1999

Structural knowledge in simple recurrent network?

Frank Shihong Hong
University of Massachusetts Amherst

STRUCTURAL KNOWLEDGE IN SIMPLE RECURRENT NETWORK?

A Thesis Presented

by

FRANK SHIHONG HONG

Submitted to the Graduate School of the
University of Massachusetts Amherst in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE

September 1999

Psychology

STRUCTURAL KNOWLEDGE IN SIMPLE RECURRENT NETWORK?

A Thesis Presented

by

FRANK SHIHONG HONG

Approved as to style and content by:

Gary F. Marcus, Chair

Alexander Pollatsek, Member

Margaret J. Speas, Member

Melinda Novak, Chair

Psychology

ACKNOWLEDGMENTS

There are many people that I must thank for their advice, support and encouragement. I thank Professor Margaret Speas for her kind acceptance of my invitation to be my committee member when she had many obligations in the Linguistics Department. I thank Professor Alexander Pollatsek for his wonderful teaching of statistics during my first year and extremely patient advising of the data analysis for this thesis.

My special thanks go to Professor Gary Marcus, my advisor, and Chairman of the committee. Without his encouragement and advice, connectionism and computer modeling would still remain a myth to me; without his suggestion, the idea of this thesis would be beyond my wildest imagination; without his influence, my interest in language acquisition would not evolve into a broad interest in Cognitive Science. Indeed, through each advice he offered, each reading he recommended and each speaker he invited, Professor Marcus has built an intellectual infrastructure for my study from which I has benefited so much.

Finally, I would like to thank Professor Neil Berthier, Mr. Liu Chang, a Ph.D. student of Electrical Engineering, Mr. Wang XiaoGuang, a Ph.D. student of Computer Vision and Mr. Suijith Svijayan, an Amherst College junior, who always

insists that he is a Cognitive Science major instead of a Psychology major, for their support in computer technology.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 General Background: Symbolism *vs.* Connectionism

Connectionism is challenging "the central dogma of Cognitive Science, that intelligence is the result of the manipulation of structured symbolic expressions." (Newell, 1980, Pinker & Mehler, 1988)(henceforth, this so-called central dogma is referred as "symbolism".) To advocates of connectionism, one of the most important appeals of connectionism, as indicated by its two other names "Parallel Distributed Processing" and "Neural Networks", is its biological plausibility, namely the massive parallel simple processing units and the connections among them are highly reminiscent of neurons and synapses in the brain. However, to followers of symbolism, connectionism's limits do not stem from its parallel mechanism but the fact that it does not process "symbols". "The issue is not whether the mind is a serial computer or a highly parallel one; it is whether the mind processes symbols, whether it has rules and representations." (Pylyshyn, 1984, p.73)

It is noteworthy that both symbolism and connectionism agree that representations are indispensable to the explanation of cognition. (Fodor & Pylyshyn, 1988, Gelder, 1990, Smolensky, 1987) The disagreement lies in the conception of the nature of representations and the mechanism processing representations. Symbolism argues that mental representations are discrete content-blind symbols and the mechanism of manipulating the symbols is governed by mental rules. Connectionism, partially inspired by neurons' activities which are analog in nature, (Anderson, 1983) proposes that mental representations are similarity-based and continuous in nature. Furthermore, to connectionism, mental rules are at most convenient descriptions of the knowledge a system has. (Rumelhart & McClelland, 1986) The knowledge of a cognitive system is represented in the connections among the widely distributed processing units. (McClelland, Feldman, Adelson, Bower, and McDermott,1986, cf. Fodor & Pylyshyn, 1988)

Human language, a paradigm case of a symbol-based mental rule system (Marcus et al., 1995, Rumelhart & McClelland, 1986) has become one of the most important arenas of connectionism and symbolism. A classical example

which supports the idea that human language is the product of a mental rule system comes from the fact that English-speaking children can add "-ed" to a novel verb, like "gutch", to form past tense in experimental settings. They can even produce "comed", "goed" etc. when they fail to retrieve the correct irregular forms from memory. As Berko (1958) said:" if a child knows that the plural of 'witch' is 'witches', he may simply have memorized the plural form. If however, he tells us that the plural of 'gutch' is 'gutches', we have evidence that he actually knows, albeit unconsciously, one of those rules which the descriptive linguist, too, would set forth in his grammar."

Rumelhart & McClelland (1986) used a two-layer network (henceforth RM model) to demonstrate that without explicit representation of the "-ed" rule, the network can generalize past tense forms of new verbs and the network's learning process is similar to children's past-tense learning. Particularly, the network, just like children, experiences an overregularization stage. Two mechanisms employed by the network are essential to its performance. First, all inputs (verb stems) and outputs (past tense forms) are represented as trigrams of phonetic features which define the similarity

space making similarity-based generalization possible. That is, the trained RM model can generate past-tense forms of new verbs on the basis of their feature overlaps with old ones. Second, to trigger the overregularization phenomenon, the experimenter has to feed the RM model with a sudden influx of regular verbs and their past tense forms during the intermediate stage of training. However, the data suggests that children's overregularization is not due to a sudden influx of regular verbs in their input. (Marcus et al., 1992, Pinker & Prince, 1988) The second mechanism is simply based on a false assumption of child language acquisition. The first mechanism is of particular interest in this study, since the notion of feature-decompositional representation is quite opposite to the symbol-based representation. "The hallmark of a symbol is that it can represent an entire class of individuals suppressing the distinctions among them." (Marcus et al., 1995) In the case of past tense, the "-ed" rule applies to all instances of the symbol "verb", except for those irregular ones listed in memory. The "-ed" rule does not pay attention to the phonetic features of the instances of the symbol "verb". Symbolic models commit to symbols which represent the

"object *qua* object" and variables which represent task relevant class of objects that "cut across feature similarity" (Pinker & Prince, 1988), while the RM model represents individual objects and variables as nothing but clusters of activated features. This fundamental feature, namely lacking symbols, is responsible for the following two aspects of the RM model's performance. First, the RM model does not generalize "-ed" properly to many new verbs; second, the RM model is too powerful to be a model of human learning. That is, it can extract some statistical correlations among features that are not found in any human language, such as "mirror-reversal of phonetic strings". (Pinker & Prince, 1988)

Furthermore, Rumelhart & McClelland (1986) mistake the notion of mental rule when they claim that the RM model's uniform mapping can replace the mental rule. In the case of English past tense, the regular form "-ed" is both statistically dominant and a product of a mental rule. However, a mental rule does not have to apply to the statistically dominant case. Marcus et al. (1995) found that German participle -t applies to a much smaller percentage of verbs than its English counterpart, and the German plural -s

applies to a small minority of nouns, though both -t and -s behave like their English counterparts as default mental rules. This German case strongly challenges the assumption behind the RM model, i.e., deriving lawful behavior by picking up the dominant statistical correlations among the features. To summarize, as far as the case of past tense is concerned, the non-symbolic RM model based on a few false assumptions of human language and language acquisition does not make a strong empirical case against the symbolism mental-rule account.

Past tense morphology is only a tiny piece of the whole rule system of language. The debate between connectionism and symbolism continues in other domains of language as more capable connectionism models are developed. One network called "simple recurrent network" developed by Jeffrey Elman (1990, 1991, 1993) is the focus of this research.

1.2 Introduction to Simple Recurrent Network

1.2.1 What Is Simple Recurrent Network

Many human behaviors (e.g. language, goal-directed behavior and planning) express themselves as temporal sequences. (Elman, 1990, Lashley, 1948, Servan-Schreiber, etal., 1988) Simple Recurrent Network (henceforth, SRN), among other connectionism models is designed to process sequential knowledge.

