Reversal	13

Table 2: Performance on Comprehension of Test Items

To further investigate the ability of these networks to make generalizations in the prefix case, we trained a group of networks on a set of 70 stems in both present and past (prefix) forms. We began with a small hidden layer and increased the number of hidden units until the network showed evidence of learning the rule. This happened first with 8 hidden units. We were primarily concerned with the production direction. Following training on 1400 repetitions of each word, the network generated prefixes for all past forms, no prefixes for present forms, and correct voicing on the prefix for 68.6% of the past forms. The network was still very poor at generating the correct stem forms, however. Table 3 shows the pattern of activation across the 8 hidden units following the presentation of the initial word boundary for various categories of inputs. At this point the network has only the information in the stem meaning units to go on.

	1	2	3	4	5	6	7	8
All patterns	.00	.12	.99	.50	.00	1.00	.00	.07
Past patterns	.00	.12	.99	.00	.00	1.00	.00	.01
Present patterns	.00	.12	.99	1.00	.00	1.00	.00	.13
C1 +voice (target)	.00	.01	.99	.50	.00	1.00	.00	.06
C1 –voice (target)	.00	.23	.99	.50	.00	1.00	.00	.08
C1 +voice (output)	.00	.01	.99	.50	.00	1.00	.00	.07
C1 –voice (output)	.00	.66	.98	.50	.00	1.00	.00	.08

Table 3: Hidden Unit Activations following Initial Word Boundary (Simulation 2)

Clearly units 1, 3, 5, 6, and 7 are irrelevant at this point in the process, and unit 4 is responsible for distinguishing past from present patterns. It is the second unit that is apparently learning to encode voicing in the initial consonant. The table shows hidden unit activations for voiced and voiceless initial consonants for the correct outputs and for the actual outputs produced by the network. These differ considerably since the network was still not very good at the task. The activation of unit 2 is close to zero for patterns for stems beginning with a voiced consonant and high for stems which it "believes" begin with a voiceless consonant.

4. Discussion

The model shows clear evidence of having learned a morphophonemic rule which it uses in both the production and perception directions. Significantly, it is able to generate appropriate forms even when a "right-to-left" rule is involved, in the prefix case. That is, the fact that the network is trained only on prediction does not limit it to left-to-right rules because it has access to a meaning which permits the required "lookahead" to the relevant feature on the segment following the prefix. How is it that the meaning, which is only arbitrarily related to the shape of the stem, can make available the voicing feature on the initial consonant of the stem? Frankly, this is still something of a mystery to us. As Table 3 indicates, on the basis of the perception and production task the networks learn to classify stem meanings in terms of whether their initial consonants are

voiced or voiceless. Strikingly, this apparently happens even when, in the case of the test words in the first simulation, the voicing information was never required during training.

Equally significantly, the network finds it much harder to learn a reversal rule of a type which is apparently difficult for human language learners. Note that some aspects of the rule have been learned. Thus in 49% of the cases the network produces a CVC syllable as the past-tense form. What it cannot do is predict the correct consonants for the past tense. What is it that makes the reversal task so difficult? Consider what happens in the affix case for the production task. The input consists of of the sequence of segments representing the past-tense form of a word trained on only in the present, together with the stem meaning seen during training and the past tense not seen in this combination during training. Faced with this novel set of patterns, the network treats it as a blend of two sorts of patterns it has seen before, one involving the sequence of segments excluding the affix together with the stem meaning, the other involving the past tense input together with the feature of the form that determine the appropriate past tense form. The relevant phonetic feature is easy to come by in the suffix case since it is part of the current segment input. In the prefix case, as noted above, it is somehow available as an acquired property of the stem meaning. For the reversal case, if we think of the novel sequence of patterns as a set, rather than a sequence, there seems to be even more sharing with familiar patterns because exactly the same set of segments is involved. Crucially, however, the main task of the network is prediction of the following segment, and for the reversal rule there is no sharing at all between the present and past forms in terms of prediction. Patterns on the hidden layer develop in response to this task, so we expect little similarity between the STM inputs for the present and past forms. Thus the network has considerably less to go on in interpreting the novel reversed patterns than it does in the affix cases. With respect to form, it is more likely to base its response on similarity with a word containing a similar sequence of segments (e.g., /sek/ and /sem/) than it is with the correct related word that is the mirror image of itself.

5. Limitations and extensions

A number of features of the present approach remain to be tested. We have not treated both irregular and regular forms and hence can shed no new light on the question of whether networks can mimic the stages that humans tend to go through in making linguistic generalizations, an important issue in the discussions associated with the Rumelhart and McClelland model and the subsequent work by Plunkett and Marchman (Pinker & Prince, 1988; Plunkett & Marchman, 1989; Rumelhart & McClelland, 1986). Though we have no reason to believe the model will be unable to handle both regularities and irregularities, we have no way of knowing what stages it will pass through.

