인공신경망을 이용한 한국어 형태음운현상 연구

이 찬 도†

요 약

언어에서 단어가 차지 하는 중요성은 매우 크다. 그럼에도 불구하고 단어를 구성하는 음운론적, 형태소론적 요소에 관한 계산적 연구는 그리 많지 않다. 대개의 전통적 언어학 이론은 추상적인 기저구조와 일련의 명시된 규칙들을 가정함으로 해서 형태음운현상을 설명한다. 그러나 이러한 접근방법은 (1) 기저구조의 가정, (2) 규칙의 발견, 그리고 (3) 규칙간의 상호관계 등에서 문제점을 내포하고 있다. 본 연구는 인공신경망이 단어를 구성하는 음소열과 그 단어의 의미를 학습하는 과정에서 규칙은 생겨난다는 가정에서 시작한다. 다양한 국어의 형태음운현상에 대한 실험결과는 인공신경망이 규칙이나 기저구조의 도움없이 형태음운현상을 학습할 수 있음을 보여준다.

A Study of Morphophonemic Processes of Korean using Neural Networks

Chan-Do Lee†

ABSTRSCT

Despite their importance in language, there have been relatively few computational studies in understanding words. This paper describes how neural networks can learn to perceive and produce words. Most traditional linguistic theories presuppose abstract underlying representations (UR) and a set of explicit rules to obtain the surface realization. There are, however, a number of questions that can be raised regarding this approach: (1) assumption of URs, (2) formation of rules, and (3) interaction of rules. In this paper, it is hypothesized that rules would emerge as the generalizations the network abstracts in the process of learning to associate forms with meanings of the words. Employing a simple recurrent network, a series of simulations on different types of morphophonemic processes was run. The results of the simulations show that this network is capable of learning to perceive whether words are in basic form or in inflected form, given only forms, and to produce words in the right form, given arbitrary meanings, thus eliminating the need for presupposing abstract URs and rules.

1. Introduction

Understanding language is hard. It requires not only language-specific knowledge, but also knowledge about the surrounding environment. There are a lot of research efforts directed to recognizing raw speech data in the name of speech recognition. Much of natural language processing (NLP) research has been dedicated to understanding written text. Yet there have not been enough studies in understanding words themselves, a major part of language. How do we know how to say a word when we intend to utter something? What makes us understand the meaning of a word when we hear it? What is it that facilitates associating a word with its meaning? There certainly is linguistic knowledge in our brains that makes these processes possible. Some involve phonology, some morphology, and some require morphophonemics, an interaction between the two to explain the phenomena. We should be able to develop a com-

· This paper was supported(in part) by NON DIRECTED RESEARCH FUND, Korea Research Foundation, 1993.
† 종신회원 : 대전대학교 정보통신공학과 조교수
논문접수 : 1994년 11월 14일, 심사완료 : 1995년 3월 17일

putational device that can accommodate them. Without this middle ground our efforts to build a computer system that can understand human languages will be futile. We must develop a system that fills the gap between speech recognition and higher-level NLP in order to produce a reasonable speech understanding system. This paper is a small, but hopefully a right step in this direction.

Most computational approaches to NLP to date have come from the traditional symbolic perspective. In this view, the basic building blocks of the language and thought are discrete symbolic entities that are manipulated by a set of rules interpreted by a central control mechanism. Despite some success, especially concerning knowledge representation, inference and syntactic parsing, symbolic NLP has not fulfilled its promise, even after 30 years of study.

The weaknesses of the symbolic approach to NLP can be remedied by employing a connectionist approach. First, connectionist models are much more robust with regard to noise. Second, connectionist networks work as parallel constraint satisfaction; processing in connectionist models involves attempts to satisfy as many constraints as possible. This is what is required for language comprehension. Third, learning is fundamental to connectionist models. They can learn from exposures to examples, thus eliminating the need for the use of a priori rules. Fourth, it can handle the temporal nature of language more easily.

This paper describes how neural networks can learn to perceive and produce words. The next section raises the problem of words and computation. Relevant literature is reviewed, similarities with the approaches that are taken in this paper are pointed out, and drawbacks are singled out. The new approach called "performance grammar" employed in this paper is introduced. Section 3 describes the model in detail, including the network architecture used, and why this particular architecture was chosen. Section 4 describes the experiments performed. It shows the stimuli used, explains how the system was trained, and summarizes the results. Section 5 discusses the results of the experiments. Also the limitations of the current study are criticized and are related to directions for future research. Finally, Section 6 discusses the achievement made by this research and outline of future research problems suggested by this study.