A prototypical SRN model consists of input units, output units, hidden units and context units. See Figure 1.1. Context units are also "hidden" units in the sense that they interact only with other nodes internal to the network. It is using context units that makes the model a "recurrent" network. Hidden units are activated by both input units and context units; on the other hand, hidden units also feed back to activate context units. In processing a sequential input, at time *t*, hidden units are activated by input at time *t* and their own activations at time *t-1* which are provided by the context units; in the mean time, hidden units feed their activations at time *t* back to context units

which will return such activations to hidden units at time *t+1*. The weights of connections between hidden units and context units are fixed at 1.0 and not subject to

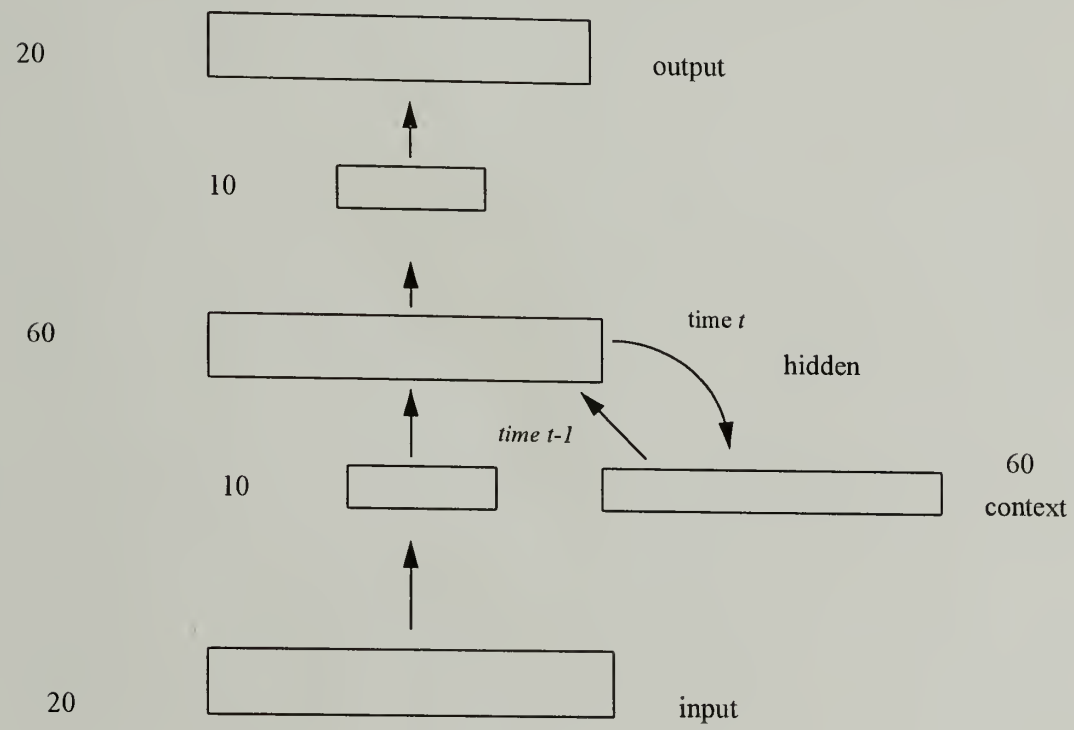

Figure 1.1 Architecture of Simple Recurrent Network

learning-based adjustment. In the learning phase, the output is compared with a teacher input and the error as a result

of that comparison is used by the back-propagation algorithm to adjust the connection weights incrementally.

1.2.2 The Performance of Elman's Simple Recurrent Network

In this research, the focus is Elman's SRN model (1990, 1991, 1993), which is the basis of his theoretical claims on language acquisition as a statistics-driven process. Basically, the learning capacity of Elman's SRN model (henceforth, ESRN) is demonstrated by its performance in a number of prediction tasks.

Elman (1990) shows that ESRN can predict one letter in the sequence "diibaguuubadiidiiguuu..." after being given the previous letter. That is, after training, given "d" as input, ESRN's output is "i" . Notice that the sequence is semi-random; consonants occurred randomly, but following a given consonant, the identity and number of following vowels are regular. (The subsequences are "dii", "ba" and "guuu".) Because of the randomness of consonants, errors in predicting consonants are high, while errors tend to be low in predicting vowels. After training with a more complicated corpus which consists of 200 sentences of varying length (4-

9 words, each word consists of letters), Elman (1990) also shows that ESRN can predict the next letter at each point in time. Similar to the "consonant-vowel" task, at the onset of each new word, the error is high; as more of the word is received, the number of errors declines. As Elman said, "the error provides a good clue as to what the recurring sequences in the input are, and these correlate highly with words." Furthermore, Elman argued that in the above mentioned model, the notion of "word" was learned rather than imposed in advance by the experimenter. Elman also claimed that the network discovered word categories such as Noun and Verb, and subcategories such as Transitive Verb and Intransitive Verb.

To extend his idea, Elman (1991) constructs a semi-artificial language to test ESRN. The sentences are formed from a lexicon of 23 items including 9 nouns, 12 verbs, the relative pronoun Who, and an end-of-sentence indicator (a period). The sentences have "agreement" (e.g. John feeds dogs./ *Boys sees Mary. Note: "*" signifies ungrammatical sentences.) and "verb argument structure". The agreement and verb argument structure become complicated in relative clauses. Furthermore, because recursion is permitted, the

agreement and verb-argument relation may be extended over a considerable distance, such as "Boys [who girls (who dogs chase) see] hear." After training the error of prediction is 0.177. (Note: initial error is 12.45; the error is not "mean squared error" since this is a non-deterministic prediction task. Instead, Elman uses network's actual outputs to compare against the likelihood of each target word in every sentence, given the sentence context up to that point.) Particularly, ESRN acquires the agreement, verb argument structure and other properties. For example, given "boy lives", ESRN predicts "end-of-sentence"; given "boy sees", ESRN predicts "Singular Noun" and "Plural Noun". (Since it has no way to tell exactly which one, singular noun or plural noun will follow.) Furthermore, given "boys who Mary chases", ESRN predicts "Plural Verb". That is, despite the intervening relative clause, ESRN knows that "boys" should agree with a plural verb. Supported by ESRN's performances, Elman made a number of claims on language acquisition in general.

1.3 Elman's Claims Based on ESRN

1.3.1 Prediction

Elman (1990) acknowledges that "While listeners are clearly able to make predictions based upon partial input, prediction is not the major goal of the language learner." Elman (1991) takes a more aggressive position "Although language processing obviously involves a great deal more than prediction, prediction does seem to play a role in processing. Listeners can indeed predict, and sequences of words which violate expectations i.e., which are unpredictable, result in distinctive electrical activity in the brain"; "if we accept that prediction or anticipation plays a role in language learning, then this provides a partial solution to what has been called Baker's Paradox... if we suppose that children make covert predictions about the speech they will hear from others, then failed predictions constitute an INDIRECT source of NEGATIVE EVIDENCE which could be used to refine and retract the scope of generalization." (emphasis added by this thesis) In addition to this theoretical reason for using prediction

tasks, one technical reason might be that ESRN is good at predicting sequence in general, (i.e. not only predicting speech) and in ESRN the role of external teacher is minimized, since the target outputs are supplied by the environment at the next moment in time.

1.3.2 Representation

Representation is essential to ESRN for both theoretical and technical reasons. In the semi-artificial language task, each word is an orthogonal vector of all 0's and a single 1. (e.g. "cat"--> 0001; "cats"--> 0010; "dog"--> 0100; "chase"--> 1000) For *n* words, each vector must be *n* bits long with one of them flipped on. To symbolism, this representation is problematic in that no morphology is represented, let alone word category information. That is, the distance between "cat" and "cats" is same as the distance between "cat" and "dog". (Marcus, 1993) However, to Elman there are sufficient theoretical justifications for using this representation paradigm. He believes that it is dangerous to presuppose linguistic representations such as "noun", "agent", even "word".

(Recall how ESRN in Elman (1990) discovers the notion of "word" from the input stream.) ESRN is preferred by Elman not only because it is parsimonious in the sense that *a priori* assumptions are limited to variables which are "directly observable" in the environment, but also because the network itself can show us what linguistic representations are needed and how they are acquired by the network. Elman is proud of the "fact" that given the orthogonal representation telling nothing about word category membership information, ESRN discovers word categories (e.g. noun, verb) and subcategories (e.g. animate/inanimate, human/inhuman, etc.) by itself.

From the technical point of view, Elman prefers distributed representation (at the hidden layer) to localist representation because the former but not the latter can provide large enough space to form "abstract representation" and "structural knowledge" at the hidden layer which "tend to be distributed across the high-dimensional (and continuous) space." (Elman, 1991, p.197) It is important to note that Elman commits to the claim that ESRN acquires "structural knowledge" through training; "structural knowledge" plays the causal role in ESRN, and such

"structural knowledge" is formed as "internal representation" at the hidden layer.

1.3.3 Statistics, Structural Knowledge, Type/Token Distinction

Statistical regularity is essential to any connectionism model including ESRN. Elman is fully aware of the classical argument made by Miller and Chomsky (1963) that certain properties of natural language make statistically based learning infeasible. To be immune to such argument against statistical learning, Elman (1993) draws a fine line between the use of statistics as the "DRIVING FORCE" and statistics as the "OUTCOME" of learning. (emphasis added by this thesis) Elman (1993) argues that ESRN uses statistics as driving force to reach an outcome of learning which can be characterized as a rule system rather than a look-up table of statistics. Furthermore, Elman (1993) argues that ESRN is able to "extrapolate beyond their training data in ways which obviate the need, for example, to see all possible combinations of words in sentences." To summarize, it is argued by Elman that by taking advantage of

only co-occurrence statistics (i.e. without innate structural knowledge or other helpful information) in the training data, ESRN can successfully acquire the structural knowledge which is supposed to be a hallmark of human language.

