Dealing with Semantic Anomalies in a Connectionist Network for Word Prediction

by

Janet Hui-wen Hsiao

B.S., National Taiwan University, Taipei, Taiwan, 1999

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCES in the School of Computing Science

Janet Hui-wen Hsiao 2002

SIMON FRASER UNIVERSITY

July 2002

All rights reserved. This work may not be reproduced in whole or in part, by photocopy or other means, without the permission of the author.
APPROVAL

Name: Janet Hui-wen Hsiao
Degree: Master of Sciences
Title of thesis: Dealing with Semantic Anomalies in a Connectionist Network for Word Prediction

Examiner Committee:
- Dr. Ke Wang
  Chair
- Dr. Robert F. Hadley
  Senior Supervisor
- Dr. Fred Popowich
  Supervisor

/Dr. Raymond Jennings
External Examiner

Date Approved: Aug 7, 2002
PARTIAL COPYRIGHT LICENSE

I hereby grant to Simon Fraser University the right to lend my thesis, project and extended essay (the title of which is shown below) to users of the Simon Fraser University Library, and to make partial or single copies only for such users or in response to a request from the library of any other university, or other educational institution, on its own behalf or for one of its users. I further agree that permission for multiple copying of this work for scholarly purposes may be granted by me or the Dean of Graduate Studies. It is understood that copying or publication of this work for financial gain shall not be allowed without my written permission.

Title of Thesis/Project/Extended Essay

__________________________

Dealing with Semantic Anomalies in a Connectionist Network for Word Prediction

__________________________

Author:

Janet Hui-wen Hsiao

__________________________

Aug. 9, 2002

(date)

(signature)
Abstract

Humans are able to recognize a grammatically correct but semantically anomalous sentence. On the task of predicting the range of possible next words in a sentence, given the current word as the input, however, many networks (e.g. Elman, 1990, 1993; Christiansen & Chater, 1994; Hadley et al, 2001) that have been proposed are capable of displaying a certain degree of systematicity, but fail in recognizing anomalous sentences.

We believe that humans require both syntactic and semantic information to predict the category of the next word in a sentence. Based on an expansion of Hadley's model (Hadley et al, 2001), we present a competitive network, which employs two sub-networks that discern coarse-grained and fine-grained categories respectively, by being trained via different parameter settings. Hence, one of the sub-networks will have a greater capacity for recognizing the syntactic structure of the preceding words, while the other will have a greater capacity for recognizing the semantic structure of the preceding pattern of words.

Also, we employ a mechanism to switch attention between the predictions from the two sub-networks, in order to make the global network more closely approximate human behavior. The results show that the network is capable of exhibiting strong systematicity, as defined by Hadley (Hadley, 1994a). In addition, it is able to predict in compliance with the semantic constraints implied in the training corpus, and deal with grammatically correct but semantically anomalous sentences. We can conclude that the network has provided a more realistic model for human behavior on the task of predicting the range of possible next words in a sentence.
Acknowledgments

The assistance and guidance given by a number of individuals has been essential for the accomplishment of this thesis. First, I wish to thank my senior supervisor, Dr. Robert F. Hadley, for his encouragement and guidance throughout this research. I am also indebted to him for his valuable help and ideas regarding this research, and instructions on academic writing. Also, my gratitude goes to my supervisor Dr. Fred Popowich and my external examiner Dr. Raymond Jennings, for their time and insightful comments that have helped the accomplishment of this thesis.

I am grateful to Dr. Jeff Elman, for providing the back-propagation training algorithm TLearn, and for his valuable help on training with TLearn. Special thanks go to Marlene Sawatsky, who has given me advice on improving academic writing style.

I would like to express my sincere thanks to my fellow graduate students Ann Grbavec and Rasmus storjohann, who have been exchanging ideas with me. I am also grateful to Ceci Chen, Stephen Tse, Jack Yeung, Ben Law, Joyce Lam, Peter Chen, and other friends for offering encouragement and help. My special thanks go to Bruce Huang, for his support, advice, and help during this period.

Most of all, I am deeply in debt to my beloved parents, for always being supportive and giving love unconditionally.
Dedication

To my beloved parents.
Content

Abstract ............................................................................................................................ iii
Acknowledgements ..................................................................................................... iv
Dedication ...................................................................................................................... v
List of Tables ............................................................................................................... viii
List of Figures ............................................................................................................. ix
Chapter 1 Introduction ............................................................................................... 1
  1.1 Connectionism and Systematicity ........................................................................ 1
  1.2 Systematicity in a Simple Recurrent Network .................................................... 3
  1.3 Semantic Constraints and Anomalies .................................................................. 4
  1.4 Thesis Organization ............................................................................................. 6
Chapter 2 Related Works and Experiments .................................................................. 8
  2.1 Simple Recurrent Network .................................................................................. 8
    2.1.1 Simple Recurrent Network ............................................................................ 9
    2.1.2 On the Issue of Strong Systematicity ............................................................. 10
    2.1.3 Christiansen and Chater's Experiments ......................................................... 11
    2.1.4 Elman's Response to Strong Systematicity .................................................... 13
  2.2 A Hebbian-Competitive Network ....................................................................... 14
  2.3 Experiments with a Simple Recurrent Network .................................................. 16
    2.3.1 Task and System Overview ........................................................................... 17
    2.3.2 Training and Test Corpora .......................................................................... 18
    2.3.3 Algorithms and Architectural Details ............................................................. 19
    2.3.4 Test Phase and Analysis of Results ................................................................. 20
Chapter 3 System and Task Overview ....................................................................... 26
  3.1 System Overview .................................................................................................. 26
3.2 Syntax and Semantic Constraints .......................................................... 30
3.3 Input and Output .................................................................................. 32
3.4 Three parts of the Task ........................................................................ 33
  3.4.1 Systematicity ................................................................................. 33
  3.4.2 Semantic Constraints ...................................................................... 34
  3.4.3 Semantic Anomalies ....................................................................... 34

Chapter 4 Training Phase ............................................................................ 37
  4.1 Training Corpus .................................................................................. 37
  4.2 Training Algorithms ........................................................................... 39
    4.2.1 Weight Initialization ..................................................................... 39
    4.2.2 Winner Selection .......................................................................... 39
    4.2.3 Weight Modification in a Competitive Cluster ............................... 40
    4.2.4 Reverse Competitive Learning Algorithm .................................. 41
  4.3 Training Cycle .................................................................................... 43

Chapter 5 Test Phase .................................................................................. 45
  5.1 Test Corpus ....................................................................................... 45
  5.2 Test System Overview ......................................................................... 47
  5.3 Test Cycle in a Sub-network ............................................................... 49
  5.4 Coherence Reinforcement .................................................................. 53
  5.5 Evaluation .......................................................................................... 55

Chapter 6 Experimental Results .................................................................. 57
  6.1 Systematicity ..................................................................................... 58
  6.2 Semantic Predictions .......................................................................... 63
    6.2.1 Simple Sentences .......................................................................... 63
    6.2.2 Embedded Sentences .................................................................... 78
    6.2.3 Sentences with a Prepositional Phrase ....................................... 88
  6.3 Semantic Anomalies .......................................................................... 93
Chapter 7  Discussion and Future Directions................................................................. 102
Bibliography.................................................................................................................. 108
Appendix A: Assignment of Features to Words ............................................................ 108
## List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1</td>
<td>Three sets of semantically anomalous sentences involving semantic anomalies in different positions of a sentence</td>
<td>46</td>
</tr>
<tr>
<td>6.1</td>
<td>The parameter settings</td>
<td>58</td>
</tr>
<tr>
<td>6.2</td>
<td>The cosine values obtained for different groups of nouns in the left sub-network, right-sub-network, and presentation layer</td>
<td>60</td>
</tr>
<tr>
<td>6.3</td>
<td>Three sets of semantically anomalous sentences involving semantic anomalies in different positions of a sentence</td>
<td>94</td>
</tr>
</tbody>
</table>
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>The architecture of Elman's recurrent network (1990)</td>
</tr>
<tr>
<td>2.2</td>
<td>The architecture of Hadley's Hebbian-competitive network (Hadley et al, 2001)</td>
</tr>
<tr>
<td>2.3</td>
<td>The format for generating sentences</td>
</tr>
<tr>
<td>2.4</td>
<td>Each word is assigned a probability of occurrence in the training corpus. Font sizes indicate each probability</td>
</tr>
<tr>
<td>2.5</td>
<td>The average activation of restricted and unrestricted words after human verbs, with 20 units in the hidden layer (The number of epochs ranges from 1,000 to 30,000)</td>
</tr>
<tr>
<td>2.6</td>
<td>The average activation of restricted and unrestricted words after all other verbs, with 20 units in the hidden layer (The number of epochs ranges from 1,000 to 30,000)</td>
</tr>
<tr>
<td>2.7</td>
<td>The average activation of restricted and unrestricted words after human verbs, with 30 units in the hidden layer (The number of epochs ranges from 1,000 to 30,000)</td>
</tr>
<tr>
<td>2.8</td>
<td>The average activation of restricted and unrestricted words after all other verbs, with 30 units in the hidden layer (The number of epochs ranges from 1,000 to 30,000)</td>
</tr>
<tr>
<td>2.9</td>
<td>The average activation of restricted and unrestricted words after human verbs, with 40 units in the hidden layer (The number of epochs ranges from 1,000 to 30,000)</td>
</tr>
<tr>
<td>2.10</td>
<td>The average activation of restricted and unrestricted words after all other verbs, with 40 units in the hidden layer (The number of epochs ranges from 1,000 to 30,000)</td>
</tr>
<tr>
<td>2.11</td>
<td>The average activation of restricted and unrestricted words after human verbs, with 60 units in the hidden layer (The number of epochs ranges from 1,000 to 30,000)</td>
</tr>
</tbody>
</table>
The average activation of restricted and unrestricted words after all other verbs, with 60 units in the hidden layer (The number of epochs ranges from 1,000 to 30,000) ................................................................. 24
3.1 Network architecture ........................................................................................................ 27
3.2 The training network ........................................................................................................ 28
3.3 The grammar for generating training and test sentences ........................................... 30
3.4 Semantic constraints ...................................................................................................... 32
4.1 The initialization of the weights between HL2 and the output layer ....................... 39
4.2 Weight modification in the reverse competitive learning algorithm ...................... 42
5.1 Network architecture during test phase ........................................................................ 47
5.2 The detailed structure on the top of the two output layers ...................................... 48
5.3 The predictions when the current input sentence is a normal sentence “boys eat cookies” ......................................................................................................................... 51
5.4 The predictions when the current sentence is the semantically anomalous sentence, “boys eat rocks” ........................................................................................................... 52
6.1 Predictions following a noun appearing in the subject position of a sentence for the left sub-network ........................................................................................................ 64
6.2 Predictions following a sequence of words having the form N V for the left sub-network ......................................................................................................................... 65
6.3 Predictions following a sequence of words having the form Noun Verb Noun for the left sub-network ............................................................................................................. 66
6.4 Predictions following a Noun-Human in the subject position of a sentence for the right sub-network ........................................................................................................ 66
6.5 Predictions following a Noun-Anim in the subject position of a sentence for the right sub-network ........................................................................................................... 68
6.6 Predictions following a Noun-Inanim in the subject position of a sentence for the right sub-network ....................................................................................................... 69
6.7 Predictions following a sequence of words having the form Noun-Human Verb-
*Eat* for the right sub-network ................................................................. 70
6.8 Predictions following a sequence of words having the form Noun-Human Verb-
*Percept* or Noun-Human Verb-Tran for the right sub-network ................. 71
6.9 Predictions following a sequence of words having the form Noun-Human Verb-
*Stren* for the right sub-network ............................................................. 72
6.10 Predictions following a sequence of words having the form Noun-Inanim Verb-
*Hit* for the right sub-network ................................................................. 73
6.11 Predictions following a sequence of words with a form Noun-Human Verb-
*Eat* Noun-Food for the right sub-network .................................................. 74
6.12 Predictions following a sequence of words having the form Noun-Human Verb-
*Percept* Noun-Inanim for the right sub-network ........................................ 75
6.13 Predictions following a sequence of words having the form Noun-Human Verb-
*Tran* Noun-Human for the right sub-network ............................................ 75
6.14 Predictions following a sequence of words having the form Noun-Human Verb-
*Tran* Noun-Human for the right sub-network ............................................ 76
6.15 Predictions following a sequence of words having the form Noun-Human Verb-
*Stren* Noun-Inanim for the right sub-network ............................................ 76
6.16 Predictions following a sequence of words having the form Noun-Inanim Verb-
*Hit* Noun-Human for the right sub-network .............................................. 77
6.17 Predictions following a sequence of words having the form Noun that for the left
sub-network ..................................................................................................... 79
6.18 Predictions following a sequence of words having the form Noun that Verb for the
left sub-network ............................................................................................... 80
6.19 Predictions following a sequence of words having the form Noun that Noun for the
left sub-network ............................................................................................... 80
6.20 Predictions following a sequence of words has the form that Verb Noun for the
left sub-network ............................................................................................... 81
Predictions following a sequence of words having the form *that Noun Verb* for the left sub-network ................................................................. 82