2. The Problem

2.1 Raising the Problem

Studying computational accounts of phonology and morphology has not attracted many AI researchers, largely due to the fact that the apparent rule-like patterns exhibited by phonology and morphology did not appeal to the practitioners in traditional symbolic AI as an interesting problem. Since most work has been on English, which does not have much in the way of interesting morphology (unlike Korean), most problems seem too easy to implement.

The other reason might be the position of phonology and morphology as in between NLP and speech recognition. NLP researchers have used words as their tools and did not bother to go down a step further. Phonology is not relevant in written language, the focus of NLP. On the other hand, researchers in speech recognition have concentrated only on real low-level signals. Yet, it should be noted that an NLP system that deals with a language like Korean must pay attention to morphology; it is simply too expensive to store all the variant forms in the lexicon. As for phonology, if the language in addition involves complex morphophonemic processes at the boundaries between morphemes (as in Korean, which has very extensive inflections), then the system also needs to handle these processes.

Studying phonology and morphology is also interesting because of the parallel between syntax and sentence semantics and morphology and word semantics. Both involve compositionality and the problem of segmenting the input. But morphophonemics is perhaps simpler to study; for one thing, it seems not to involve recursion.

In this paper I study morphophonemic processes with the aid of a neural network, hoping to bring forward enough convincing results to shed some new lights on the language processing research community.

2.1.1 Words and Computation

Studying words is important and also very interesting, since the word is the central unit in language where the phonological and semantic poles come together. Words are the smallest unit in the language hierarchy that embrace both meaning (semantic pole) and speech sounds (phonological pole) and can stand alone. Words are also the dividing line between low-level linguistic studies in phonology and morphology and higher-level studies in syntax, semantics, and pragmatics.

For my purposes, a word is not just a written piece of text such as *ssal* (쌀) 'rice', or *sagwa* (사과) 'an apple', but a series of phonetic segments associated with a meaning, for example, the word with the meaning of *rice* is represented as

(1) /s/ RICE
(2) /ɑ/ RICE
(3) /l/ RICE

where the items in uppercase represent meaning and the expressions with phonetic characters surrounded by slashes refer to the word. My stimuli in the experiments were presegmented signals labeled with phonetic features. This was necessary to bridge the gap between two related, yet firmly divided fields of study: speech recognition with raw speech signals and high-level NLP with written text. The level of words in my study is higher than the raw speech signal, yet lower than the written text, thus taking advantage of the ease of processing associated with written language understanding and not falling into the trap of ignoring real speech signals. Even though I am not using real speech data, what I am dealing with here is spoken language; written language does not reflect the phonology that is one of my concerns.

2.1.2 Phonology, Morphology, and Morphophonemics

Phonology is the study of the systems underlying the selection and use of sounds in the languages of the world. It focuses on the internal representation of sound units and tries to explain the nature of phonological phenomena. It deals with the sequential and phonetically conditioned patterning of sounds in language. Phonology plays a very important role in the perception and production of a word. For example, in order to recognize a nominative suffix as such, a Korean listener presumably needs to "know" the consonant deletion rule as well as the nominative case. The same applies to speech production. Without knowing the consonant deletion rule a speaker could not possibly add the right suffix to a novel noun, and would fail to produce the right nominative form.

Morphology is the study of word structures. Even though the processes of word formation may vary from one language to another, all languages have the means to create new words by combining morphemes and therefore exhibit the rule-governed creativity that seems to be typical of human languages.

The interaction between the phonological and the morphological components of grammar is reflected by the presence of allomorphs. One example of allomorphic variation is the Korean nominative suffix. The Korean nominative morpheme has two variants: /ɨn/ (은) and /nɨn/ (는), depending on the final segment in the noun stem. For example, the nominative case of *pa* (파) 'green onion' is pronounced /panɨn/ (파는) and that of *kal* (칼) 'a knife' is /kalɨn/ (칼은). To account for these variants in traditional phonology, one must posit an underlying abstract representation, and one or more rules have to be invoked which transform the UR into either /ɨn/ (은) or /nɨn/ (는) given the correct environment. The derivation of a form like *kalŭn* (칼은) 'knife + nominative' begins with the UR of the morphemes *kal* (칼) and *nominative*, and the rules turn these into the surface form /kalɨn/ (칼은).