How can ESRN acquire structural knowledge? Elman's explanation is as follows. For convenience, let's use "context information" instead of "co-occurrence statistics". Recall that "context units" are essential to Simple Recurrent Network. Hidden units are always activated by both input units and context units. That is, there are no representations of "words in isolation". Each word is represented along with its context. It is using context information that makes ESRN acquire the structural knowledge, as Elman believes. To illustrate, let's use geometric terms. As mentioned above, the hidden units activation pattern is represented in a high-dimensional space. In such space, each word+context (again no word in isolation) occupies its own specific position. Call each of such word+context a "token". "Similar tokens" ("similar" is defined by co-occurrence statistics.) are near to each other in such space. A "type" may emerge out of such neighborhood

relationship among tokens. For example, "John" as a type emerges out of many "John+context". Recursively, a bigger type, say "noun" may emerge out of neighborhood relationship among many "John", "Mary", "window" etc. A type is represented as the mean vector of tokens. That is, such space can be viewed as a tree with branches which in turn have smaller branches. Elman argues that ESRN has not only context-insensitive types (something like symbols in symbolism terms) but also context-sensitive tokens. Elman (1991) also addresses the issue of the difference between connectionism and symbolism in general. He argues that connectionism "begins the task of abstraction at the other end (i.e. token) of the continuum." (Elman, 1991, p. 221), while symbolism comes from the type end; "...it is not obvious what is meant by a rule. In the most general sense, a rule is a mapping which takes an input and yields an output."

To demonstrate the importance of context information to the acquisition of structural knowledge, Elman (1990) replaces the word *man* in the training data set with a new word *zog*. (That is, the vector of *zog* is different from those vectors the model is trained with.) A new set of 10000

sentences is created with such replacement wherever *man* occurs. The new training set is presented to a trained network (No new learning is allowed to occur.) Inspection shows that "The internal representation for the word *zog* bears the same relationship to the other words as does *man* in the original training set." (Elman, 1990, p. 201) That is, *zog* is assigned nounhood among other structural properties by the trained network. It strongly suggests that ESRN really has structural knowledge, otherwise how can *zog*, a new word "inherit" (i.e. without learning) such structural properties from the network? For convenience, call this test "zog test".

1.4 Human Subjects' Learning of Finite State Grammars

ESRN's learning of structural sequences is reminiscent of human subjects' learning of artificial grammars. In the late 1950s and early 1960s, inspired by Chomsky's decisive critique of the behavioristic model of language acquisition, some psychologists were interested in so called "implicit learning" referring to the notion that children acquire the grammar of their language in an implicit fashion (Chomsky,

1957, 1959) rather than establishing S-R associations by explicit imitating. Using sentences (i.e. sequences of letters) generated by a finite state grammar as stimuli, Reber (1967) found that despite being uninformed of the structural nature of the stimuli and in the setting of a memorization task, subjects "learned to become increasively sensitive to the grammatical nature of the stimuli." Furthermore, based on what they learn from the mere exposure of the stimuli, subjects can recognize grammatical sequences which they had not seen during the learning session. This entails an intriguing question: what is the nature of the knowledge that subjects have as a result of such learning? Reber (1989) addressed this question by using a knowledge transfer paradigm which is used in this research.

In Reber (1989), two finite state grammars (denoted X and Y) were created and matched with each of two sets of lexicons/letters (denoted 1 and 2), creating four artificial languages, L-X1, L-X2, L-Y1, L-Y2. See Figure 1.2. Note that L-X1 shared exactly the same Syntactic Structure with L-X2, so did L-Y1 and L-Y2. L-X1 and L-X2 used a different explicit lexicon to construct sentences of each language, so

did L-Y1 and L-Y2. In the first task, four groups of subjects were asked to memorize each of four sets of

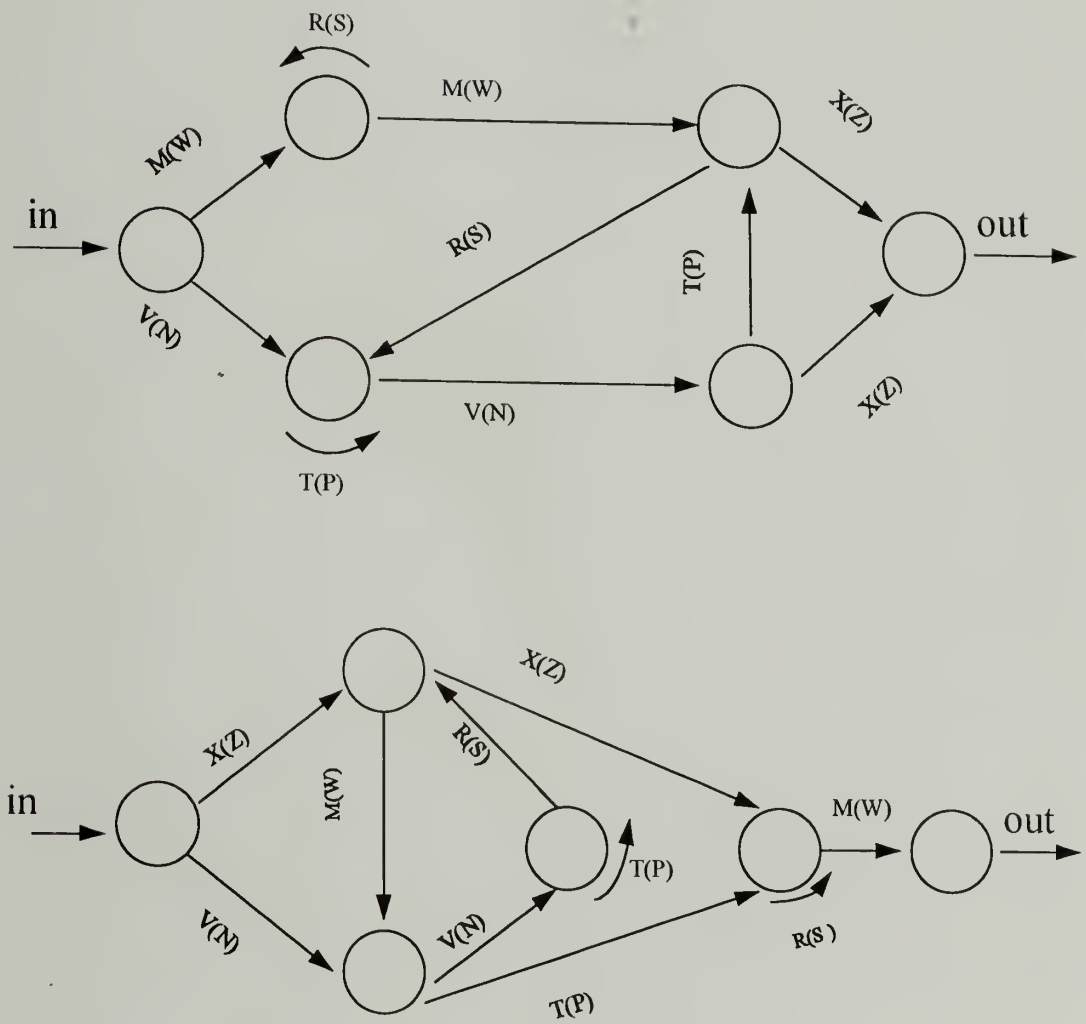

Figure 1.2 Two Finite State Grammars and Two Sets of Letters

sentences. In the second task, experimental groups were defined by the nature of four transfer types: Lexicon-

Change, Syntax-Change, Both-Change and Null-Change. In the second task, subjects were again asked to memorize the training data. The prediction was that if subjects acquired the abstract structural (syntactic) knowledge, then changing lexicon would bring less detrimental effect on subjects' performance than changing syntax would. However, if subjects were learning to string together explicit lexicon, Lexicon-Change condition would entail more detrimental effect on their performance. Reber claimed that no *a priori* prediction could be made for the Both-Change condition. Null-Change was used as the control group. Testing results, as Reber (1989) reported, suggested that "whenever the syntactic structure of a language was changed on the transfer task, it had a detrimental effect on subjects' behaviors, while a change in the explicit lexicon had no noticeable effect." The implication of this result was that what subjects acquired was abstract structural knowledge and this knowledge could be carried over to an appropriate new situation, such as the situation in which the lexicon of the sentences but not the syntax was changed. This carry-over effect is dubbed as "Structural Knowledge Transfer Effect" in this thesis.