Predictions following a sequence of words having the form *Noun-Human that* for the right sub-network .......................................................... 83

Predictions following a sequence of words having the form *Noun-Inanim that* for the right sub-network .......................................................... 84

Predictions following a sequence of words having the form *Noun-Human that Verb-Eat* for the right sub-network ........................................ 85

Predictions following a sequence of words having the form *Noun-Human that Verb-Percept or Noun-Human that Verb-Tran* for the right sub-network .......... 85

Predictions following a sequence of words having the form *Noun-Human that Verb-Stren* for the right sub-network ........................................ 86

Predictions following a sequence of words having the form *that Verb-Eat Noun-Food* for the right sub-network (The predictions following a sequence of words having the form *that Verb Noun* all have a similar distribution) ..................... 87

Predictions following a sequence of words having the form *that Noun-Human Verb-Tran* for the right sub-network (The predictions following a sequence of words having the form *that Noun Verb* all have a similar distribution) ...................... 88

Predictions following a sequence of words having the form *(Noun) Verb Noun with* for the left sub-network .................................................. 89

Predictions following a sequence of words having the form *(Noun) Verb Noun from* for the left sub-network ................................................. 90

Predictions following a sequence of words having the form *Verb-Eat Noun-Food with* for the right sub-network (The predictions following a sequence of words having the form *Verb-Tran Noun-Human with or Verb-Tran Noun-Anim with* also have a similar distribution) ................................................. 91

Predictions following a sequence of words having the form *Verb-Stren Noun-Inanim with* for the right sub-network ........................................ 91
Predictions following a sequence of words having the form \((\text{Noun-Inanim}) \text{ Verb-Hit Noun-Human with}\) for the right sub-network. (The predictions following a sequence of words having the form \(\text{Verb-Hit Noun-Anim with, Verb-Hit Noun-Inanim with, Verb-Hit Noun-Human from, Verb-Hit Noun-Anim from, or Verb-Hit Noun-Inanim from}\) also have a similar distribution). ................................................................. 92

The activation of each feature in the output layer of the right sub-network when the current input is a semantically anomalous sentence with a novel semantic pattern.................................................................................................................................................. 95

The predictions from the left sub-network when the preceding sequence of words is “boys hit”, “cookies eat” or “rocks eat”. The same predictions can be also found in the presentation layer ........................................................................................................................................ 96

The predictions following a sequence of words “boys eat rocks”, “rocks eat boys”, or “cookies eat boys” for the left sub-network. The same predictions can be also found in the presentation layer ........................................................................................................................................ 97

The predictions following a sequence of words “girls eat rocks with” for the left sub-network. The same predictions can be found in the presentation layer .......... 98

The predictions following a sequence of words “rocks eat girls from” for the left sub-network. The same predictions can be found in the presentation layer .......... 98

The predictions following a sequence of words “girls eat rocks with sandwiches” or “rocks eat girls from boys” for the left sub-network. The same predictions can be found in the presentation layer ........................................................................................................................................ 99

The predictions following a sequence of words “boys that hit” for the left sub-network. The same predictions can be found in the presentation layer .......... 100

The predictions following a sequence of words “boys that hit books hit books” for the left sub-network. The same predictions can be found in the presentation layer ........................................................................................................................................ 101
Chapter 1 Introduction

Chapter 1

Introduction

1.1 Connectionism and Systematicity

The understanding of the human cognitive processes has been a goal of cognitive scientists for several decades and has not yet been attained. In the 1950’s, with the advent of digital computers, researchers started to seek the answer to the puzzle of human cognitive processes from the analogy of computers. They employed a computational framework in which information could be stored and retrieved to guide behavior. According to this framework, human cognition is based on symbols, rules, and operations on the symbols. This approach has been referred to as classicism (or symbolism) in cognitive science.

An alternative approach emerged in the 1980’s in the form of connectionist models, which were inspired by the parallel-distributed information processing of neural connections in the brain. This movement, referred to as connectionism, hopes to explain the human cognitive process using artificial neural networks. From then on, the dispute between classicism and connectionism has not ceased. The classicists criticize the implementational nature of connectionism, while connectionists continue the work of addressing experimental data to refute those criticisms.
Chapter 1 Introduction

One of the main criticisms of connectionist models is their apparently weak capacity of exhibiting systematicity in human thought and language production (Fodor and Pylyshyn, 1988). Consequently, in the last twenty years, a substantial research literature has been established on this issue (McClelland and Kawamoto, 1986; Chalmers, 1990; Elman, 1990, 1993, 1998; Pollack, 1990; Smolensky, 1990; Christiansen and Chater, 1994; Hadley, 1994a, 1994b; Niklasson and van Gelder, 1994; Phillips, 1994, 2000; Hadley et al, 1997, 1999, 2001; Marcus, 1998). Systematicity can be defined thus: "the ability to produce / understand some sentences is intrinsically connected to the ability to produce / understand certain others" (Fodor and Pylyshyn, 1988). For example, a native English speaker who knows how to understand "Mary loves John" in English also knows how to understand "John loves Mary". Similarly, a network with the capacity of exhibiting systematicity should be able to generalize the use of a word to a new combination of components or a new syntactic position.

Since 1990, several cognitive scientists have concentrated on the capacity of simple recurrent networks (SRNs) to display systematicity in the task of predicting the range of possible next words in a sentence (Elman, 1990, 1993, 1998; Christiansen and Chater, 1994; Hadley, 1994a, 1994b; Marcus, 1998; Phillips, 2000). An SRN is a standard feed-forward network with an extra recurrent layer (see section 2.1). As discussed in (Hadley, 1994a), SRNs display systematicity only in very limited contexts and for a small fraction of words involved. Connectionist models that are capable of displaying robust forms of systematicity do exist. The majority of these networks, however, employ classical representational layers and combinatorially-endowed wiring (Hadley and Hayward, 1997; Hadley and Cardei, 1999), which are the typical characteristics of a classical model.

Hadley presented a Hebbian-competitive network (Hadley et al, 2001) that satisfies the definition of strong syntactic systematicity (Hadley, 1994a), without previously acquired classical semantic representations or the presence of combinatorially-

---

1 Or rather, we could say that the behavior of the network is to "anticipate" the range of possible next words in a sentence.
endowed wiring. Strong systematicity, briefly stated, requires a connectionist network to correctly generalize the use of a significant portion of its vocabulary to novel syntactic positions, within both simple sentences and embedded sentences. Hadley’s research shows that this model not only offers a purely self-organizing form of connectionist learning to approximate the behavior of a SRN, but it also possesses some advantages over standard SRNs, such as the speed of learning and the degree of systematicity it can achieve.

1.2 Systematicity in a Simple Recurrent Network

While several cognitive scientists doubt the SRNs’ ability to exhibit robust forms of systematicity, when a rich grammar is used and a larger vocabulary is involved (Hadley, 1994a; Marcus, 1998), Elman in 1998 presented the result of an experiment and claimed that SRNs are capable of exhibiting robust forms of systematicity. In his experiment, the network’s task was to predict the next word in a sentence, given a prior sequence of words as input. During training, the word “boys” was precluded from appearing in the direct object position. The results appeared to show that the network was able to predict “boy” as a grammatically acceptable item in the direct object position during later testing.

However, as indicated by Elman, although the parental language input for children is both very rich and extensive (Hart & Risley, 1995), children certainly do not hear all possible words in all possible syntactic contexts. In addition, as one’s vocabulary increases, the probability of encountering many words with limited exposure also increases. Thus, it is very unlikely that children only have a single word that does not appear in all possible syntactic contexts in their parental language input. Elman did not address this issue in his 1998 experiment, and one of our concerns is whether SRNs can still exhibit robust forms of systematicity when more words are initially encountered only in a single syntactic position. A discussion and an experiment regarding this issue are presented in Chapter 2.
1.3 Semantic Constraints and Anomalies

The issue of the semantic constraints over different subcategories of nouns was barely addressed in (Hadley et al, 2001). An example of such semantic constraints is that the word following the verb “eat” should be an edible noun, such as “cookies”. By definition, semantic constraints are the semantic patterns we habitually encounter, and a semantically anomalous sentence is a sentence containing semantic patterns that violate the semantic nature of these semantic constraints. An example of semantically anomalous sentences is “boys eat rocks”, which violates the semantic constraint that the word following the verb “eat” should be an edible noun. Here we are dealing with a subset of such sentences.