2.1.3 The Challenge

In this paper I concentrate on the issue of learning to perceive and produce words in terms of morphophonemic processes which involve both phonology and morphology.

Morphophonemic processes are generally regarded as symbolic, and rule-governed. Accounting for such processes using neural networks presents many challenges that have been noted by researchers who subscribe to classical symbolic tenets, including Pinker and Prince [1] and Fodor and Pylyshyn [2]. According to their arguments, connectionist models cannot command the compositional semantics that is supposedly essential to NLP. My study is an attempt to show that neural networks are indeed capable of dealing with compositionality. The networks were taught to map combinations of meanings onto combinations of forms, and then decide which part goes with which. They had to discover how to map constituents of form onto constituents of meaning and to use this knowledge to interpret and generate novel forms.

In classical symbolic systems the acquisition of URs has been difficult to account for, and this fact has in part motivated the idea of innate predispositions for certain linguistic structures. The very existence of URs that can be manipulated by rules was one of the points Pinker and Prince [1] and Fodor and Pylyshyn [2] made against the adequacy of connectionist systems in explaining morphological processes. If we can overcome these difficulties by explaining some morphophonemic rules without the benefit of any *explicit* rules, or URs, it will strongly support the appropriateness of neural networks to the task of language processing.

2.2 Related Work

Computational accounts of phonology and morphology were mostly ignored by researchers in AI until the early 1980's, when Koskenniemi [3] noticed the usefulness of the state transducers and developed a two-level morphology model. His work influenced later research in computational morphology. Since 1988, when George Lakoff [4] called the attention of the connectionist community to phonology as a challenging problem, there has been some fruitful research directed towards the problem of phonology. In this section a brief overview of some of related work both in phonology and morphology will be presented.

2.2.1 Models of Past-tense Morphology

Rumelhart and McClelland [5] (hereafter referred to as "RM") showed that a simple two-layer pattern associator can acquire the marking of the past tense in English. Their model maps representations of present tense forms of English verbs onto their past tense versions without any involvement of semantic characterization. The RM model has been throughly scrutinized and criticized by [1] and it was the beginning of many heated discussions among AI researchers that followed. RM does not model actual language processing as it occurs in human beings:their models merely map representations of present tense forms of English verbs onto their past tense versions without any involvement of semantic characterization at all, whereas human beings accomplish the task of turning a form to meaning and meaning to form.

2.2.2 Cognitive Phonology

Cognitive phonology [4], a development of the ideas of [6], is an effort to eliminate the need for rule orderings. Ordered-rule interaction is a necessary foundation in the standard versions of conventional generative phonology. It uses a multilevel representation for the utterance, to which multiple rules may apply in parallel.

There are no explicit rule orderings; all constructions at a given level apply simultaneously. However, by allowing both inter-and intra-level constructions, the theory achieves the effect of extrinsic rule orderings. Touretzky and Wheeler have developed a connectionist implementation of Lakoff's ideas, making some modifications along the way [7]. Despite its powerful mechanisms that can deal with complex rule interactions, Touretzky and Wheeler's model still shares several points with standard models: (1) the assumption of abstract levels and underlying forms, (2) the presence of hard wired rules, and (3) the nature of the model as a *competence model*.

2.2.3 Production Models

Hare's work [8], Chauvin [9], and Dorffner [10] are very similar to my work reported in this paper. Hare's view is that prosodic structure, such as vowel harmony, is the sum of the generalizations a speaker abstracts in the process of learning, without the benefit of the rules. My work is similar to Hare's work, but it differs in two ways. First, I investigate the power of a somewhat different type of network. Second, I am concerned with both production and perception as achieved by a single network that can accommodate both processes.

Chauvin examines empirical results related to first-word acquisition in infants and explores how and why similar phenomena occur in a PDP model. The basic architecture of the network allows encoding of labels and "images" in a common level of representation and subsequent extraction of labels from images and images from labels. This task of comprehension and production applied to a neural network is very similar to my approach presented in this paper. The difference lies in the representation of linguistic forms. Chauvin concentrates on semantics ignoring phonology, and so does not need a dynamic representation of words, while the particular problem domain I have been exploring, requires a dynamic network.