To summarize, Reber (1967, 1989) showed that human subjects can learn the syntax of a finite state grammar from the mere exposure of the sentences generated by that grammar; the syntax is represented as content-blind symbol-based rules which subjects can mentally consult in order to do recognition and reproduction tasks. In the finite state grammar paradigm, the very content-blindness of the symbolic representation is reflected in the "Structural Knowledge Transfer Effect".

1.5 Statement of The Problem

As mentioned above, Elman (1990, 1991, 1993) commits to the following claims.

First, starting from non-structural representations and using co-occurrence statistics as "driving force" of learning, ESRN can acquire structural knowledge, such as the syntax of a finite state grammar.

Second, the structural knowledge plays a causal role underlying ESRN's prediction performance.

Third, as the *Zog* test indicated, though the word *zog* was not in the training data set, a trained ESRN can assign structural properties to this new word. That is, ESRN can go beyond the mere statistical regularity of its training data.

Fourth, like all connectionism models, the knowledge of ESRN is represented in the connections among the processing units. It is noteworthy that the computer program of ESRN has a special file storing the weights of the connections. The program of ESRN allows loading this file of weights (i.e. file of knowledge) to other networks.

The issue at stake is: can ESRN really go beyond the co-occurrence statistics and acquire genuine structural knowledge? The prediction is that a sufficiently trained ESRN will have significant worse performance in the transfer task, despite the fact that the syntax, but not the lexicon on the transfer task was identical to that on the pre-transfer task. Failing to transfer the knowledge ESRN gained during pre-transfer task to transfer task in the Lexicon-Change condition will undermine the claim that ESRN can acquire genuine structural knowledge.

To test this prediction, a human subject experiment and a computer simulation were conducted. Both experiments used the knowledge transfer paradigm developed in Reber (1989). The human subject experiment was a modified version of Reber (1989) to confirm the existence of the Structural Knowledge Transfer Effect in human subjects; the computer simulation put ESRN in the knowledge-transfer paradigm.

CHAPTER 2

METHOD

2.1 Human Subject Experiment

2.1.1 Subjects

A group of 71 students (most of them were undergraduate students taking Psychology courses in the University of Massachusetts at Amherst.) participated this experiment. Some subjects gave up during the experiment; some subjects didn't finish the experiment due to technical problems, such as failures of local area network. None of the incomplete data was used. Eventually 48 subjects' data were used for the analysis.

2.1.2 Design

The experiment consisted of two tasks. There was a five-minute break between two tasks. In both tasks, subjects were instructed to do a memorization task and no mention was

made of the grammar or the rule of the sentence generation. The stimuli was referred as "letter sequences" rather than "sentences". Task 1 was the pre-transfer stage in which subjects did the original learning of the stimuli generated by a finite state grammar. Task 2 was intended and designed to be the knowledge transfer stage.

On task 1, all subjects were treated identically. They were asked to try their best to memorize the letter sequences presented on the computer. Subjects were asked to do the same job on task 2. However, on task 2, the subjects were divided into four experimental groups which were Lexicon-Change Group, Syntax-Change Group, Null-Change Group and Both-Change Group.

2.1.3 Stimuli

Four languages generated by two sets of letters and two Finite State Grammars were used as stimuli. The first letter set includes "V, S, P, T and X"; the second letter set includes "F, J, U, M and N". The two grammars (referred as "New Grammar" and "Old Grammar" for convenience) were copied from Reber's 1989 work. See Figure 1.2. That is, four

languages were referred as "New-1", "New-2", "Old-1" and "Old-2" respectively. For each language, 43 unique sentences of length 3-8 letters consisted of the stimulus pool. Each subject was randomly assigned to one of the four experimental groups as above mentioned. For each subject on each task, 18 sentences were selected at random from the stimulus pool of a specific language. Since there were four different languages, there were 16 possible "transfer paradigms" which were further divided into the four experimental groups in terms of the type of change: Lexicon-Change, Syntax-Change, Null-Change and Both-Change. See Table 2.1.

2.1.4 Procedure

All subjects filled the consent form (see Appendices A) before they went to the lab. Each subject had a three-set practice before real experiment. After the practice, the subjects were left alone in the lab doing the tasks. The computer gave the instruction before presenting stimuli. See Appendices B for the instruction. For each subject on each

Table 2.1 Illustration of Subject Division

Task 1	Task 2	Experimental Group
New-1	New-1	Null-Change
New-2	New-2	Null-Change
Old-1	Old-1	Null-Change
Old-2	Old-2	Null-Change
New-1	New-2	Lexicon-Change
New-2	New-1	Lexicon-Change
Old-1	Old-2	Lexicon-Change
Old-2	Old-1	Lexicon-Change
New-1	Old-1	Syntax-change
New-2	Old-2	Syntax-change
Old-1	New-1	Syntax-change
Old-2	New-2	Syntax-Change
New-1	Old-2	Both-Change
New-2	Old-1	Both-Change
Old-1	New-2	Both-Change
Old-2	New-1	Both-Change

task, 18 sentences selected in the way as mentioned above were divided into 9 sets of 2 sentences in each set. The stimuli were presented by a Macintosh computer letter by letter rather than sentence by sentence. That is, subjects did not see a whole sentence, like MNF on the screen. What they saw was that "M" stayed on the screen for 2 seconds then disappeared; "N" showed up after the disappearance of "M" and stayed for 2 seconds then disappeared, so and so forth. At the beginning of each sentence, a "*" appeared as a cue and after the last letter of each sentence, a "$" appeared as the signal of ending a sentence. There were 3 seconds interval between two sentences. After each set of two sentences was shown, the subject was asked to reproduce the two sentences they just saw by typing their recall into the computer. The computer informed the subject which sentences were reproduced correctly and which were not. No information was given about the nature of errors. The subject's original performance were recorded into a separate file. More importantly the computer recorded the number of incorrect sentences reproduced by the subject for each set. It was this "number of errors" for each set that was used as the measurement of subjects' learning performance and was

subject to statistical analysis. A set of sentences would keep appearing on screen until the subject reproduced the full set correctly. This procedure continued until all 9 sets were learned. This procedure was used in both tasks.

2.2 Computer Simulation with ESRN

2.2.1 Subjects

A ESRN-template was created. It had 10 input units, 10 output units, 40 hidden units, 40 context units. Based on this template, 10 nets were created as the equivalent of 10 human subjects in order to have a more reliable data based on the average of 10 nets' performance. Since the localist representation was used in the input and output level and each letter can be represented by any one of the 10 units, 10 nets were only different from each other in terms of the 10-unit representation code of the five letters. See Table 2.2 for the illustration.

Table 2.2 Ten Representation Codes for Ten Nets

Net 1											Net 2	Net 3	Net 4	Net 5
T:	1	0	0	0	0	0	0	0	0	0	P	X	V	S
S:	0	1	0	0	0	0	0	0	0	0	T	P	X	V
V:	0	0	1	0	0	0	0	0	0	0	S	T	P	X
X:	0	0	0	1	0	0	0	0	0	0	V	S	T	P
P:	0	0	0	0	1	0	0	0	0	0	X	V	S	T
Net 6											Net 7	Net 8	Net 9	Net 10
T:	0	0	0	0	0	1	0	0	0	0	P	X	V	S
S:	0	0	0	0	0	0	1	0	0	0	T	P	X	V
V:	0	0	0	0	0	0	0	1	0	0	S	T	P	X
X:	0	0	0	0	0	0	0	0	1	0	V	S	T	P
P:	0	0	0	0	0	0	0	0	0	1	X	V	S	T

2.2.2 Design

The simulation also consisted of two tasks. The first task was the pre-transfer stage and the second task was intended and designed to be the knowledge transfer stage. In the first task, each ESRN net was trained with sentences

generated by a finite state grammar. (In fact, the stimuli used for the simulation were exactly the stimuli used by one of the human subjects. See Appendices C.) In the second task, each trained net was exposed to a new set of training data. The new training data in task 2 was generated by the same finite state grammar as that of task 1. The only difference between training data of task 1 and task 2 was a vector-representational one. That is, for example, 1000000000 in task 1 was changed into 0000010000 in task 2; 0100000000 in task 1 was changed into 0000001000 in task 2. Changing vector-representational corresponds to changing letters from P, V, T, S and X to M, N, F, J and U or *vice versa* in human subject case.