Ideally, the network is expected to make semantic predictions for normal sentences (i.e. grammatically and semantically correct sentences) in compliance with semantic constraints implied in the training corpus, and make syntactic predictions, instead of semantic predictions, for grammatically correct but semantically anomalous sentences. If we train the network to generalize the input words more and recognize fewer subcategories, it may not have the capacity to discover all the semantic constraints implied in the training corpus. On the other hand, if we train the network to recognize more subcategories, it will lose the information about general categories and hence lose the ability to generalize well enough to make syntactic predictions for a grammatically correct but semantically anomalous sentence. Therefore, to train the network with suitable parameters, which will enable the network to handle both situations well, is a challenging task.

Based on an expansion of Hadley’s work in (Hadley et al, 2001), we create a more challenging training corpus, such that in addition to the occurrences of relative

---

2 The definition here is a “working definition” only for purposes of this thesis.
3 A training corpus is a collection of recorded utterances used for training a computational model, especially for the descriptive analysis of a language.
Chapter 1 Introduction

clauses and prepositional phrases, all sentences are generated according to a set of semantic constraints. We believe that humans have the concept of hierarchy of categories, which is helpful in dealing with semantic anomalies. So that when we encounter a semantically anomalous sentence such as "boys eat rocks", we can still recognize it as a grammatical sentence. Hence, a mechanism to learn information from both general categories (e.g., noun) and subcategories (e.g., human noun) is required in the network design.

A good way to achieve this is to use two sub-networks which respectively learn information about categories and subcategories, by using different training parameter settings. We assume that a network can recognize a grammatical sentence if a period is predicted at the right place and if it can make predictions according to correct English grammar. A failure to make substantial semantic predictions suggests that current input contains a novel semantic pattern that the network does not habitually encounter during training, i.e., a semantically anomalous sentence. Moreover, during testing, we use a mechanism to coordinate the information exchange between the two sub-networks, in the hope that it will help the network make predictions close to human behavior.

With regard to the task of examining the degree of systematicity this network is capable of displaying, during training, some nouns are precluded from appearing as grammatical subjects (NS nouns), and some are precluded from appearing as grammatical direct object (NO nouns). During later testing, the network is able to generalize the use of these restricted nouns to novel syntactic positions. Moreover, we demonstrate that the employment of a coherence reinforcement process (see chapter 6 for details) even helps the network make better predictions than those from any of the two sub-networks alone.

Both the network program and the program for generating training and testing corpora are implemented in C++. The program for generating corpora generates sentences according to a set of predefined syntax and semantic constraints (see section 3.2). Parameters can be manually altered to change the numbers of sentences with different syntax to be generated (see section 3.2) and, for training purpose, to restrict words from appearing in certain grammatical positions (see Chapter 4 for details).
Chapter 1 Introduction

Sentences are generated randomly according to these parameters, and are printed out into a text file. The network program then takes this text file as input for training or testing. During testing, the network program also prints corresponding output into a text file for later analysis (details for training and testing procedures can be found in Chapters 4 and 5).

Finally, we would like to emphasize that the network proposed here is only intended to provide a possible language acquisition framework to deal with semantic anomalies within a connectionist network. It is not meant to provide a general language acquisition mechanism. Nevertheless, we believe that some mechanism similar to what we have proposed here might conceivably be found in a larger language acquisition system to explain how humans deal with semantically anomalous sentences.

1.4 Thesis Organization

The remainder of the thesis is organized as follows. Chapter 2 discusses related works regarding the task of predicting the range of possible next words in a sentence, including Elman’s, Christiansen and Chater’s, and Hadley’s research regarding the issue of systematicity. Also, the result of an experiment with a simple recurrent network in response to Elman’s 1998 work is also presented. Chapter 3 gives an overview of the architecture of the proposed network, and states the network’s tasks and goals. The syntax and semantic constraints for generating training and test corpora and the encoding of input and output of the network are explained. The three parts of the task, systematicity, semantic constraints, and semantic anomalies, are also presented respectively. Chapter 4 and Chapter 5 present the corpora and algorithms used during the training and test phases, such as the generation of corpora, weight initialization and modification, and the training and testing cycle. Also, a reverse competitive learning algorithm is introduced in chapter 4, and the details of the coherence reinforcement process are given in chapter 5. The experimental results are presented in Chapter 6. Three sets of experiments to examine the network’s capacity of exhibiting systematicity, and
Chapter 1 Introduction

responses to semantic constraints and semantic anomalies are explained and the results are discussed. Finally, conclusion and further discussion, regarding the issues of systematicity and the interaction between syntactic and semantic information, are conducted in Chapter 7.
Chapter 2

Related Works and Experiments

This chapter covers a variety of works and generalization experiments related to the research herein. Section 2.1 reviews Elman's simple recurrent network, which has gained a great deal of attention during the past decade, concerning the task of predicting the range of possible next words in a sentence. Christiansen and Chater's experiments with a simple recurrent network are also presented in section 2.1. The later experiments are directed at the capacity of a simple recurrent network for generalizing the use of a word to different grammatical positions. Section 2.2 reviews Hadley's Hebbian-Competitive network, with regard to the task of predicting the semantic features of the next word in a sentence. In section 2.3, we present the results of a separate experiment in which we try to duplicate Elman's experiments (Elman, 1998), with a slight modification of the training corpus, to further challenge the generalization capacity of a simple recurrent network.

2.1 Simple Recurrent Networks

Jeff Elman, in his paper “Finding Structures in Time” (Elman, 1990), introduced a connectionist network that has recurrent links to provide the network with dynamic
Chapter 2 Related Works and Experiments

memories. This network is referred to as a *simple recurrent network* (SRN). As shown in Figure 2.1, the network consists of an input layer, an output layer, a hidden layer, and a context layer. A layer consists of a sequence of separate units. The links from the hidden layer to the context layer are one-to-one copy links. Other links are all fully connected and trainable links. That is, every node in a layer is connected through trainable links to every node in the layer directly above.

The network is trained through the back-propagation algorithm. After activation from the input layer spreads up to the hidden layer, the activation pattern that results in the hidden layer is copied from hidden units to context units on a one-to-one basis, and later fed back to the hidden layer together with the input in the next cycle.

![Figure 2.1: The architecture of Elman's recurrent network (1990). The activations are copied from hidden units to context units on a one-to-one basis. Dotted links represent entire sets of trainable connections.](image)

### 2.1.1 Initial Discovery of SRN's Capacity

In (Elman, 1990), Elman experimented with a simple recurrent network to see if it was able to discover lexical classes from a sequence of sentences. To begin with, he generated a training corpus consisting of simple sentences, according to a simple grammar. Each word in the training corpus was encoded with a “local encoding” of zeros and a single bit
set to one. Whenever the word was presented, the corresponding “bit” was set to one. All other units in the input layer are set to zero. This encoding scheme guaranteed that each vector representation of a word was orthogonal to every other vector, and reflected no information about the word class or meaning.

During training, each word in a training sentence was presented to the input layer one at a time, without breaks between continuous sentences. The target output for the network was the word following the given input word. During testing, the results showed that the predictions in the output layer reflected the likelihood ratio of potential successors. Elman concluded that instead of memorizing the order of words appearing in the training corpus, the network revealed a capacity for generalizing the classes of words from the co-occurrence statistics and the composition of these classes.

Elman also examined the internal representations of the network developed in the hidden layer during testing. After a hierarchical clustering analysis, an implicit hierarchy of categories was discovered. This suggested that the network was able to develop internal representations for input vectors, to reflect information about their potential successors and their lexical categories, solely from the input word order.

2.1.2 On the Issue of Strong Systematicity

In 1994, Hadley defines three different levels of systematicity with increasing difficulty for a learning system to meet: weak, quasi, and strong systematicity. According to Hadley, a connectionist network (or “c-net” in Hadley’s terminology) “exhibits at least weak systematicity if it is capable of successfully processing (by recognizing or interpreting) novel sentences, once the c-net has been trained on a corpus of sentences which are representative”. A training corpus is representative if “every word (noun, verb, etc.) that occurs in some sentence of the corpus also occurs (at some point) in every permissible syntactic position” (Hadley, 1994a).

Quasi-systematicity can be ascribed to a learning system if “(a) the system can exhibit at least weak systematicity, (b) the system successfully processes novel sentences
Chapter 2 Related Works and Experiments

containing embedded sentences, such that both the larger containing sentence and the embedded sentences are (respectively) structurally isomorphic to various sentences in the training corpus, (c) for each successfully processed novel sentence containing a word in an embedded sentence (e.g., “Bob knows that Mary saw Tom”), there exists some simple sentence in the training corpus which contains that same word in the same syntactic position as it occurs within the embedded sentence (e.g., “Jane saw Tom”) (Hadley, 1994a).

Finally, a system exhibits strong systematicity if “(i) it can exhibit weak systematicity, (ii) it can correctly process a variety of novel simple sentences and embedded sentences containing previously learned words in positions where they do not appear in the training corpus (i.e. the word within the novel sentence does not appear in that same syntactic position within any simple or embedded sentence in the training corpus)” (Hadley, 1994a).

Hadley (Hadley, 1994a) hence argued that the training regime adopted by Elman (Elman, 1988, 1989, 1990) does not appear to afford strong systematicity, since during training every word in the network’s vocabulary had been presented in every possible syntactic position.

2.1.3 Christiansen and Chater’s Experiments

In Christiansen and Chater’s paper “Generalization and Connectionist Language Learning” (Christiansen and Chater, 1994), they discussed and reported some connectionist simulations using SRNs to capture the generalizations that Hadley discussed, and to extend to other examples. They trained the network with sentences generated by a non-deterministic grammar, which is significantly more complex than the one used by Elman (Elman, 1990, 1991). The grammar involved subject noun/verb number agreement, verbs with different argument structures (transitive, intransitive, and optionally transitive), relative clauses with single and multiple embeddings, prepositional phrases, conjunction of noun phrases (e.g., “John and Mary”), and sentential
Chapter 2 Related Works and Experiments

complements (e.g., "John says that Mary runs"). 10,000 sentences were generated according to this grammar for training.