Dorffner shows how the interpretation and generation of words can be modeled in a connectionist model. His model has an architecture that learns words by categorizing sensory input from two channels, "phonetic" and "visual," into concepts, and then associates co-occurring concepts from the two realms into arbitrary words or symbols. Like Chauvin, he also is not concerned with phonology.

2.2.4 Parametric Stress Model

Chomsky [11] argues that a child is equipped with biologically programmed innate capacity to acquire and utilize a linguistic system. The task of language acquisition is thus simply to finetune the particular aspects of the linguistic system to which the child is exposed. Based on this theory, Dresher and Kaye [12] constructed a parameterized, symbolic model for the acquisition of stress systems. This model assumes the existence of 11 preset parameters as a part of universal grammar; the task of the model is to fix the value of each parameter. So much given information makes us wonder whether a child really learns a stress system this way. Gupta and Touretzky [13] approached the same problem using neural networks without having "built-in" constructs of metrical theory, thus concluding that the Chomskyan view of language acquisition may be unwarranted.

2.2.5 Two-level Morphology

Koskenniemi[3] puts forward a morphology in which all rules apply simultaneously, and in which each rule can be compiled into a finite state transducer. The name "two-level morphology" reflects the setup where only the underlying lexical and the surface levels ever "exist," thus avoiding complicated and often troublesome rule interactions.

Two-level morphology avoids rule interactions, since rules (the automata) work together in parallel; a configuration is accepted if all rules pass. One contradicting rule is enough to ruin the correspondence.

Although two-level morphology succeeds in avoiding rule interactions, it still lacks two fundamental elements of a realistic model: (1) rules have to be hand-coded and supplied by a designer and thus (2) no learning takes place in the model.

2.2.6 Summary

So far, some of the models that deal with computational phonology and morphology have been surveyed. Some of the models lack learning, while others are not psychologically plausible.

My model is an attempt to overcome many of the problems raised by the models mentioned above by having both segmental and semantic inputs and outputs in the network, thus performing the more psychologically plausible process of the production of a sequence of segments given a meaning or the selection of a meaning given a sequence of segments. My model is a *dynamic* model. The model is given one segment at a time as input. Furthermore, my model does not presuppose any linguistic parameters, URs, or rules.

2.3 "Performance" Grammar

There are various theories about how and why some observed phonological phenomena occur in the way they do. However, most traditional theories presuppose abstract URs and a set of explicit rules to obtain their surface realizations. Modern generative grammar is based on the notion of "deriving" forms through the application of a series 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. Most classical symbolists believe that surface forms are really derived from URs with the application of rules. Pinker and Prince's [1] critical analysis of the RM past tense model is crucially based on the claim that the linguistic and developmental facts provide good evidence for rules and URs.

There are, however, a number of questions that have been raised regarding this approach. That is, given only surface input forms together with meanings inferable from context, how is a learner to figure out how the form-meaning relation gets mediated by URs? Where do assumptions of URs come from? How are rules found and how are they related to each other?

It is customary to assume that a language learner is helped by having certain predispositions about language wired in; however, I begin with an approach which is far more constrained. I 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. First, we need some means of representing patterns that take place in time, that is, we need my model to have the capacity to develop a kind of short-term memory that preserves past history.

Second, we need a means of handling both meanings and forms. Throughout this paper, by "form" I mean a series of phonemes, while by "meaning" I mean the lexical entry of the word in question, together with relevant grammatical features. For example, the phrase pronounced */panɨn/* (파는) has the meaning *GREEN ONION + NOMINATIVE*. What we need is a mechanism that can incorporate both form and meaning in such a way that the knowledge that is learned is potentially usable in both perception and production tasks.

What I am describing in this section is a rudimentary sort of "performance" grammar; its goals are different from the goals of generative grammar, which is "competence" grammar. I am arguing not against rules and URs per se, ***but against*** generative/transformational *explicit* competence rules and URs; I argue *for* the *implicit* performance rules and URs modeled in a connectionist architecture. For example, the fact that the nominative of *kal* is pronounced as */kalɨn/* (칼은) can be explained in competence grammar as:

/kalnɨn/ Underlying Representation
/kalɨn/ Consonant ("n") deletion
[kalɨn] Surface Form

In contrast, performance grammar does not require any explicit rules and URs. These are to be learned in the process of generalizing over a set of training instances that exhibit the alternations between stem and stem plus nominative case, for example,