2.2.3 Stimuli

The training data was a copy of the stimuli used by one of the human subjects on task 1. Each letter of the sentences was translated into a 10-bit vector with one bit flipped on. There were 10 vector-representation codes as listed in Table 2.2. The testing sentence "TPPTXVPS" was not in the training data.

2.2.4 Procedure

On task 1, each net was trained up to 100,000 sweeps. Each sweep was an exposure of a letter. During the training, a built-in mechanism saved the network's weights status into separate files every 4,000 sweeps. After the training, each net was tested by asking it to predict each letter of the testing sentence. A program was used to check out how many correct predictions were made by each net. Two indices were used to measure the nets' prediction performance. The primary index was the number of correct predictions made by the nets; the secondary index was the average Luce ratio. The Luce ratio was used as a measurement when the net made the same amount of correct predictions with different amount of training, which actually was a commonplace in this simulation. The Luce ratio is "the ratio of the highest activation on the output layer to the sum of all activations on that layer." (Servan-Schreiber, et al., 1988) The Luce ratio suggested how confident the net's predictions were. In fact, the Luce ratio is a more sensitive function of the training experience in terms of sweeps. Different nets reached their best performance with various amount of

training in terms of sweeps. In task 2, the trained net loaded with the weights file which resulted in its best performance in task 1 was trained another 100,000 sweeps with a new set of training data. During the training of task 2, the built-in mechanism also saved weights status into separate files every 4,000 sweeps. Because of the separate weights files, the testing was performed in the following way. For example, to find out ESRN's prediction performance with 24,000-sweep training experience, the weights file of 24,000-sweep training experience was loaded into the to-be-tested net, then the net was tested with the testing sequence.

CHAPTER 3

RESULTS

3.1 Human Subject Experiment

3.1.1 Dependent Measures

The "number of errors" in reproduction for each set of stimuli was used as the dependent measure in human subject experiment. See Table 3.1 for the illustration of how the errors were counted.

Table 3.1 Illustration of How to Count Subjects' Errors

Set	Presented	Recalled	# of Errors
1	TTV; TTP	PTV; TTP	1
1	TTV; TTP	TVV; TPPP	2
1	TTV; TTP	TTV; TTP	0
2	VVPP; PTV	VVPP; PTV	0

For the hypothetical case in the Table 3.1, 3 errors were counted for the first set and 0 error was counted for the second set respectively. Figure 3.1 shows the mean number of errors per set for all experimental groups across the 18 sets of the experiment. Table 3.2 shows the mean number of errors for each group on each of two tasks.

Table 3.2 Breakdown Means of 9-Set Data

Group	Task 1	Task 2	Difference
Lexicon-Change	2.48	1.69	0.79
Syntax-Change	3.51	3.12	0.39
Null-Change	3.59	2.07	1.52
Both-Change	2.92	2.32	0.60

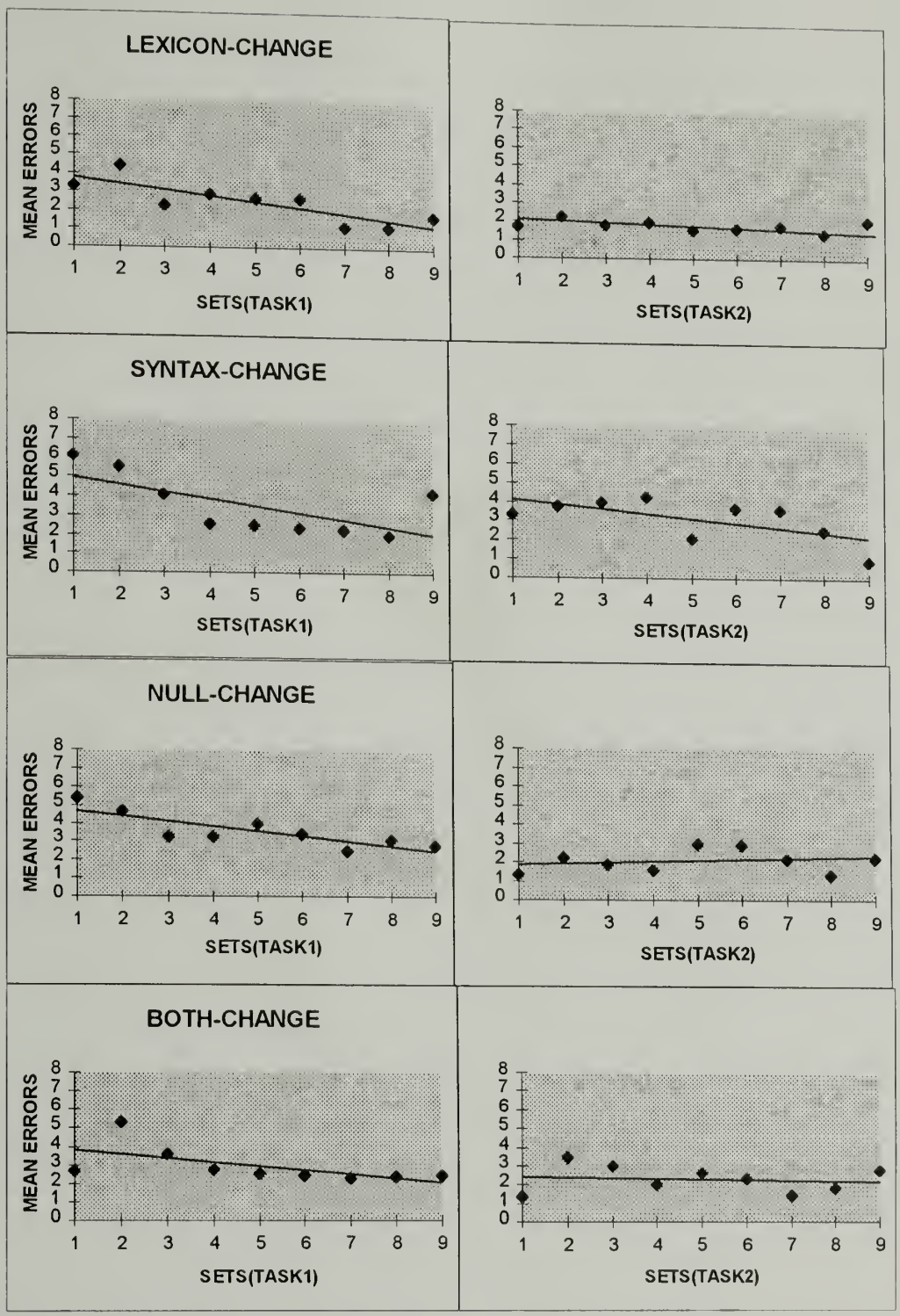

Figure 3.1 Mean Errors on Each of The 18 Sets

3.1.2 Analysis

Three blocks of analysis were conducted. Each block of analysis included ANOVA of data of both tasks, ANOVA of task 1, ANOVA of task 2 and comparisons of experimental groups on task 2.

The first block of analysis used each subject's 9-set performance on both tasks. A three-way ANOVA examining Task(1/2), Group(Lexicon-Change/Syntax-Change/Null-Change/Both-Change) and Set (1 through 9) with Task and Set as within subject factors was conducted.

Three significant effects were found: Task, $F(1, 44)=17.955$, $p<.000$; Set, $F(8, 352)=5.574$, $p<.000$; Task * Set, $F(8, 352)=3.498$, $p<.001$. It is noteworthy that the Task * Group interaction effect was not significant $F(3, 44)=2.413$, $p<.079$. A significant Task * Group interaction effect as reported in Reber (1969) is crucial in confirming the prediction of this study. This critical unexpected effect will be addressed later in this Chapter. A separate two-way ANOVA of data from task 1 showed that the only significant effect was Set, $F(8, 352)=7.043$, $p<.000$. The lack of Group effect was as expected, since all groups were

supposed to be treated identically on Task 1. The significant Set effect suggested a learning trend in this stage which was also as expected. A separate two-way ANOVA of data from task 2 showed that the only significant effect was Group effect, $F(3, 44)=2.815$, $p<.05$. This Group effect, as expected, suggested that four groups of subjects, though performed identically on task 1, performed differently on task 2 because of different experimental conditions involved.

To find out the way in which the four groups were different from each other, more comparisons between groups were conducted. A two-way ANOVA of the data from Lexicon-Change and Null-Change group on task 2 found neither significant Group effect, $F<1$, nor other significant effects. A two-way ANOVA of the data from Syntax-Change and Both-Change Group on task 2 did not find any significant effects either. Based on this result, the data from Syntax-Change and Both-Change group were collapsed together, so were the data from Lexicon-Change and Null-Change Group. A two-way ANOVA of this two-group collapsed data found a significant Group effect, $F(1, 46)=4.905$, $p<.032$ and that was the only significant effect. These results were also

reported by Reber (1969). Based on those results, Reber (1969) claimed that "whenever the syntax was changed on task 2 (i.e. in the transfer stage), it had a 'detrimental effect' on subjects' performance, while a change in the lexicon of the language had no noticeable effect." However, results of this study did not support such a clear-cut conclusion. Two more comparisons were conducted. A two-way ANOVA of the data from Lexicon-Change and Syntax-Change group revealed a significant Group effect, $F(1, 22)=6.722$, $p<.017$, just as expected.