Their first experiment concerned generalization in genitive contexts (e.g., in a sentence with nouns indicating possession, such as "Mary’s dogs …"). During training, the words “girl” and “girls” were precluded from occurring in a genitive context. During testing, they found that the network predicted all nouns following “Mary’s …” except “girl” and “girls”. They further examined predictions following “Mary’s girls …”, where a plural marker was expected. The result showed that the network only provided the plural genitive marker with a small amount of activation. They then concluded that the network was not able to generalize strongly in genitive contexts. However, the small amount of activation of the plural genitive marker following “Mary’s girls”, and the fact that the network was still able to make correct predictions following “Mary’s girls run”, suggested that the network might be able to learn to generalize strongly in genitive contexts, if a different representation was used or the training details were altered.

Another experiment concerned generalization in noun phrase conjunctions. During training, the word “boy” never appeared in a noun phrase conjunction, and hence never preceded a plural verb (e.g., “John and boy run”). During testing, they found that the network was able to predict a plural verb following “John and boy …” or “John and boy from town …”. They concluded that the network was able to produce the strong generalization that a noun phrase conjunction always takes a plural verb, and this generalization can occur across several words.

The authors further argued that the failure of the genitive example illustrated some deficit in the network’s capacity for strong generalization. However, the network was claimed to exhibit a robust form of generalization in the experiment with noun phrase conjunctions. This suggested that the capacity for strong generalization might not be beyond SRNs.

---

This claim is highly contentious, as indicated by Hadley (Hadley, 1994b), since the network had failed to predicted the general category of the word following “and” in the same sentence. That is, only insignificant predictions were assigned to “boy” and “boys” to follow “John and …”.
2.1.4 Elman’s Response to Strong Systematicity

In 1998, in response to Hadley’s and Marcus’s questioning about simple recurrent networks’ capacity for generalization, Elman presented another experiment. His experiment aimed at examining the capacity of a simple recurrent network for generalizing outside the training space.

In Elman’s 1998 experiment, the task of the network was to process a sequence of words, one at a time, taken from a pool of sentences, and to predict successive words. The pool of sentences consisted of only simple sentences, generated according to a simple grammar, in which each verb had different possible arguments. Also, each word was assigned a different frequency of occurrence in the training corpus. Given the grammar, there were only 1,030 different possible sentences in the pool. Various corpora having different sizes, ranging from 20 sentences to 5,000 sentences, were created by taking random sentences from the pool.

Elman first took a training corpus of 1,000 sentences, in which the word “boy” never appears in a direct object position for any verb, and examined whether the network can predict “boy” as a possible direct object following the verb “talk-to”. The network was then trained through 10,000 trials (or “epochs”, i.e. via 10,000 complete passes through the corpus), and predictions were measured at different points in time. The results suggested that in the early stage of training, the network’s predictions conformed fairly closely to the raw statistics of data, and the network could be said to operate a rote strategy. Thus, the network predicted other human nouns more strongly than the word “boy”, following the verb “talk-to”.

As the number of training trials increased, the predictions for “boy” also increased gradually. At 10,000 training trials, “boy” was predicted as a grammatically acceptable item in the direct object position. However, after 20,000 training trials, the gap between the predictions for “boy” and for other human nouns was increasing again. He then concluded that the networks did appropriately generalize the use of words, based on the
Chapter 2 Related Works and Experiments

“behavior-based similarities” (i.e. “what words a word usually appears with”) among words, and the relative amount of data and experience that were available. That is, “boy” was seen in shared context with other human words, and the absence of “boy” in the direct object position is considered irrelevant. Hence, the network predicted “boy” as a grammatically acceptable item in the direct object position, regardless of its absence in the direct object position during training. However, if there was significantly more training, in which “boy” did not appear in direct object position, the network weakened this generalization.

2.2 A Hebbian-Competitive Network

Hadley et al (2001), in their paper “Syntactic Systematicity Arising from Semantic Predictions in a Hebbian-Competitive Network”, proposed a connectionist network that satisfies the definition of strong systematicity (Hadley, 1994a), without classical semantic representations and error feedback.

By contrast with Elman’s experiments on SRNs, the task of Hadley’s network was to predict the semantic category of the next word in a sentence, instead of the syntactic category. Every word in the network’s vocabulary was encoded with distributed semantic representations. That is, each bit of an encoding corresponded to a semantic feature, and the bit was flipped on if the word has the given feature. A comparatively sparse set of training data, generated from a non-deterministic grammar, was created. Besides simple sentences, the training and test corpora also contained sentences with object-relative clauses, subject-relative clauses, and prepositional phrases.

During training, two-thirds of the nouns were restricted to a single syntactic position, and the depth of embedding was restricted to two. These restrictions were dropped during testing. The results showed that despite these restrictions, the strength of predictions for restricted nouns was virtually indistinguishable from those of unrestricted nouns.
Chapter 2 Related Works and Experiments

The network architecture is shown in Figure 2.2. As can be seen, the network contains an input layer, a first hidden layer (HL1), a second hidden layer (HL2), and an output layer. HL1 contains three distinct regions, named (left to right) region A, region B, and region C. A dotted link indicates a set of trainable and fully connected links. That is, each node in the layer below is connected to each node in the layer directly above, through a trainable link. The links between region B and region C are one-to-one copy links with fixed weights +1.

The links between the input layer and region A in HL1, and between HL1 and HL2, are trained via a competitive learning algorithm (von der Malsburg, 1973). In other words, region A of HL1 and the whole of HL2 serve as competitive clusters. In a competitive cluster, only the node that is most activated by the current input is selected as the "winner", and has a chance to adjust its weights in response to this input pattern. Partly for this reason, the algorithm is able to classify the input patterns according to their "similarity".

During a cycle, the content of region A is spread to both region B and HL2. The activation spread to region B will trigger a chain reaction to spread the prior content of
region B to region C. In the next cycle, this new content of region B and region C, together with the new winner generated in region A will form a triadic pattern and will be fed to HL2 (see section 4.3 for more details).

On the other hand, The links between HL2 and the output layer are trained via a simple non-competitive Hebbian learning rule (Hebb, 1949), to ensure that appropriate semantic vectors are predicted in the output layer.

In short, region A in HL1 has a post-training role as a classifier to categorize input vectors, regardless of their positions in a sentence. HL2 has a post-training role as a pattern recognizer to recognize all distinct triadic patterns presented in the three regions of HL1. The authors believed that these post-training roles help the network predict with systematicity, regardless of the depth of tail recursion in a sentence.

The main purpose of Hadley’s work, in contrast to Elman’s, was to show that a shift from local representations to distributed semantic representations could do much better with the problem of systematicity. In a separate experiment, they employed back-propagation to train a standard SRN with the same training and test corpora. The results showed that their original competitive network provides an approximation to SRNs, with advantages of biological plausibility and learning speed over SRNs. The authors believe that they have provided something strongly akin to a competitive implementation of SRNs. Also, their work has shown that systematicity at the syntactic level derives from predictability at a semantic level.

2.3 Experiments with a Simple Recurrent Network

As presented in section 2.1.2, Elman’s 1998 experiment suggested that simple recurrent networks do have a capacity for generalizing outside the training space. However, in his experiment, there was only one word, “boy”, restricted to a single syntactic position. We have previously stated that it is very unlikely that children have only a single word that fails to appear in all possible syntactic contexts in their parental language input. Hence, it
Chapter 2 Related Works and Experiments

is our concern whether SRNs can still exhibit robust forms of systematicity when more words are restricted to a single syntactic position.

In response to Elman’s work, we designed a separate experiment to examine further the network’s capacity for generalization. That is, we restricted more nouns from appearing in the object position, instead of the solitary one in Elman’s experiment, and investigated whether the network can still predict restricted nouns following a verb.

2.3.1 Task and System Overview

The connectionist network used here is a simple recurrent network (Elman 1990), as shown in Figure 2.1. It is designed to process words, one at a time, taken from a pool of sentences, generated according to a simple syntax displayed in Figure 2.3:

NOUN-HUM -> boys | girls
NOUN-ANIM -> dogs | cats
NOUN-INANIM -> books | rocks
NOUN-FOOD -> cookies | sandwiches
VERB-EAT -> eat | bite
VERB-PERC -> smell | see
VERB-TRAN -> see | chase

S -> NOUN-HUM VERB-EAT NOUN-FOOD
S -> NOUN-HUM VERB-PERCEPT NOUN-INANIM
S -> NOUN-ANIM VERB-EAT NOUN-FOOD
S -> NOUN-HUMAN VERB-TRAN NOUN-HUMAN

Figure 2.3: The format for generating sentences.
Chapter 2 Related Works and Experiments

Following Elman's approach, all nouns and verbs are initially assigned linearly independent vector representations. Local encoding is adopted. That is, each word is represented by a single unit within the input and output layers. A unit is flipped on if the corresponding word is presented. Each word of a sentence is presented to the input layer in sequence. Between two consecutive sentences, an additional encoding (i.e. the period), which is represented by the last unit in the vector, is presented to indicate the end of a sentence. The primary task of the network is to predict the syntactic category of the next word.

In order to challenge the generalizational capacity of the network, during training, two of the human nouns (boys and girls) and two of the animal nouns (cats and dogs) are restricted to a single syntactic position. In particular, they never appear in the direct object position during training. During testing, we examine whether the network can generalize the use of those words to novel syntactic positions, or in other words, whether the network is able to predict occurrences of those restricted words in the direct object position.

2.3.2 Training and Test Corpora

All sentences in the corpora employ the syntax shown in Figure 2.3. However, in the training corpus, two human nouns and two animal nouns are restricted to appear only as grammatical subjects. The training corpus consists of 1,000 randomly sampled sentences. The number of training epochs is 30,000, and the network's behavior is measured after every 1,000 epochs.

Also, in the training corpus, each word is assigned a probability of occurrence, as shown in Figure 2.4. Font sizes indicate different probabilities of occurrence. In the same group of words (e.g. human-nouns, human-verbs, etc.), the probability assigned to the word with the largest font (e.g. boys in the human-noun group) is five times greater than the word with the smallest font (e.g. girls in the human-noun group). Other words are also
assigned a probability 2, 3, or 4 times the probability of a word with the smallest font (see Figure 2.4).

On the other hand, the test corpora contain all possible sentences that can be generated by the grammar shown in Figure 2.3. There are two separate test corpora, one for examining the network’s capacity for predicting human nouns following human verbs, and the other for examining the capacity for predicting all nouns following all other verbs. The former corpus contains 256 sentences, and the later contains 2,560 sentences.

2.3.3 Algorithms and Architectural Details

As indicated in Figure 2.1, the model used here is a simple recurrent network. Both input and output layers contain 31 nodes. Among the 31 nodes, 30 nodes are assigned to the 30 different words in the corpus, and the remaining one is assigned to the end of sentence marker (i.e. the period). Different sizes of hidden layer (20, 30, 40, and 60 units) have
been employed in different experiments. The context layer is of the same size as the hidden layer.