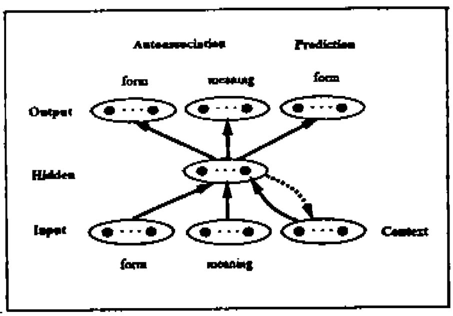
KNIFE+ZERO-CASE→ */kal/*
KNIFE+NOMINATIVE→ */kalɨn/*.

I think my approach is more psychologically plausible; and if such an approach can handle perception, production, and acquisition, then generative grammar would become superfluous. I do not believe that each underlying segment goes through a derivation employing *explicit* rules to produce a "surface" segment. More important is the fact that for speakers meanings trigger the correct phonological/phonetic productions, while for listeners phonetic/phonological material directly evokes the correct word meanings.

3. The Model

When designing the model, I considered the capability to accommodate temporal processing to be one of its most important features. Since the morphophonemic processes that I am studying in this paper are temporal, I need some form of short-term memory in which to store the previous events.

I used a relatively constrained three-layer network, one in which feed-forward connections are supplemented by limited feedback connections. (Fig. 1) shows the network architecture used.



(Fig. 1) Architecture of the network used

The input layer consists of three cliques of units: the Form clique, the Meaning clique, and the Context clique. The Form clique in the input and output layers each consists of 13 units representing a phonological segment. Each Meaning clique consists of seven units, six of which represent a stem meaning (hereafter referred to as *s-meaning*) and one of which represents the grammatical feature (hereafter *g-feature*; by *meaning*, I refer to *s-meaning* and g-feature together) of the word. The network has a variable number of hidden units and an equal number of Context units. Each of the first two cliques receives input from the outside, while the Context units receive a copy of the activations on the hidden layer from the previous time step. Given the current form and meaning, the network is trained to replicate them on one part of the output layer (autoassociation) and to predict what comes next in the sequence (prediction). Backpropagation [14] is used to train the network.

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

There are many reasons why the particular architecture was chosen. The model is a slightly modified version of the simple recurrent network (SRN) developed by Elman [15]. It has the ability to learn the kind of temporal processing that is a prerequisite to the perception and production of words, by storing the past history on context units. The model was trained on autoassociation as well as prediction. Autoassociation was used to force the network to distinguish the different input patterns on the hidden layer. The task of prediction exactly matches that of production, since the system is trying to decide which phonological forms to produce following the current one. By incorporating both Form and Meaning units, the model has the ability to associate form with meaning as well as meaning with form, making it possible for it to learn to perceive and produce words.

4. Experiments

In a series of experiments, networks with the architecture described above were trained on various morphophonemic forms.

The results indicate that the network used is capable of learning morphophonemic rules by encoding them on the connection weights and using them in perception and production tasks. That is, given training on the stem form, but not the nominative case of *kal* (칼) 'a knife', the network was later able to generate the appropriate nominative suffix following the stem or to determine the grammatical case of */kalɨn/* (칼은), a form it had never seen.

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

(4) KNIFE+ZERO-CASE→/kal/
(5) KNIFE+NOMINATIVE→/kalɨn/
(6) NECK+ZERO-CASE→/mog/

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

(7) NECK+NOMINATIVE→/mog/+??.

However, this solves the problem in only the production direction. The model should also be able to predict meanings, given forms. The model was trained on (8), (9), and (10) and tested on (11) to see if it was able to get the correct grammatical ending.

(8) /kal/→KNIFE+ZERO-CASE
(9) /kalɨn/→KNIFE+NOMINATIVE
(10) /mog/→NECK+ZERO-CASE
(11) /mogɨn/→NECK+??.

4.1 Stimuli

Input words were composed of sequences of segments. Since morphophonemic processes do not treat phonemes as atomic, unanalyzable wholes but refer instead to their constituent phonetic properties like voicing, tenseness for vowels, and tongue position, it was necessary that such fine-grained information be present in the network. To meet this end, each segment was represented as a binary vector encoding modified Chomsky-Halle phonetic features [16, 17]: 1 for the presence of a particular feature and 0 for its absence.