However there were three non-significant comparisons which were not as expected. First, a two-way ANOVA of the data from Lexicon-Change and Both-Change group found no significant Group effect, $F(1, 22)=1.44$, $p<0.243$; second, no significant difference between Syntax-Change and Null-Change group, $F(1, 22)=3.693$, $p<0.068$; third, there was no significant difference between Null-Change and Both-Change group.

Finally, as expected, there was no significant difference between the group collapsing Lexicon-Change and Both-Change together and the group collapsing Syntax-Change and Null-Change together. Therefore, in total only two

significant comparisons were found. One was between Lexicon-Change and Syntax-Change group; the other was between the group collapsing Lexicon-Change and Null-Change together and the group collapsing Syntax-Change and Both-Change together. More comments on this result will be given in Chapter 4.

The first block of analysis yielded one crucial result, i.e., the lack of Task * Group effect which made the claim of "Structural Knowledge Transfer Effect" dubious. The second series of analysis, a more powerful analysis was carried out based on the assumption that if there was a knowledge transfer whatsoever, it should have occurred between the end of task 1 and the beginning of task 2. As Reber (1969, P. 118) commented "... discussions of negative and positive transfer refer to performance on the initial sets of task 2 relative to the asymptote achieved during task 1."

The second block of analysis used only the last 4-set data of task 1 and the first 4-set data from task 2. See Table 3.3 for the breakdown means. A three-way ANOVA of the data from both task 1 and task 2 revealed only one significant effect: Task * Group, $F(3, 44)=3.809$, $p<.016$. The non-significant Set effect was in keeping with the

assumption that subjects' performance achieved asymptote during the end of task 1. A separate two-way ANOVA using 4-set data from task 1 found no significant effect, exactly as

Table 3.3 Breakdown Means of 4-Set Data

Group	Task 1	Task 2	Difference
Lexicon-Change	1.688	1.896	-0.21
Syntax-Change	2.5	3.94	-1.44
Null-Change	2.96	1.75	1.21
Both-Change	2.35	2.44	-0.09

expected. A separate two-way ANOVA using 4-set data from task 2 found a significant Group effect, $F(3, 44)=4.468$, $p<.008$. As in the first block of analysis, more comparisons among the experimental groups were conducted. No significant difference between Lexicon-Change and Null-Change and between Syntax-Change and Both-Change Group was found. The difference between the group collapsing Lexicon-Change and Null-Change and the group collapsing Syntax-Change and Both-

Change was again significant, $F(1, 46)=7.804$, $p<.008$. The difference between Lexicon-Change and Syntax-Change group was significant, $F(1, 22)=9.314$, $p<.006$. In contrary to the expectation, no significant difference between Lexicon-Change and Both-Change or between Null-Change and Both-Change was found. However, contrary to the result in the first block of analysis, the difference between Syntax-Change and Null-Change group was significant, $F(1, 22)=9.304$, $p<.006$.

Obviously, there were some agreements and disagreements between the results of the analysis based on 9 sets and the analysis based on 4 sets. Before giving a complete list of those agreements and disagreements, the third block of analysis is reported below.

Both 9-set and 4-set analysis had the same result that the difference between the group collapsing Lexicon-Change and Null-Change and the group collapsing Syntax-Change and Both-Change Group was significant. The third series of analysis used the collapsed data and it was still concerned about the Task * Group Effect. A three-way ANOVA using 9-set data found: Task, $F(1, 46)=17.636$, $p<.000$, Task * Group, $F(1, 46)=4.33$, $p<.043$, Set, $F(8, 368)=5.558$, $p<.000$, Task *

Set, $F(8, 368)=3.46$, $p<.001$. As expected, a separate two-way ANOVA of task 1 found no significant Group effect but a significant Set effect, $F(8, 368)=7.107$, $P<.000$. A separate two-way ANOVA of task 2 found a significant Group effect, as reported in the first and second block of analysis, $F(1, 46)=4.905$, $p<.032$. A three-way ANOVA using 4-set (collapsed) data found the only significant effect was Task * Group, $F(1,46)=4.735$, $p<.035$. As expected a separate two-way ANOVA of task 1 found neither a significant Group effect, nor other significant effects. A separate two-way ANOVA of task 2 found a significant Group effect, as reported in the second block of analysis, $F(1,46)=7.804$, $p<.008$.

It seems that the results based on the data of two collapsed groups and 4 sets for each task are most consistent with the predictions of this study and Reber's (1969) reports. That is, subjects can transfer the abstract syntactic knowledge in the appropriate new situation. However, this claim was cast doubts by a few critical unexpected effects which will be addressed in next Chapter.

As promised earlier, Table 3.4 and 3.5 list the agreements and disagreements between 9-set analysis and 4 set analysis; between four-group analysis and two-group

Table 3.4 Comparing Results of ANOVAs

	Four Groups			Two Groups		
	9 Sets		4 Sets	9 Sets		4 Sets
Effects	R	H	H	R	H	H
Task	**	**	not	N/A	**	not
Set	**	**	not	N/A	**	not
Task * Group	*	not	*	N/A	*	*
Task * Set	not	**	not	N/A	**	not
Overall Group	not	not	not	N/A	not	not
Task-1 Group	not	not	not	not	not	not
Task-2 Group	*	*	**	*	*	**
Task-1 Set	**	**	not	N/A	**	not
Task-2 Set	not	not	not	not	*	not

Note: "R" stands for Reber (1969); "H" stands for this thesis; "*" stands for "p<.05"; "**" stands for "p<.01". "not" stands for non-significant.

analysis, and between data in this study and Reber's data in 1969.

Table 3.5 Comparisons Between Groups on Task 2

Comparisons	9 Sets		4 Sets
	R	H	H
L *vs* N	not	not	not
S *vs* B	not	not	not
L&N *vs* S&B	*	*	**
L *vs* S	*	*	**
L *vs* B	*	not	no
S *vs* N	*	not	**
L&B *vs* S&N	N/A	not	N/A
N *vs* B	*	not	not

Note: "L" stands for Lexicon-change; "S" stands for Syntax-Change; "N" stands for Null-Change; "B" stands for Both-Change.

3.2 Computer Simulation with ESRN

3.2.1 Dependent Measures

The ESRN's prediction performance is illustrated as follows.

correct:	1	0	0	0	0	0	0	0	0	0
ESRN:	0.1	0.2	0.6	0.3	**0.7**	0.2	0.1	0.1	0.1	0.1

A program idealized the ESRN's prediction as the following.

ESRN:	0	0	0	0	1	0	0	0	0	0

That is, the program took the maximum activation as 1 and other activations as 0. In this example case, the network's prediction was wrong. The correct prediction was referred as "hit". It is noteworthy that this treatment is in favor of the ESRN. Based on such idealization, the program yielded the "number of hits" for each testing. The number of hits ranged from 0 to 8, since the testing sentence had 8 letters. For each task, averaging the "number of hits" for each testing over 10 nets yielded "mean hits". See Figure 3.2 for the data. The nets made more correct predictions on task 1 than on task 2. The mean difference was 0.554. A paired t-test revealed that the difference was significant,

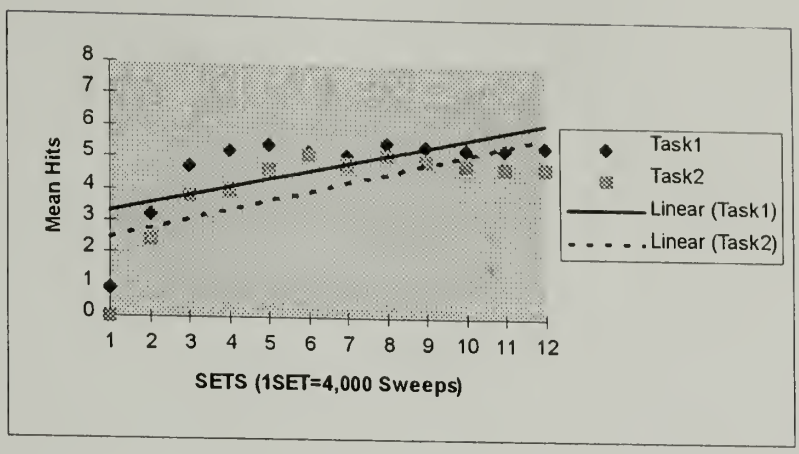

Figure 3.2 Mean Hits of the First 12 Sets for Both Tasks

$t(25)=9.222$, $p<.000$. This result was clearly in favor of the prediction made in Chapter 1. The nets' performance on task 2 showed no positive transfer of knowledge whatsoever despite they were loaded with the weights files as results of learning on task 1. The Elman's claim that ESRN can acquire structural knowledge as humans do was seriously undermined by this result.