Recall that links from the hidden layer to the context layer are one-to-one copying links. On the other hand, links from the input layer to the hidden layer, from the hidden layer to the output layer, and from the context layer to the hidden layer, are all fully connected, trainable links.

The weights on all trainable links are initialized with random values from the interval $[-0.1, 0.1]$, and trained through the back-propagation algorithm. The learning rate of back-propagation is set to 0.01. The momentum and the initial bias offset are all set to zero. After activation from the input layer spreads up to the hidden layer, the activation pattern is copied from hidden units to context units on a one-to-one basis, and later fed back to the hidden layer together with the input in the next cycle.

2.3.4 Test Phase and Analysis of Results

The test phase involves feeding each sentence from the test corpora, word by word, to the network. The activation value of each unit in the output layer indicates the network’s prediction for the probability of the corresponding word to succeed the given input word. Recall that, in order to challenge the generalization capacity of the network, during training, two human nouns and two animal nouns are restricted to a single syntactic position (i.e. the subject position). We call those nouns restricted words. To examine whether the network can generalize on the basis of other words in the same syntactic categories, we compared the activation of restricted words to unrestricted words. In particular, during the test phase, the average activations of both restricted and unrestricted words were calculated respectively when different numbers of training epochs were reached. We also compared the results of networks with different sizes (20, 30, 40, and 60 units) of hidden layer. These comparisons are displayed in Figures 2.5 to 2.12. The number of epochs ranges from 1,000 to 30,000 for each experiment.
Chapter 2 Related Works and Experiments

Average activation of restricted and unrestricted nouns following human verbs
(with 20 hidden units)

Figure 2.5: The average activation of restricted and unrestricted words after human verbs, with 20 units in the hidden layer (The number of epochs ranges from 1,000 to 30,000). Notice that the “restricted” bars are barely visible.

Average activation of restricted and unrestricted nouns following all other verbs
(with 20 hidden nodes)

Figure 2.6: The average activation of restricted and unrestricted words after all other verbs, with 20 units in the hidden layer (The number of epochs ranges from 1,000 to 30,000). Notice that the “restricted” bars are barely visible.
Chapter 2 Related Works and Experiments

Average activation of the restricted and unrestricted words following human verbs (with 30 hidden nodes)

Figure 2.7: The average activation of restricted and unrestricted words after human verbs, with 30 units in the hidden layer (The number of epochs ranges from 1,000 to 30,000). Notice that the "restricted" bars are barely visible.

Average activation of restricted and unrestricted nouns following all other verbs (with 30 hidden nodes)

Figure 2.8: The average activation of restricted and unrestricted words after all other verbs, with 30 units in the hidden layer (The number of epochs ranges from 1,000 to 30,000). Notice that the "restricted" bars are barely visible.
Chapter 2 Related Works and Experiments

Average activation of restricted and unrestricted words following human verbs
(with 40 hidden nodes)

![Graph showing activation of restricted and unrestricted words following human verbs.]

Figure 2.9: The average activation of restricted and unrestricted words after human verbs, with 40 units in the hidden layer. (The number of epochs ranges from 1,000 to 30,000). Notice that the "restricted" bars are barely visible.

Average activation of restricted and unrestricted nouns following all other verbs
(with 40 hidden nodes)

![Graph showing activation of restricted and unrestricted nouns following all other verbs.]

Figure 2.10: The average activation of restricted and unrestricted words after all other verbs, with 40 units in the hidden layer. (The number of epochs ranges from 1,000 to 30,000). Notice that the "restricted" bars are barely visible.
Chapter 2 Related Works and Experiments

Average activation of restricted and unrestricted words following human nouns
(with 60 hidden nodes)

Figure 2.11: The average activation of restricted and unrestricted words after human verbs, with 60 units in the hidden layer (The number of epochs ranges from 1,000 to 30,000). Notice that the “restricted” bars are barely visible.

Average activation of restricted and unrestricted words following all other verbs
(with 60 hidden nodes)

Figure 2.12: The average activation of restricted and unrestricted words after all other verbs, with 60 units in the hidden layer (The number of epochs ranges from 1,000 to 30,000). Notice that the “restricted” bars are barely visible.
The results show that after only 1,000 epochs, there is a huge difference between the activation of restricted and unrestricted words. Following further epochs, the network learns slowly and sometimes haphazardly, but its behavior still conforms fairly well to the statistics of training data. At the same time, the activation of restricted nouns is weakening and barely visible. Although there is sometimes a slight increase between 8,000 and 20,000 epochs, contrary to Elman’s results, there is no significant increase in activation for restricted nouns before the training reaches 30,000 epochs. Therefore, the network seems to be only operating with a rote strategy. It does not display any capacity for generalizing restricted words. Elman’s experiments (Elman, 1998) purportedly showed that the network’s capacity for generalization improved after 5,000 epochs. However, there was only one word restricted in his experiments, and there are four in our case. In conclusion, we can find no evidence that a simple recurrent network will ever display a capacity for generalization, when there is a non-trivial number of restricted words in the training corpus. This implies that a simple recurrent network does not have the capacity for strong generalization.

\[5\] Notice that, in Figures 2.6, 2.8, 2.10, and 2.12, there is a slight increase between 8,000 to 20,000 epochs. This resembles Elman’s results very slight. However, the activation values for restricted words are still very insignificant when compared with those for unrestricted words.
Chapter 3

System and Task Overview

The task of the network proposed here is to learn to predict semantic features of the next word, given a prior sequence of words. The network is expected to predict in a way that satisfies the requirement of strong systematicity (Hadley, 1994a), and deals with semantically anomalous sentences.

3.1 System Overview

Figure 3.1 shows the network architecture. The arrows in the figure represent entire sets of links between two layers. Dotted arrows indicate trainable links. The training of the network involves only the portion inside the dotted square. We call this portion the training network. A mechanism for learning information about both general semantic categories and subcategories, by utilizing two sub-networks and using different parameter settings, is placed into the network design to see if this helps the network deal with semantic anomalies.
Figure 3.1: Network architecture. The arrows represent entire sets of links between two layers. Dotted arrows indicate trainable links. The training of the network involves only the portion inside the dotted square (see later for details). On the other hand, the testing of the network also involves the portion on top of the dotted square (see chapter 5 for details).
Chapter 3 System and Task Overview

Figure 3.2 shows the architecture of the training network. The network consists of four layers: an input layer, a first hidden layer (HL1), a second hidden layer (HL2), and an output layer. In short, we put two of Hadley’s Hebbian-competitive networks (Hadley et al., 2001) side by side, sharing the same input layer, and train them independently. Different parameter settings are used in the two sub-networks to make the left sub-network recognize general categories (e.g. the correct usage of English grammar) and the right sub-network recognize sub-categories (e.g. human nouns, animal nouns, or inanimate nouns).

In each of the sub-networks, HL1 contains three distinct areas, which are named (left to right) area A, area B, and area C. Links between the input layer and area A are fully connected. That is, every node in the input layer is connected to every node in area A. Links between area A and area B, and between area B and area C, are untrainable one-to-one copy links and have weights of +1, so that area B and C can store the previous successive contents of area A. Those memory copies may resemble aspects of human short-term memory. The links from HL1 to HL2 and from HL2 to the output layer are all fully connected.
Chapter 3 System and Task Overview

Area A of HL1 is a winner-take-all (WTA) competitive cluster, receiving input from the input layer. HL2 is also a single competitive WTA cluster, receiving input from all three areas of HL1. The links are trained by a competitive algorithm described in (Hadley et al, 2001) and explained later in section 4.2. The links from HL2 to the output layer are trained via a reverse competitive learning algorithm, described later in section 4.2.4. Area A has a post-training role of categorizing the input feature vectors into semantic groups. The role of HL2 is to be a higher order pattern recognizer to categorize the ternary patterns that appear in the three areas of HL1. The output layer receives activation from HL2 and is trained to make semantic predictions of the next word.

During the test phase, another mechanism, which consists of a presentation layer, a competitive cluster with two gating nodes, left-buffer, right-buffer, and vocabulary buffers, is put on top of the training network (see Figure 3.1). These enable the two sub-networks to compete with each other in order to present their predictions up to presentation layer. This competition is based on the coherence of the predictions of each sub-network with the network’s vocabulary. Coherence is a measure of similarity between the predicted vector and the various semantic vectors in the network’s vocabulary. The more the predicted vector resembles the preassigned semantic features of a certain word in the network’s vocabulary, the greater the degree of coherence it has (see section 5.4 for details).

Predictions from both sub-networks will be reinforced according to their degree of coherence, and those with greater coherence will get more reinforcement to win the competition. Since this process involves a coherence-based reinforcement, we call it coherence reinforcement process (see Chapter 5 for details). This is conjectured to be a simulation of the human decision process between two information sources, and with this mechanism, sometimes the predictions in the presentation layer are even better than those from any one of the two networks alone. We will revisit this coherence reinforcement process in later chapters.
3.2 Syntax and Semantic Constraints

The connectionist network presented here is an expansion of Hadley's Hebbian-competitive network (Hadley et al, 2001). It is designed to process words, one at a time, taken from a pool of sentences generated according to a simple syntax displayed in Figure 3.3. It shows that the vocabulary consists of 16 nouns and 15 verbs, plus a relative pronoun "that", two prepositions "with" and "from", and a period ".".

S -> NP V NP .

NP -> N | N RC | N PP

N -> NOUN-HUM | NOUN-ANIM | NOUN-INANIM | NOUN-FOOD

V -> VERB-EAT | VERB-PERC | VERB-TRAN | VERB-STREN | VERB-HIT

NOUN-HUM (human nouns) -> boys | girls | men | women

NOUN-ANIM (animal nouns) -> dogs | cats | birds | mice

NOUN-INANIM (inanimate nouns) -> books | rocks | bats | chairs

NOUN-FOOD (food nouns) -> cookies | sandwiches | chocolate | noodles

VERB-EAT (verbs involving eat) -> eat | bite | chew | swallow

VERB-PERC (verbs involving perceptions) -> smell | see | watch | sniff

VERB-TRAN (transitive verbs) -> see | chase | love | follow

VERB-STREN (verbs involving strength) -> move | break

VERB-HIT (verbs involving hit) -> hit | bump

RC -> that V NP

RC -> that N V

PP -> Prep NP

Prep -> from | with

Figure 3.3: The grammar for generating training and test sentences.
Chapter 3 System and Task Overview

All words have been previously assigned semantic feature vectors (See Appendix A). Less than half of all the possible features are assigned to a given word. The immediate goal is to get the network to make predictions for the semantic features of the next word, given a current word, in order to examine its generalization capacity.