Twenty words were selected for each simulation. Twelve words from each set were designated "training" words; the remaining eight were "test" words. For each of these basic words, there was an associated inflected form. The network was trained on both the basic and inflected forms of the training words and only on the basic forms of the test words. Therefore there were 32 different input patterns for each simulation. Words were presented one segment at a time. Each word ended with a word boundary pattern consisting of all zeroes.

Each "meaning" consisted of an arbitrary pattern, composed of three activated units on out of a set of six "stem" units, representing the meaning of the "stem" of one of the 20 input words plus a single unit representing the grammatical feature of the input word (0 for basic, 1 for inflected).

4.2 Training Regimen

Two separate simulations were performed for each of the morphophonemic rules. Pilot studies were performed to estimate the optimum size of the hidden layer.

The meaning inputs and the target meaning outputs were constant throughout the presentation of a word. The network was trained on autoassociation for both forms of inputs, that is, both segments and meanings, as well as on prediction for the next segment.

To accelerate the training time *adaptive training* [18, 19] was used. The learning rate began at 0.25 and was decreased by a factor of 0.75 when there was no improvement over a period of five epochs. A fixed momentum of 0.9 was used.

To test the network's performance on the production task, the network was given the appropriate segments for the stem successively, along with the meaning of that stem and the g-feature unit on indicating inflection. Then the prediction output units were examined at the point where the inflected morpheme should appear. Using Euclidean distances, each output pattern was converted to the nearest phoneme. This phoneme was then compared to the desired phoneme.

To test perception performance, the network was given the sequence of input segments for a word, the stem meaning units were set to the appropriate pattern, and the g-feature unit was given the initial value of 0.5. At the presentation of each new segment, the g-feature unit was copied from the output on the previous time step. The output g-feature unit was examined after the appearance of either the appropriate inflected form or the word boundary.

4.3 Experiment 1: nominative case in/nin (주격 조사 은/는)

4.3.1 Method

The "suffix" rule attaches a /nɨn/ to the noun stem for the nominative case and "n-deletion" rule deletes /n/ from the suffix when the noun ends with a consonant to avoid consonant clash; In summary, this process adds a /nɨn/ or an /ɨn/ to the noun and the suffix agrees on the -consonantal feature with the previous segment.

The input corpus for this simulation consisted of a set of 20 Korean nouns. Twelve of these were designated "training words", while the other eight were "test" words. The network had 27 hidden units and also 27 Context units. Training continued until the network performed correctly on the training set, that is, until the error for every output was less than 0.05: 696 epochs for the first run; 6250 for the second.

4.3.2 Results

The results of this experiment are summarized in ⟨Table 1⟩.

⟨Table 1⟩ Results of nominative case experiment

	Production		Perception		n
	zero-case	nominative	zero-case	nominative	
Training	100%	100%	100%	100%	24
Testing	100%	87.5%	100%	100%	16

When the network was tested on the nominative case task, the network predicted the correct segments for 14 of the 16 suffixes in the two runs. The two suffix errors involved substituting /b/ for /n/ in /hyonɨn/ (효는). When the nominative morpheme appeared in the perception task, the output number unit behaved appropriately for all of the 16 test items.

4.4 Experiment 2: L-deletion (ㄹ 탈락)

4.4.1 Method

In this rule type the consonant /l/ is deleted from the end of a verb stem when it is followed by a */ni/*, */si/* etc. The input corpus consisted of a set of 20 Korean verbs, all of which ended with */lda/*. The first twelve words were "training" words and the remaining eight, "test".

The network performed best with 17 hidden units and the same number of Context units. The network was trained 751 epochs for one run and 2357 epochs for the second run.

4.4.2 Results

The network was able to learn the l-deletion task very well. It produced /*lda*/ when the g-feature unit was off and /*ni*/ when the unit was on for the training words. The results for the test words are shown in ⟨Table 2⟩. It correctly produced /*ni*/ for the test words when it was given only /*lda*/ during training. It was very good at perceiving a novel word with /*ni*/ as an inflected word.

⟨Table 2⟩ Results of deletion experiment

	Production	Perception	n
	% Segments correct	% Feature correct	
l-deletion	100	100	16

4.5 Experiment 3: b-irregular (ㅂ-불규칙)

4.5.1 Method

This rule changes a /*b*/ in a verb stem to a /*w*/ when it precedes a /ɑ/, /ə/ etc. The input corpus consisted of a set of 20 Korean verbs. Each verb ended with /bda/, which was changed into /wɑ / or /wə / depending on the preceding vowel.