CHAPTER 4

DISCUSSION

4.1 Comparing Human Subject Experiment with Reber's 1969 Study

First of all, it is noteworthy that the human subject experiment was a modified version of Reber (1969) in which subjects saw stimuli sentence by sentence rather than letter by letter as what I did in this study. The reason for this modification was that this study was trying to have a fair comparison between human subjects and ESRN, since a reasonable intuition is that being exposed to the whole sentence might facilitate human subjects to acquire structural knowledge while ESRN can only see the stimuli letter by letter. Furthermore, being fed only one unit of a sequence at a specific time was one of the important features of ESRN, so it was inappropriate to change ESRN's capacity in this regard. One more measure for having a fair comparison between human subjects and ESRN might be asking subjects to predict a new sentence letter by letter after

learning, just like what ESRN did in testing. If this prediction rather than reproduction paradigm had been used, subjects had to be instructed explicitly to learn a grammar rather than to be instructed to do a memorization task. That is, there would be no "implicit learning" any more. However, previous researches done by Reber (1967, 1989) showed that subjects failed to learn the grammar if they were instructed explicitly to do so. For this concern, the reproduction paradigm was used. A few more differences between human subject experiment in this study and Reber's are as follows. First, Reber (1969) presented the stimuli with 5 * 8 inch index card; while this study used computer presentation. Second, Reber (1969) used three sentences in each set with 6 sets on each task, while in this study, in order to compensate the difficulty posed by the letter-by-letter presentation paradigm, each set included only two sentences with 9 sets on each task. Third, Reber (1969) used 64 subjects with 16 in each group, while this study used 48 subjects with 12 in each group.

Secondly, it is remarkable that despite so many differences between Reber's 1969 study and this 1996 study in terms of implementation the experiment, there were still

overwhelming agreements in their findings. From Table 3.2 and Table 3.3, we can see that eleven out of sixteen reported significant and non-significant effects of any theoretical interests in Reber's study were agreed by 9-set analysis in this study; another eleven out sixteen reported significant and non-significant effects of interests in Reber (1969) were also agreed by 4-set analysis in this study; two inconsistent effects were as expected, since in some aspects there should be differences between using 4 sets and 9 sets. For example, there was a significant Set effect on task 1 in Reber's report; however we should not expect such Set effect in the last 4 sets, which was exactly what was found in the 4-set analysis. Once again, this result suggested that 4-set analysis had a more consistent account of the data. Unfortunately, there was no opportunity to see how many effects in this study were agreed by Reber's study, since this study did much more tests than Reber reported.

4.2 How to Interpret Structural Knowledge Transfer Effect

Presumably, the Structural Knowledge Transfer Effect that this study was looking for is that the knowledge subjects learned from task 1 is abstract in nature, so that on task 2 they can use that abstract knowledge to improve their performance (in terms of making fewer errors), if the Syntax is not changed from that on task 1. This prediction implies that on task 2 there should be a significant difference between Lexicon-Change and Syntax-Change; between Syntax-change and Null-Change; between the group collapsing Lexicon-change and Null-Change and the group collapsing Syntax-change and Both-Change. All these three crucial predictions were confirmed by the 4-set analysis. However, two other intuitive implications by the Structural Knowledge Transfer Effect, a significant difference between Null-Change and Both-change; between Lexicon-Change and Both-Change were not confirmed in either 9-set or 4-set analysis.

These two out-of-expectation non-significant differences may have three possible interpretations. First, the "structural knowledge transfer effect" claimed by Reber(1969) is simply not fully confirmed by this study.

Second, because both out-of-expectation findings involve Both-Change group, while on the other hand many other findings agree with Reber(1969), it is likely that Both-Change group had produced extreme values or other experimental errors. Third, the effect of changing both lexicon and syntax may be more complicated than a mere linear sum of the effect of changing lexicon and the effect of changing syntax. Reber(1969) had a similar out-of-expectation finding in this regard. He found that "There is a (non-significant) tendency for Both-Change group to make fewer errors than Syntax-Change group hinting that changing the syntax while using the same letters produces more interference than changing both." In fact, Reber (1969) made it clear that "It is difficult to establish *a priori* whether this (referring to Both-Change) should be a negative, neutral, or a positive transfer paradigm." In any event, it should be acknowledged here that the data of this study did not fully support the claim of "structural Knowledge Transfer Effect".

Another perspective of interpreting Structural Knowledge Transfer Effect is concerned when the transfer occurs. Presumably subjects acquired the syntax at the later

stage of task 1 instead of the beginning stage. If there was a knowledge transfer whatsoever (structural or non-structural; positive or negative) the most obvious transfer effect should be seen at the beginning stage of task 2, though it may last to the end of task 2. This idea was confirmed by the fact that a significant Task * Group effect was not found in 9-set analysis but in 4-set analysis.

4.3 ESRN's Performance

As mentioned in Chapter 3, the nets' prediction performance was sampled every 4000 sweeps from total 100,000 sweeps on each task. There were nine possible values of the number of hits (0--8). The nets reached their best performance with various amount of training. See Table 4.1 below.

It was crucial to find out the very best performance and load the weights file underling that performance to begin task 2. It was often the case that the number of hits remained unchanged after a lot of training was done while the Luce ratio changed along with the increase of the training experience. Therefore, the Luce ratio was used to

make a fine differentiation among the nets' performance levels.

Table 4.1 Training Amount for the Best Performance

	Task 1		Task 2	
Net	Highest	# of Sweeps	Highest	# of Sweeps
1	7 (3.32)	68,000	5 (2.21)	36,000
2	6 (2.93)	100,000	8 (3.71)	64,000
3	6 (2.78)	72,000	8 (2.94)	20,000
4	6 (3.04)	92,000	1 (0.64)	4,000
5	7 (3.18)	48,000	6 (2.57)	32,000
6	6 (2.60)	36,000	6 (2.52)	36,000
7	6 (2.77)	44,000	6 (2.66)	48,000
8	8 (3.07)	16,000	7 (2.62)	24,000
9	6 (2.57)	28,000	6 (3.07)	88,000
10	6 (1.79)	8,000	5 (2.71)	100,000

Note: In the column under "Highest", the numbers on the left refer to "number of hits"; the numbers on the right refer to Luce ratio.

It is noteworthy that 100,000 sweeps may seem excessive. There were two reasons for using so large amount of training sweeps. First, the number of sweeps heavily depended on arbitrary parameters such as the net's "learning rate" and "momentum". In this case, learning rate was 0.02 and momentum was 0.08. Both were very low. Second, using a large number of sweeps could make sure that the potential best performance was reached.

4.4 Conclusion: Implications for Language Acquisition

The structural property of linguistic knowledge (Chomsky, 1986, Cook, 1988, Haegeman, 1991, Pinker, 1995) is one of the most important empirical discoveries for the study of language in modern times. Linguistic rules are structure-dependent. The building blocks of sentences are nouns, verbs, clauses etc., but not words. Consequently, one important question for the students of language acquisition is where those word categories come from. Many linguists , psychologists and other cognitive scientists believe that certain categories are innate. However, based on ESRN, Elman (1990, 1991, 1993) argued that those word categories (even

the very notion of word) are learnable through unstructured experience (e.g. speech stream represented as binary vectors in ESRN), therefore there is no need to ascribe those word categories to be innate. Obviously, whether Elman's argument is convincing or not depends upon if ESRN can go beyond the co-occurrence statistics of the training data and acquire the genuine structural knowledge. What this study tried to demonstrate was that in the same knowledge-transfer paradigm, human subjects can transfer the abstract syntactic knowledge to an appropriate new situation in which the lexicon but not the syntax was changed from the previous training; while ESRN with sufficient experience had to start from scratch to learn a new language despite the syntax was still the one used in the previous training data. The empirical finding of this study, though not exactly as predicted, did undermine Elman's claim of ESRN' learning capacity. In order to make a stronger case showing that ESRN cannot go beyond co-occurrence statistics and acquire abstract structural knowledge, future study should further explore the Structural Knowledge Transfer Effect both in human and computer simulation in detail. Specifically, in human case, the question whether Both-Change group is

different from Lexicon-Change group or Null-Change Group must be resolved; in simulation case, ESRN should be exposed to other conditions, namely Both-Change, Null-Change and Syntax-Change. In addition, the method of measuring ESRN's performance should also be modified.