In order to examine the network’s response to semantically anomalous sentences, the sentences in the training corpus are generated according to a set of semantic constraints: all the simple sentences and all clauses of complex sentences must fall into one of the semantic structures defined in the semantic constraints. See Figure 3.4.

NOUN-HUM VERB-EAT NOUN-FOOD
NOUN-HUM VERB-EAT NOUN-FOOD with NOUN-HUMAN
NOUN-HUM VERB-PERCEPT NOUN-INANIM
NOUN-HUM VERB-TRAN NOUN-HUM
NOUN-HUM VERB-TRAN NOUN-HUM with NOUN-HUMAN
NOUN-HUM VERB-TRAN NOUN-ANIM
NOUN-HUM VERB-TRAN NOUN-ANIM with NOUN-HUMAN
NOUN-HUM VERB-STREN NOUN-INANIM
NOUN-HUM VERB-STREN NOUN-INANIM with NOUN-HUMAN
NOUN-HUM VERB-STREN NOUN-INANIM with NOUN-INANIM
NOUN-ANIM VERB-EAT NOUN-FOOD
NOUN-ANIM VERB-EAT NOUN-FOOD with NOUN-ANIM
NOUN-ANIM VERB-TRAN NOUN-ANIM
NOUN-ANIM VERB-TRAN NOUN-ANIM with NOUN-ANIM
NOUN-ANIM VERB-TRAN NOUN-HUM
NOUN-ANIM VERB-TRAN NOUN-HUM with NOUN-ANIM
NOUN-ANIM VERB-STREN NOUN-INANIM
NOUN-ANIM VERB-STREN NOUN-INANIM with NOUN-ANIM
NOUN-ANIM VERB-STREN NOUN-INANIM with NOUN-INANIM
Chapter 3 System and Task Overview

3.3 Input and Output

The input and output of the network proposed here are semantic feature vectors consisting of 60 units. Each word in the vocabulary is encoded using 60 features taking binary values. A unit in the encoding of a word is set to one if the word exhibits the feature, and zero otherwise. Among the 60 features, 23 features are assigned to nouns, 21 are to verbs (see next paragraph), and the remaining 16 features are reserved for the words (1) “that”, (2) “with”, (3) “from”, and (4) the period “.”, which do not have straightforward semantic information. These 16 features are divided equally and assigned to the four words above. They might be viewed as syntactic representations (Hadley et al, 2001), since these four words serve as functional words, which are semantically light and used to signal structure. Also, we assign the verb feature “emotive” additionally to the nouns “girls” and “women” (See Appendix A). They have been assigned this feature for experimental purposes, to see if this overlap of noun features and verb features in semantic representation affects the formation of separate recognizers for noun features and verb features in area A of HL1.
Chapter 3 System and Task Overview

The creation of semantic features here is admittedly incomplete and somewhat arbitrary. However, it has conveyed the general approach we have adopted. That is, if these semantic features do exist in the human language acquisition mechanism, the proposed network is able to provide a possible computational model for dealing with semantic anomalies.

Features assigned to nouns include: *has-weight, has-size, light, inanimate, talks, laughs, two-legs, furry, bites, four-legs, rigid, tools, heavy, nutritious, edible, tasty, large, small, barks, meows, human-made, flat, flexible*. On the other hand, features assigned to verbs include: *action, rapid, involves-animation, involves-eat, involves-mouth, feeling-nice, non-human-object, perceiving, transitive, animal-action, work-on-inanimate, needs-strength, vicious, feeling-bad, inanimate-subject, hurt, emotive, fast, slow, involves-eyes, involves-nose*. The semantic features we employ here are not complete generally. However, as stated in (Hadley et al, 2001), they serve to convey the general approach we have adopted.

3.4 Three parts of the Task

The task of the connectionist network presented here consists of three parts: to examine the degree of systematicity the network is capable of displaying, to test the network’s capacity for making predictions in compliance with the semantic constraints, and to explore the network’s response to semantically anomalous but grammatically correct sentences.

3.4.1 Systematicity

In order to examine the degree of systematicity the network is capable of displaying, during training some nouns are precluded from appearing as grammatical subjects (NS nouns), and some are precluded from appearing as grammatical direct object (NO nouns). After training, we examine whether the network is able to generalize the use of those
restricted nouns to novel syntactic positions, that is, to have the same response as unrestricted nouns.

The predictions from both left and right sub-networks are going to be examined. We expect that the predictions from the right sub-network will be semantically closer to what is actually presented in the test corpus than those from the left sub-network. This is because the right sub-network is trained to recognize sub-categories of the vocabulary and is able to make more specific semantic predictions than the left sub-network. In addition, we would like to examine whether the coherence reinforcement process helps the network have better predictions than those from any of the sub-networks.

### 3.4.2 Semantic Constraints

Humans can often predict the general semantic characteristics of the next word in a sentence according to semantic constraints. The right sub-network is also expected to be able to do this, since it is trained to recognize sub-categories of input words, and hence has a better capacity of discerning the semantic structure of input words in the context. We create a test sentence for each semantic constraint defined in Figure 3.4, and examine whether the network is capable of predicting the range of possible next words in compliance with the semantic constraints for these sentences. The predictions from the left sub-network are also examined for comparison.

### 3.4.3 Semantic Anomalies

The last part of the task is to examine the network’s capacity for dealing with grammatically correct but semantically anomalous sentences. We expect the network to be able to recognize a semantically anomalous sentence, which violates the semantic constraints defined in Figure 3.4, and at the same time make predictions according to the
grammar defined in Figure 3.3. We assume that the network can recognize a grammatical sentence if a period is predicted at the right place or it can make predictions according to the grammar; failing to make substantial semantic predictions suggests a new semantic pattern that is not habitually encountered during training, i.e., a semantic anomaly (see section 5.3 for an example).

Recall that the right sub-network is trained to recognize subcategories of input words, and has a better capacity for discerning the semantic structure of input words in context than the left-subnetwork. On the other hand, the left sub-network is trained to recognize general categories of input words, and has a better capacity for discerning the syntactic structure of input words in context. A semantically anomalous but grammatically correct sentence has a new semantic structure for the right sub-network, since the right sub-network has never seen any semantically anomalous sentences during training. Conversely, a semantically anomalous but grammatically correct sentence does not have a new syntactic structure for the left sub-network. Thus, we expect that when encountering such a sentence, the right sub-network may not be able make substantial predictions, while the left sub-network can still make predictions according to the syntax of the preceding words. Subsequently, with the help of the coherence reinforcement process, the predictions for the left sub-network will be reinforced and presented as the final output of the network.

With regard to the task of exploring the network’s response to semantically anomalous but grammatically correct sentences, we created three sets of sentences, as semantic anomalies may occur in different positions in a sentence. The first two sets concern semantic anomalies in a simple sentence. The first set contains sentences with semantically anomalous combination of the agent and the action, such as “books eat”. The second set contains those with semantically anomalous combinations of the action and the patient, such as “eat rocks”. The third set concerns semantic anomalies occurring

---

6 The grammar and semantic constraints defined here are not meant to be the entire English grammar or thorough semantic constraints. They have been simplified for experimental purposes. However, we believe that they are useful in examining what we expect the network to be capable of exhibiting.
within a relative clause or a prepositional phrase. The response of both sub-networks to these three sets of semantically anomalous sentences are examined respectively.
Chapter 4

Training Phase

A training phase is designed to give the network ample opportunities to acquire the capacity for a given task. During a training phase, input from a training corpus will be fed into the network, and the weights on all trainable links will be adjusted according to a weight modification rule. The network usually has to go through several passes of training to acquire the capacity.

4.1 Training Corpus

The training corpus contains 10,000 sentences, randomly generated in accordance with both the grammar defined in Figure 3.3 and the semantic constraints defined in Figure 3.4. Also, in order to examine the capacity of the network to generalize outside of training data, we have to ensure that the training corpus can give the network ample opportunities to do so. Some constraints of the training corpus generation are:

- Six of the sixteen nouns (namely, books, boys, cookies, sandwiches, chocolate, and noodles) are precluded from appearing as grammatical subjects during training (referred to as NS nouns). Note that cookies, sandwiches, chocolate, and noodles
never appear as a grammatical subject in the corpus because of the semantic constraints.

- Four of the sixteen nouns (namely, men, dogs, cats, and women) are precluded from appearing as grammatical objects during training (referred to as NO nouns).
- Six of the sixteen nouns (namely, girls, mice, birds, bats, chairs, and rocks) are permitted to appear in any grammatically acceptable position (referred to as unrestricted nouns). Rocks will be the only inanimate noun ever to occur in the position of subject during training, and is permitted to occur only in conjunction with the verbs “bump” and “hit”.
- Half of the sentences in the training corpus are simple sentences with the form NOUN VERB NOUN.
- 25% of the sentences in the training corpus contain a single prepositional phrase.
- 25% of the sentences in the training corpus contain one or two relative clauses. Half of those contain only one relative clause and are generated according to the following 2 structures with equal frequency:
  that VERB NOUN, as in “that eat cookies”.
  that NOUN VERB, as in “that boys chase”.

The other half contain two consecutive relative clauses and are generated according to the following two structures with equal frequency:
  that VERB NOUN that VERB NOUN, as in “that chase cats the see mice”.
  that VERB NOUN that NOUN VERB, as in “that chase cats that boys see”.

In addition to the constraints stated above, all the sentences generated have to comply with the semantic constraints.
Chapter 4 Training Phase

4.2 Training Algorithms

4.2.1 Weight Initialization

Prior to the training phase, weights on links entering a node in a competitive cluster (e.g., area A of HL1 and HL2) are assigned random values from the closed interval [0, 1]. Weights are also normalized to ensure that the sum of weights entering a given node is one.

On the other hand, the links entering the output layer are trained via a reverse competitive learning algorithm, described below. The weights on the links from each node in HL2 to the output layer add up to one and are distributed evenly. See Figure 4.1.