As the previous experiments, after the pilot run the size of hidden layer was decided upon: the network performed best with 27 units in the hidden layer. The network was stopped after 501 training epochs for one run and 232 epochs for another.

4.5.2 Results

The network learned the set of training words quite successfully in less time than those for other two experiments. Segments were produced correctly and the network correctly predicted grammatical feature.

The network apparently had difficulty in coming up with the correct morpheme for the b-irregular task. The network predicted the correct segments /*wɑ*/ or /*wə*/ only 75% of the time (only 12 cases out of the 16 words in the two runs were correct). This is not surprising given the task of substituting a segment. The network performed quite successfully on the perception task. It was able to perceive a word ended with /*wɑ* / or /*wə* / as an inflected word. The results are summarized in ⟨Table 3⟩.

⟨Table 3⟩ Results of mutation experiment

	Production	Perception	n
	% Segments correct	% Feature correct	
b-irregular	75.0	100	16

5. Discussion

The model successfully learned morphophonemic rules by making generalizations from the exemplars by changing the connection weights during the process of learning. The set of weights the model developed in producing the desired inflected morphemes constrained the model's outputs to follow the desired patterns, and what looks like a rule-governed behavior is in fact embodied in these weights. However, the network does not yet achieve all that I would like.

The experiments reported here were carried out on only a small, and severely restricted input corpus. Only 20 words were considered in each simulation run. To be able to claim the plausibility of this model as an adequate system that can process morphophonemic phenomena, I need to expand my research to a bigger data set.

There were some unrealistic procedures employed in training. The network was given only noiseless input data. However, in reality, it is not possible to receive all words without any noise at all. The network was also given a priori word boundaries. A more realistic task would be to present the words in succession without any boundaries, so that the network would learn to detect the word boundaries and process them accordingly. If the model were given raw speech data, the two procedures of filtering and segmentation of some sort would need to play very important roles in perception [20, 21].

One of the severe disadvantages of the current model is that the network was hand designed, especially in that the size of the hidden layer had to be empirically determined. This is one of the biggest drawbacks inherent to most connectionist models. My distant goal is to design a self-evolving network: given only the problem, the model would come up with the optimal network.

6. Conclusion

The model is capable of abstracting generalizations from exemplars, thus eliminating the need for pre-defined abstract URs and explicit rules, a major part of generative phonology.

I believe my study has shed some light on how a particular type of cognitive phenomenon can be accounted for without the use of explicit symbols or rules. It remains to be seen how much my approach to the relatively trivial processes dealt with must be modified to deal with more complex processes and the elaborate mechanisms for handling them posited by traditional phonologists and their connectionist descendants.

Words are composed of smaller units and there is a hierarchy from more abstract units to less abstract units. In the current study, my main concern was with the areas of phonology and morphology. I presented feature matrices corresponding to single phonemes as input to the network and showed that the network could learn morphophonemic processes, thereby perceiving and producing the correct words given the appropriate form or meaning. It is safe to assume that the activations which follow the presentation of a feature matrix segment compromise a distributed representation of that phoneme. Following a cluster analysis demonstrating the network's natural categorization of phonemes, we might be able to use these internal representations as input for another higher level network, thus constructing representations for syllables, an issue which I bypassed in the current study. Even though the network is told nothing about where syllable boundaries occur, how many syllables there are in a given word, or even that syllables exist, it might be able to encode and use the syllable structures.

A related problem is that, I presented words in isolation, as presegmented signals labeled with phonetic features, which is not the case for real speech data. Input should be an uninterrupted stream of raw speech data. It should filter out noise from the speech signal, and segment the signal into different categories (phonemes) according to appropriate (phonetic) features. It should discover for itself the existence of the basic units of language, including the syllable, the morpheme, and the word. It should analyze distributed representations on the hidden layer, use them to feed the next level in the hierarchy, and add new levels as need arises. Once it has finished being trained in perceiving the input signal, the resulting network should then be able to perform production tasks. The quest for these answers will be my ultimate goal in the process of my research. The study presented in this paper is a right step in this direction and hopefully will be a significant component of a complete speech·understanding system.