In this study, only the most active unit in the 10-unit output layer was taken as the prediction and compared with the testing sequence which the net had never seen. This measurement may underestimate the nets' knowledge of the syntax, if any. It is noteworthy that at each node of the finite state grammar(see Figure 1.2), there are two possible pathes to follow for producing the next letter. This is like in natural language, both "noun-verb-noun" and "noun-verb-verb" may be grammatical. The experimenter can arbitrarily choose the probability of each path in generating the next letter. In this study, each path was given 50% opportunity in generating the next letter. Thus, a network even with perfect understanding of the syntax underlying the stimuli still cannot predict the next letter with absolute certainty. Therefore, taking only the most active unit as the nets' prediction is not fair to the nets. Instead, both the most active and the second most active should be taken

as predictions and compared with two possible letters dictated by the syntax. Table 4.2 illustrates with a 8-letter testing sequence, how the nets can be scored.

One caveat is that though linguistic knowledge is structural in nature, it is not true that linguistic knowledge is the only structural knowledge human mind has. One implication of this caveat is that human subjects may not use the same mechanism in learning the artificial language used in this study as that they use in acquiring their first language in childhood. However, this study is empirically sufficient and logically sound to cast significant doubts on Elman's argument that ESRN acquires structural knowledge by being exposed to unstructured stimuli and the mechanism used by ESRN in learning the finite state grammar is essentially what children use in language acquisition.

Connectionism, especially its underlying technology, is a new approach to the understanding of cognition. However, neither the idea that the mechanism of mind is essentially associations of mental units nor the objection to the very idea of association is new. In 1948, K. S. Lashley made some comments on various serial behaviors including language.

Table 4.2 A Modified Measurement of Nets' Performance

Target	PAT	MA	2ndMA	Score
#5	#1	#4	#1	1
#5	#1	#1	#1	1
#1	#5	#1	#1	1
#4	#2	#4	#3	1
#3	#4	#3	#4	2
#5	#2	#5	#2	2
#2	#2	#5	#1	0
#1	#3	#3	#5	1
			total:	9

Note: The numbers under "Target" refer to the units that flip on in the 10-unit vector. For example, #5 means that the fifth unit should be flipped on, i.e., "00001000". "PAT" refers to "possible alternative target". "MA" refers to "most active".

"It has been found in studies of memorization of nonsense syllables that each syllable found in the series has associations, not only with adjacent words in the series, but also with more remote words. The words in the sentence have, of course, associations with more remote words as well as adjacent ones. However, the combination of such direct associations will not account for grammatical structure..... It is certain that any theory of grammatical form which ascribes it to direct associative linkage of the words of the sentence overlooks the essential structure of speech. The individual items of the temporal series do not in themselves have a temporal 'valance' in their associative connections with other elements. The order is imposed by some other agent."

This comment was made in the Hixon Symposium on Cerebral Mechanisms in Behavior. And it was later cited and recommended in Chomsky's landmark critique of Behavioristic account of language--Review of B.F. Skinner, Verbal Behavior.(Chomsky, 1959) It may be fair to say that ESRN is merely a modern incarnation of the associationism that Lashley denounced half a century ago.

APPENDIX A

HUMAN SUBJECT CONSENT FORM

I understand that I will participate in a research project on human memory.

I understand that I will be presented with English letter sequences as stimuli. The experiment will consist of two sessions. Each session is worth 1.5 credits. My memory performance, but not my name will be recorded.

I understand that I am free to discontinue participation at any time and still receive credit. I agree to participate in this project.

Subject's Signature:

Date of Experiment:

APPENDIX B

INSTRUCTION TO HUMAN SUBJECTS

This is a memory test. The experiment will consist of two sessions with a 5-minute break between them. In each session you will be presented 9 sets of letter-sequences; each set has two letter-sequences. Each letter sequence will be presented letter by letter. There will be a two-second pause between letters. Each letter sequence will be preceded by a "$" and the end of a sequence will be indicated by a "*". After each set is presented, you will be asked to reproduce the two sequences in the order shown. Only after you have reproduced a set correctly, will you be allowed to go on. Good luck and get ready!

Press the "G" key for go and type "return" when you are ready.

APPENDIX C

THE TRAINING DATA FOR THE NETWORKS

TTS, TTXVPS, VXXXXXVS, TPTXVS, TPPPTXVS, TPPPPPTS, VXVPXVPS, VXVPS, VXXXXVPS, TPTS, VVPXVS, VVPXXVPS, TPPTS, TTXVPXVS, TPPTXXVS, VXVS, VXVPXXVS, TTXXXVPS.

BIBLIOGRAPHY

Anderson, James A. (1983) Cognitive and Psychological Computation with Neural Models. *IEEE Transactions on Systems, Man, and Cybernetics*, Vol, SMC-13, No. 5.

Chomsky, Noam. (1957) *Syntactic Structure*. The Hague: Mouton.

Chomsky, Noam. (1959) Review of B.F. Skinner, Verbal Behavior. *Language*, 35, 26-58.

Chomsky, Noam. (1986) *Knowledge of Language: Its nature, Origin and Use*. New York, N.Y., Praeger.

Cook, V.J. (1988) *Chomsky's Universal Grammar: An Introduction*. Cambridge, Massachusetts, Blackwell Publishers.

Dienes, Zoltan. (1992) Connectionist and Memory-Array Models of Artificial Grammar Learning. *Cognitive Science*, 16, 41-79.

Elman, Jeffrey L. (1990) Finding Structure in Time. *Cognitive Science*, 14, 179-221.

Elman, Jeffrey L. (1991) Distributed Representations, Simple Recurrent Networks, and Grammatical Structure. *Machine learning*, 7, 195-225.

Elman, Jeffrey L. (1993) Learning, and Development in Neural Networks: the Importance of Starting Small. *Cognition*, 48, 71-99.

Gelder, Tim Van (1990) Compositionality: A Connectionist Variation on A Classical Theme. *Cognitive Science*, 14, 355-384.

Haegeman, Liliane. (1991) *An Introduction to Government and Binding Theory*. Cambridge, Massachusetts, Blackwell Publishers.

Lashley, K. S. (1948) The problem of Serial Order in Behavior. in *Cerebral Mechanisms in Behavior*. (1967) New York and London, Hafner Publishing Company.

Marcus, Gary. (1993) Commentary on Jeffrey Elman's "Learning and Development in Neural networks: the Importance of Starting Small" manuscript, Department of Psychology, University of Massachusetts at Amherst.

Marcus, Gary et al. (1992) Overregularization in Language Acquisition, *Monographs of SRCD*, serial No. 228 Vol. 57, No. 4.

Marcus, Gary (1995) German Inflection: The Exception That Proves the Rule. *Cognitive Psychology*, 29.

Newell, A. (1980) Physical Symbol Systems. *Cognitive Science*, 4, 135-183.

Pinker, Steven., & Mehler, J. (1988) Preface. *Cognition*, 28.

Pinker, Steven., & Prinice, Alan. (1988) On Language and Connectionism: Analysis of A Parallel Distributed Processing Model of Language Acquisition. *Cognition*, 28, 73-193.

Pinker, Steven., (1994) *The Language Instinct*. New York, NY, W. Morrow and Co.

Pylyshyn, Z.W. (1984) *Cognition and Computation: Toward a Foundation for Cognitive Science*. Cambridge, Mass: MIT Press.

Reber, Arthur S. (1967) Implicit Learning of Artificial Grammars. *Journal of Verbal Learning and Verbal Behavior*, 6, 885-863.

Reber, Arthur s. (1989) Transfer of Syntactic Structure in Synthetic Languages. *Journal of Experimental Psychology*, Vol. 81, No. 1, 115-119.

Reber, Arthur S. (1989) Implicit Learning and Tacit Knowledge. *Journal of Experimental Psychology: General*, Vol. 118, No. 3, 219-235.

Rumelhart, D.E. & McClelland, J.L. (1986) On Learning the Past Tense of English Verbs. *Parallel Distributed Processing*, Vol 2. Cambridge, Mass: MIT Press.

Servan-Schreiber, David., & Cleeremans, Axel., & McClelland, James L. (1988) Encoding Sequential Structure in Simple Recurrent Networks. *Tech. Re. CMU-CS-88-183 Carnegie Mellon University.*