![Figure 4.1: The initialization of the weights between HL2 and the output layer.](image)

4.2.2 Winner Selection

Both area A in HL1 and all of HL2 function as competitive clusters and are trained through a competitive learning algorithm proposed by von der Malsburg (1973). Here we are using the winner selection method used in (Hadley, et al, 2001). In the latter, two activation values, received activation and refined activation, are calculated respectively. Received activation for a given node is simply the weighted sum of the activation flowing
from all the input nodes connected to it. Refined activation of a given node is calculated using the following equation:

\[
\text{refined} = c \times \text{received} + \left\{ (1 - c) \times \frac{\text{received}}{\text{Max}} \right\}
\]

where \(\text{Max}\) is the maximum received activation for the given node; \(c\) is a constant that can be tuned experimentally. Nodes in the same cluster compete with each other based on their refined activation. The winning node is set to an activation value of +1, and others are all set to zero. The constant \(c\) here gives the winner selection a bias in favor of either the raw received activation or the ratio of current received activation to the previous maximum. If \(c\) is close to one, the winner selection will favor a winning node to recognize several overlapping input patterns. If \(c\) is close to zero, it will have a tendency to set up a one-to-one mapping between a distinct input pattern and a distinct winning node. In other words, the smaller the constant \(c\), the greater the tendency of large number of groups of input patterns being formed in a competitive cluster.\(^7\)

In our implementation, we used a larger constant \(c\), 1.0 for the left sub-network and 0.6 for the right sub-network the nodes in area A of HL1 and a smaller one, 0.1, for the nodes in HL2. Since in area A of HL1, we hope that input patterns with many overlapping features can be recognized by the same winning node. On the other hand, in HL2, we hope to have a one-to-one mapping between the syntactic patterns discovered in HL1 and the nodes in HL2. Also, for the nodes in area A of HL1, a larger constant \(c\), 1.0, is used in the left sub-network than that in the right sub-network (i.e. 0.6). That is because we expect to train the right sub-network to recognize more categories (i.e. subcategories) in area A of HL1.

### 4.2.3 Weight Modification in a Competitive Cluster

In a competitive cluster, such as area A of HL1 and HL2 in the proposed model, once the winning node has been selected, weights on links leading to the winning node are

---

\(^7\) The adjustment of the constant \(c\) to form different numbers of groups of the input patterns is suggested by Robert F. Hadley.
Chapter 4 Training Phase

modified according to a competitive learning rule proposed by von der Malsburg (1973). Based on this rule, weight modification only happens to winning nodes. The weight modification equation for a winning node is:

\[ \Delta \omega_{ij} = g \frac{c_{ik}}{n_k} - g \omega_{ij} \]

where \( i \) is the index of the input pattern in the input layer below the competitive cluster, ranging from 1 to the size of the input layer; \( j \) is the current winning node in the competitive cluster; \( n_k \) is the number of active nodes in the input pattern; \( c_{ik} \) is equal to 1 if node \( i \) in the input pattern is active, and 0 otherwise; \( g \) is the learning rate, which is one of the parameters that will affect the number of groups of the input patterns that can be formed by a competitive cluster. Usually a smaller learning rate favors a larger number of groups being formed.

4.2.4 Reverse Competitive Learning Algorithm

As for the coherence reinforcement process mentioned earlier, we require the sum of the weights on all the links from each node in HL2 to the output layer to be equal to one, so that the total activation predicted in the output layer will also sum to one (see Figure 4.1). Before training, the weights are distributed evenly on the outgoing links from each node in HL2. The weight modification equation is:

\[ \Delta \omega_{ij} = g \frac{c_{ik}}{n_k} - g \omega_{ij} \]

Where \( i \) is the index of the output layer; \( j \) is the current active node in HL2; \( n_k \) is the number of active nodes in the output layer; \( c_{ik} \) is equal to one if node \( i \) in the output layer is active and 0 otherwise; \( g \) is the learning rate. Notice that the modification equation resembles the one in the original competitive learning algorithm. It is actually a reverse competitive learning algorithm since we have put the input layer and the competitive cluster upside down with respect to the original algorithm.
Chapter 4 Training Phase

The basic idea of this algorithm is that we have to weaken the links connected from the winning node to inactive nodes in the output layer, and strengthen the links to active nodes. Hence, the algorithm takes a small amount of weight, decided by the learning rate, from all links connected to inactive nodes in the output layer, and then redistributes the weight to the links connected to active nodes (see Figure 4.2). That is, we transfer the weights from the links between an inactive and an active node to those between two active nodes.

With this method, we can strengthen the links between two active nodes and weaken those between an active and an inactive node, while keeping the sum of weights from any given node in HL2 to the output layer to be equal to one. This method actually preserves the basic idea of Hebbian learning, which states, "The connections between two neurons might be strengthened if the neurons fire simultaneously" (Hebb, 1949). In other words, Hebbian learning applies a small amount of increment to the links between two active nodes.

One of the advantages of using the reverse competitive learning algorithm over this simple Hebbian increment model, is that the sum of activation presented in the output layer will be restricted to one, instead of being incremented without a limit.

Figure 4.2: Weight modification in the reverse competitive learning algorithm. The process takes a small amount of weights, decided by the learning rate, from the links connected to inactive nodes in the output layer (left), and then redistributes these weights to the links connected to active nodes (right).
Chapter 4 Training Phase

The assumption of this design is that prior to training, every semantic feature will be equally weakly predicted. That is, each semantic feature has an equal probability of appearing in the prediction of the next word. If a node in HL2 never wins during training, the weights on the links from this node to the output layer will never be changed. Thus, every semantic feature in the output layer will still have equally weakly predicted, indicating equal probability of appearing in the prediction of the next word.

4.3 Training Cycle

During the training phase, 50,000 sentences are selected randomly from the training corpus and fed into the network continuously. For a given input sentence, each word is fed into the network one at a time in sequence. In between sentences, a period is fed into the network to make the transition. This approach has been adopted by other cognitive models on similar tasks, such as the simple recurrent network (Elman, 1990) and the Hebbian-competitive network (Hadley et al, 2001), since a period may correspond to an extended silence between sentences in speech.

We assume that there is a stream of input words coming into the network in sequence, like a continuous stream of speech. In a training cycle, the feature vector of the current input word is presented in the input layer. The activation flows from the input layer to area A in HL1, competition occurs, and a winner is selected in area A, indicating the category of the current word. Weight modification then applies to all links between the input layer and the winning node in area A. Later, the activation from area A of HL1 spreads out through two sets of links, one going up to HL2, and the other going through the copy links to the area B. The activation going to area B will trigger a chain reaction to cause the prior activation of area B to spread to area C. Consequently, the prior content of area B is copied into area C, and so is the prior content of area A copied into area B. At the same time, the original activation of all three areas spreads upwards to HL2.
Again, a competition occurs, and a winning node is selected in HL2, and weight modification happens afterward on the links between all three areas in HL1 and HL2.

At this moment, the next input word is coming into the network, and its semantic feature vector is presented in both the output layers and the input layer. The weights on the links between HL2 and the output layer are then adjusted through the \textit{reverse competitive learning algorithm}, described in section 4.2.4. In the meantime, activation from the input layer starts to spread up to area A of HL1, and the next training cycle begins.

The two sub-networks in the proposed network are trained independently, with the same input and output, the same training algorithm, but different parameter settings, i.e. the learning rate and the constant $c$ in the winner selection rule. They then develop different weight configurations. We use a larger learning rate, 0.5, and a larger constant $c$, 1.0, in the area A of the left sub-network, to make it form groups of general categories, and a smaller learning rate, 0.02, and a smaller constant $c$, 0.6, in the area A of the right sub-network, to make it form groups of sub-categories.
Chapter 5 Test Phase

Chapter 5

Test Phase

A test phase is required, following the training phase, to examine what capacities the network has acquired during training. During a test phase, the weights on all links are usually frozen. That is, no learning occurs during this phase. Also, a test corpus that is able to give the network ample opportunities to exhibit its capacity is created. Input from the corpus will be fed into the network, and the corresponding output will be examined to see if the trained network is capable of fulfilling the given task.

5.1 Test Corpus

In order to test the degree of systematicity that the proposed network is capable of exhibiting, another corpus of 10,000 sentences is generated. The same grammar (see Figure 3.3) and semantic constraints (see Figure 3.4) for generating the training corpus are used, but now there are no restrictions on the grammatical positions which each word can occupy, as long as it satisfies the grammatical and semantic constraints.

Regarding the task of examining the capacity of a network to deal with semantically anomalous but grammatically correct sentences, three different test sets are created. The first set contains sentences involving an semantically anomalous combination of the subject and the verb. An example of this kind of sentence is "rocks
Chapter 5 Test Phase

eat cookies”. The second set contains sentences involving an semantically anomalous combination of the action and the patient, such as “boys eat tables”. The last set contains sentences involving a semantic anomaly within a relative clause or a prepositional phrase, such as “rocks that eat cookies hit books”. Table 5.1 shows the three sets of sentences created.

<table>
<thead>
<tr>
<th>Set 1</th>
<th>Set 2</th>
<th>Set 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>boys hit books</td>
<td>boys eat rocks</td>
<td>girls eat rocks with sandwiches</td>
</tr>
<tr>
<td>rocks eat boys</td>
<td>rocks eat boys</td>
<td>rocks eat girls from boys</td>
</tr>
<tr>
<td>cookies eat boys</td>
<td>cookies eat boys</td>
<td>boys that hit books hit books</td>
</tr>
</tbody>
</table>

Table 5.1: Three sets of semantically anomalous sentences involving semantic anomalies in different positions of a sentence.

Notice that a semantic anomaly, specifically defined in the models proposed here, is any sentence that violates the semantic constraints used for generating the training corpus, i.e. the semantic constraints defined in Figure 3.4. For experimental purposes, the semantic constraints contain some simplifying assumptions that admittedly are not always in compliance with English semantics. For example, Noun-Human can be the subject of the verb hit in English semantics, but this is not the case in our semantic constraints. Hence, the sentence “boys hit books” is considered a semantically anomalous sentence in our case. Also, the sentences “rocks eat boys” and “cookies eat boys” both have two semantic anomalies in different positions and hence appear in both set 1 and set 2.
5.2 Test System Overview

During the test phase, another mechanism to orchestrate the interaction between the predictions from both sub-networks is placed on top of the original network (shown in Figure 5.1 inside the dotted box). As shown in Figure 5.1, besides the input layer, HL1, HL2, and the output layers (i.e. the left output layer and the right output layer) in the original design, the test network also contains vocabulary buffers to store the semantic vectors of each word in the network's vocabulary, a left-buffer to store a copy of the left output layer, a right-buffer to store a copy of the right output layer, a presentation layer to store the final semantic predictions, and a competitive cluster of two gating nodes, a left
Chapter 5 Test Phase

gating node and a right gating node, to gate the activation from the output layers to the presentation layer.

During the test phase, the two sub-networks will make predictions respectively and compete with each other to present a result to the presentation layer through the competitive gating-node cluster. Section 5.4 provides details of this competition.

Figure 5.2: The detailed structure on the top of the two output layers.