References

[1] Pinker, S. & A. Prince, "On language

and connectionism : Analysis of a parallel distributed processing model of language acquisition," Cognition, 28, pp. 73-193, 1988.

[2] Fodor, J. & Z. Pylyshyn. "Connectionism and cognitive architecture: A critical analysis," Cognition, 28, pp. 3-71, 1988.

[3] Koskenniemi, K., Two-Level Morphology: A General Computational Model for Word-Form Recognition and Production (Publication No. 11), Department of General Linguistics, University of Helsinki, Helsinki, Finland, 1983.

[4] Lakoff, G., "Cognitive phonology," Paper presented at the Annual Meeting of the Linguistics Society of America, 1988.

[5] Rumelhart, D. & J. McClelland, "On learning the past tense of English verbs," J. McClelland and D. Rumelhart (Eds.), Parallel Distributed Processing, Vol. 2, pp. 216-271, MIT Press, Cambridge, 1986.

[6] Goldsmith, J., "Phonology as an intelligent system," D. Napoli, and J. Kegl (Eds.), Bridges between Psychology and Linguistics: a Swarthmore Festschrift for Lila Gleitman, Lawrence Erlbaum Associates, Hillsdale, 1991.

[7] Touretzky, D. & D. Wheeler, "A computational basis for phonology," D. Touretzky (Ed.), Advances in Neural Information Processing Systems 2, pp. 372-379, Morgan Kauffmann, San Mateo, 1990.

[8] Hare, M., "The role of similarity in Hungarian vowel harmony : a connectionist account," Connection Science, 2, pp. 123-150, 1990.

[9] Chauvin, Y., Symbol acquisition in humans and neural (PDP) networks, PhD dissertation, University of California, San Diego. 1988.

[10] Dorffner, G., A sub-symbolic connectionist model of basic language functions, PhD dissertation, Indiana University, Bloomington, IN. 1990.

[11] Chomsky, N., Aspects of the Theory of Syntax, MIT Press, Cambridge, 1985.

[12] Dresher, B. & J. Kaye, "A Computational Learning Model for Metrical Phonology", Cognition, 34, pp. 137-195, 1990.

[13] Gupta, P. & D. Touretzky, "Connectionist Models and Linguistic Theory: Investigations of Stress Systems in Language," Cognitive Science, 18, pp. 1-50, 1994.

[14] Rumelhart, D., G. Hinton, & R. Williams, "Learning internal representations by error propagation," D. Rumelhart and J. McClelland (Eds.), Parallel Distributed Processing, Vol. 1, pp. 319-362, MIT Press, Cambridge, 1986.

[15] Elman, J., "Finding structure in time," Cognitive Science, 14, pp. 179-211, 1990.

[16] Chomsky, N. & M. Halle, The Sound Pattern of English, Harper and Row, New York, 1968.

[17] Huh, Woong, Gugoumunhak (Korean Phonology), 2nd Ed., Saem Publishing Co, Seoul, 1989.

[18] Fahlman, S., "Faster-learning variations on back-propagation : An empirical study," Proceedings of the 1988 Connectionist Summer School, pp. 38-51, Morgan Kauffmann, San Mateo, 1988.

[19] Allen, R., "Adaptive training of connectionist state machines," Proceedings of ACM Computer Science Conference, p.

428, 1989.

[20] Cartwright T. & M. Brent, "Segmenting Speech Without a Lexicon:The Roles of Phonotactics and Speech Sources," Proceedings of the First Meeting of the ACL Special Interest Group in Computational Phonology, pp. 83-90, 1994.

[21] Hanes, M., S. Ahalt, & A. Krishnamurthy, "Acoustic-to-Phonetic Mapping Using Recurrent Neural Networks," IEEE Transactions on Neural Networks, Vol. 5, No. 4, pp. 659-662, 1994.

이 찬 도

1975년 서울대학교 사범대학 독일어과(문학사)
1984년 미국 Arizona State University 독일어과(M.A.)
1987년 미국 Indiana University 전산과학과(M.S.)
1991년 미국 Indiana University 전산과학과(Ph.D.)
1991년~92년 KAIST 인공지능 연구센타 선임연구원
1992년~현재 대전대학교 정보통신공학과 조교수
1994년~현재 정보처리학회지 편집위원
관심분야 : 인공지능, 음성언어 처리, 인공 신경망, 유전 알고리즘 등