In future work we will also be concerned with the ability of the model to handle more complex phonological processes. Recently Lakoff (1989) and Touretzky and Wheeler (1989) have developed connectionist models to deal with complicated interacting phonological rules. While these models demonstrate that connectionism offers a valid alternative to conventional serial approaches to phonology, the models do not learn phonology; the knowledge is wired in from the start.

As noted above, it will also be of interest to analyze the representations that are built on the hidden layer. To what extent do similar patterns appear in the production and perception of a given word? Are there characteristic patterns corresponding to the two forms of the tense affixes?

There are important limitations to our approach. First, although networks of the type studied here are capable of yielding complete meanings given words and complete words given meanings, they have difficulty when expected to respond to novel combinations of known meanings or forms. Note that in the present study we have not demonstrated that the network can produce or comprehend correct novel forms. That is, in comprehension, we did not expect it learn to generate the stem meaning, and in production, we kept it on track by giving it the correct previous segment on each time step. It will be important to discover ways to make the system robust enough to respond appropriately to novel combinations of meanings and forms.

A further limitation concerns the plausibility of the task we have given the network. Human language learners do not have the luxury of learning words in isolation. We are currently experimenting with a version of the basic network shown in Figure 1 which takes individual words on its form inputs and distributed representations of sentence semantics on its meaning inputs. Work by Elman (1989) already indicates that SRNs given words as input can become sensitive to syntactic structure. While we are having some success with this network, it does not solve the ultimate problem, which is that of learning a lexicon/grammar given inputs in the form of phonological segments (or better, acoustic time slices) rather than complete words. This seems to require a more complex version of the network of Figure 1, one with a subnetwork dedicated to phonological inputs and syllables and another subnetwork dedicated to words and phrases. One fundamental problem is what sort of prediction task to train the latter on. That is, given inputs that are not segmented into words, how is the network to learn to predict the next word? We are developing a version of the model that would train the word/phrase subnetwork to predict the state of the hidden layer in the phonological subnetwork at the next stress prominence.

6. Conclusions

It is by now clear that a connectionist system can be trained to exhibit rule-like behavior. What is not so clear is whether networks can learn the sorts of rules necessary for compositional semantics, whether they can discover how to map constituents of form onto constituents of meaning and to use this knowledge to interpret and generate novel forms. These sorts of rules seem to require the kind of constituency which is not available to networks making use of distributed representations, and this presumed deficiency was behind the influential critique of Fodor and Pylyshyn (1988).

The present study is an initial attempt to demonstrate that networks may not be as limited as has been previously though. We have shown that, given "meanings" and temporally distributed representations of words, a network can learn to isolate stems and affixes, associate them with their meanings, and, in a somewhat limited sense, use this knowledge to produce and interpret novel forms.

References

Elman, J. L. (1988). *Finding structure in time* (Report No. 8801). La Jolla, CA: University of California, San Diego, Center for Research in Language.

Elman, J. L. (1989). *Representation and structure in connectionist models* (Report No. 8903). La Jolla: University of California, San Diego, Center for Research in Language.

Fodor, J. A., & Pylyshyn, Z. W. (1988). Connectionism and cognitive architecture: A critical analysis. *Cognition, 28*, 3-71.

Lakoff, G. (1989). A suggestion for a linguistics with connectionist foundations. In D. Touretzky, G. Hinton, & T. Sejnowski (Ed.), *Proceedings of the 1988 Connectionist Models Summer School* (pp. 301-314). San Mateo, CA: Morgan Kaufmann.

Pinker, S., & Prince, A. (1988). On language and connectionism: Analysis of a parallel distributed processing model of language acquisition. *Cognition, 28*, 73-193.

Plunkett, K., & Marchman, V. (1989). *Pattern association in a back propagation network: Implications for child language acquisition* (Report No. 8902). La Jolla, CA: University of California, San Diego, Center for Research in Language.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning internal representations by error propagation. In D. E. Rumelhart, J. L. McClelland, & the PDP Research Group (Ed.), *Parallel Distributed Processing: Explorations in the microstructures of cognition, Vol.1: Foundations* (pp. 318-362). Cambridge, MA: MIT Press.

Rumelhart, D. E., & McClelland, J. L. (1986). On learning the past tenses of English verbs. In J. L. McClelland, D. E. Rumelhart, & the PDP Research Group (Ed.), *Parallel Distributed Processing. Explorations in the microstructures of cognition: Vol. 2: Psychological and biological models* (pp. 216-271). Cambridge, MA: MIT Press.

Touretzky, D. S., & Wheeler, D. W. (1989). *A connectionist implementation of cognitive phonology* (Report No. CMU-CS-89-144). Pittsburgh: Carnegie Mellon University, School of Computer Science.

van Gelder, T. (forthcoming). Compositionality: A connectionist variation on a classical theme. *Cognitive Science*.