In the left sub-network, links from the left output layer to the left-buffer (or the right output layer to the right-buffer in the right sub-network) and from the left-buffer to the vocabulary buffer are all one-to-one copy links and have weights of +1. On the other hand, links from the left-buffer to the left gating node, and from the left gating node to the links between the right output layer and the presentation layer, are fully connected. The same applies to the links in the right sub-network. The vocabulary buffers, the left-
Chapter 5 Test Phase

buffer, the right-buffer, and the presentation layer all have the same size as the output layers. Each node in the presentation layer is connected to the corresponding nodes in the output layers of the two sub-networks. For example, the first node of the presentation layer is connected to the first node of the left output layer and the first node of the right output layer, forming a ternary structure (see Figure 5.2).

The left and right gating nodes receive activation from the left-buffer and the right-buffer, respectively. Each outgoing link of a gating node, shown in Figure 5.1 in the gating node cluster, serves as a modifier link to inhibit the activation in the output layer of the opposite sub-network from going up to presentation layer. In other words, the left gating node inhibits the prediction of the right sub-network, and the right gating node inhibits the prediction of the left sub-network.

A gating node inhibits the activation from spreading when it is in a high-level activation status. In the initial state, both gating nodes have the same high-level activation. Consequently, the left-gating node prevents the activation of the right sub-network's output layer from spreading into the presentation layer, and at the same time, the right-gating node blocks the activation of the left sub-network. Hence, in the initial state, no predictions are made in the presentation layer. If the two gating nodes receive different amounts of activation in a later stage, a competition will occur and a winner will be selected in the competitive gating-node cluster. The losing gating node will cease its inhibition, and the winning sub-network will be able to spread its predictions up into the presentation layer.

5.3 Test Cycle in a Sub-network

Recall that in the training network previously mentioned, we employ two sub-networks that are trained with the same input and output, but different parameter settings. During testing, the test cycle resembles the training cycle, except that the weights on all links are frozen, and the next word that follows the current input word is not activated within the
Chapter 5 Test Phase

output layer. In other words, the two sub-networks take the same input, go through the same process as in the training cycle, except that no weight modification occurs, and input words are presented only in the input layer.

In more detail, during a test cycle in a sub-network, each word of a test sentence is presented to the input layer in turn. A winner is then selected in area A of HL1. The activation of this node is set to one, and continues to spread up to both HL2 and area B of HL1. The activation spreading to area B of HL1 then triggers a chain reaction to spread the prior content of area B into area C. At the same time, the original activation of all three areas of HL1 spreads up to HL2. A competitive process occurs again to select a winner in HL2, and the activation of this node is set to one. It then fires up to the output layer and makes semantic predictions of the next word. Each sub-network will go through the same process and make semantic predictions in its respective output layer. After that, the next word is presented in the input layer, and a new test cycle starts.

The two sub-networks in the model proposed here will go through the same process, as described above, to make predictions respectively. Because of the different parameter settings used during training, the two sub-networks have developed different weight configurations, and will respond differently to input sentences. For example, when the current input sentence is a normal sentence, such as “boys eat cookies”, (see Figure 5.3), and the current input word is “cookies”, the category information of the first word, “boys”, and the second word, “eat”, have been respectively stored in area C and B of HL1. The left sub-network recognizes “boys” as a Noun, and “eat” as a Verb, while the right sub-network recognizes “boys” as a Noun-Human and “Eat” as a Verb-Eat.
Chapter 5 Test Phase

Figure 5.3: The predictions when the current input sentence is a normal sentence "boys eat cookies". Notice that when the word "cookies" is presented in the input layer, the network has seen "boys" and "eat". The left sub-network recognizes this sentence as an instance of "Noun Verb Noun", while the right sub-network recognizes it as an instance of "Noun-Hum Verb-Eat Noun-Food". Since these patterns have been seen during training, both sub-networks are able to predict a period after this sentence.

When the word "cookies" comes into the input layer, area A of HL1 in the left sub-network will recognize "cookies" as a Noun, since the left sub-network is trained to recognize general categories. On the other hand, area A of HL1 in the right sub-network will recognize "cookies" as a Noun-Food, since it is trained to recognize sub-categories. A winner in HL2 is then selected in each sub-network, representing a distinct triadic pattern in HL2. Theoretically, these patterns should have been seen during training, so a period will be predicted by both sub-networks. Thus, for normal sentences, both networks should give good predictions.

When the current input is the word "rocks" in a semantically anomalous but grammatically correct sentence, such as "boys eat rocks", (see Figure 5.4), the left sub-network still recognizes "rocks" as a Noun, but the right sub-network recognizes it as a Noun-Inanim. Thus, in the right sub-network, an entirely new triadic pattern is formed in HL1, and a new winner will consequently be picked in HL2. Since this winner has never
won during training, the weights on the links from the winner in HL2 to the output layer have never been adjusted. Recall that prior to the training, each link from HL2 to the output layer is assigned an equal fractional weight. This fractional weight, if not adjusted during training, will later remain to generate only fractional prediction in the output layer. Hence, when encountering a semantically anomalous but grammatically correct sentence, the right sub-network only generates an equal fractional prediction for every feature. These unsubstantial predictions indicate a semantic anomaly. On the other hand, the left sub-network still predicts a period, indicating its capacity for recognizing a grammatical sentence.

Figure 5.4: The predictions when the current sentence is the semantically anomalous sentence, "boys eat rocks". Notice that when the word "rocks" is presented in the input layer, the network has seen "boys" and "eat". The left sub-network recognizes this sentence as an instance of "Noun Verb Noun", while the right sub-network recognizes it as an instance of "Noun-Hum Verb-Eat Noun-Inanim". Since the pattern in the right sub-network has never been seen during training, the right sub-network cannot make any substantial predictions, indicating a semantic anomaly, while the left sub-network will still predict a period, indicating that the given sentence is a grammatical sentence.
5.4 Coherence Reinforcement

When predictions from both sub-networks are activated in the lowest output layers, the two sub-networks will compete with each other. The one whose predictions have greater coherence with the network’s vocabulary will win the competition and present its result to the presentation layer. As explained previously, coherence is a measure of similarity between the predicted vector and the various semantic vectors in the network’s vocabulary. The more the predicted vector resembles the preassigned semantic features of a certain word in the network’s vocabulary, the greater the degree of coherence it has.

In other words, if the predicted vector covers a broad range of semantic features, and it is hard to tell which word the vector is predicting, then that predicted vector will have less coherence with the network’s vocabulary. We boost the activation level of the predicted vectors in the two output layers according to their degree of coherence. The one with greater coherence will be activated in the presentation layer. This process is called coherence reinforcement process, and is explained in detail below.

After predictions are activated in the output layers, the content of each output layer is spread up and copied into the corresponding buffer through the copy links. That is, the content of the left output layer is copied into the left-buffer, and the content of the right output layer is copied into the right-buffer. Each word in the network’s vocabulary will be presented in the vocabulary buffer in turn. The coherence reinforcement process then occurs between the vocabulary buffer and the left-buffer, by comparing each vocabulary word in the vocabulary layer with the predicted semantic vector in the left-buffer. The same applies to the right-buffer in the right sub-network.

The links between the vocabulary buffer and the left-buffer or the right-buffer are one-to-one copy links connecting corresponding nodes. For each pair of nodes between the left-buffer and the vocabulary buffer, or the right-buffer and the vocabulary buffer, if either member of the pair has a value below a predetermined reinforcement threshold, a boost of activation will not occur. However, if both of them have values above the threshold, a boost of activation, which is proportional to the square of the activation value
of the node in the left-buffer or the right-buffer, will be added to the node in the left-buffer or the right-buffer. Each word in the vocabulary will be activated in the vocabulary buffer in turn and go through the same coherence reinforcement process\(^8\). After reinforcement is complete, the sub-network whose predicted vector has greater coherence with the network’s vocabulary will be the one that has greater activation in total in its output layer buffer.

Recall that prior to training, every outgoing link from a node in HL2 to the output layer is given an equal fractional weight (section 4.2.1). This equal fractional weight, if not incremented during training, will later not be able to generate activation above the reinforcement threshold in the left-buffer or the right-buffer. Hence, this fractional prediction will not be reinforced. In other words, if any node that has never been selected as a winning node during training is later selected as the winner in HL2, none of the features in the subsequently predicted semantic vector will have activation greater than the reinforcement threshold. Hence, none of the predictions for these features will be reinforced. So, if the predictions from the other sub-network have gained some reinforcement, they will be eventually activated in the presentation layer.

The competitive gating-node cluster, which consists of a left gating node and a right gating node, manages predictions that will eventually be activated in the presentation layer between the two sub-networks. The left gating node receives activation from all nodes in the left output layer, and the right gating node receives activation from all nodes in the right output layer. In the initial state, the two gating nodes have the same high-level activation and inhibit the predictions from being activated in the presentation layer. More specifically, the left gating node inhibits the predictions of the right sub-network, and the right gating node inhibits those of the left sub-network. Thus, no activation is presented to the presentation layer at this time.

---

\(^8\) If there is indeed a process in the brain similar to the coherence reinforcement process proposed here, we would expect this process to occur in parallel. That is, each word occupies a buffer and each buffer has one-to-one copy links to both left-buffer and right-buffer.
Chapter 5 Test Phase

Since the two sub-networks have been trained with different parameter settings, the predictions in the output layers are also different. After the coherence reinforcement process is performed on the left-buffer and the right-buffer, the two gating nodes will also receive different activation values. Hence, the competition starts in the competitive gating-node cluster, and the gating node that initially received less activation will cease its inhibition. If the left gating node is the one receiving less activation, it will cease the inhibition, and the predictions in the right output layer will be activated in the presentation layer, and vice versa.

Notice that during the coherence reinforcement process, only the activation in the left-buffer and the right-buffer is reinforced. The original activation values in the output layers are still intact. Therefore, it is the original activation in the output layer of the winning network that is spread up to the presentation layer.

The two gating nodes also prevent activation from spreading up to the presentation layer during the training phase. Recall that, during training, the semantic vector of the next word is presented in both output layers. As described above, the content of the output layers is spread up and copied into the left-buffer and the right-buffer respectively. Since the left-buffer and right-buffer have the same content, they will receive the same amount of reinforcement during the coherence reinforcement process. Consequently, both gating nodes will receive the same activation, stay in the high activation level, and inhibit the activation in the output layers from spreading up to the presentation layer. Thus, during training, the nodes in the presentation layer will not be activated at any time.

5.5 Evaluation

What the network predicts is semantic feature vectors. A good measure of how close two vectors are is the cosine value of the angle between the vectors. The greater the cosine value is, the closer the two vectors are. The maximum possible cosine value is 1.0, and
Chapter 5 Test Phase

d this means that the two vectors are identical. Hence, we evaluate the predictions by calculating the cosine value of the angle between the predicted semantic vector and the semantic vector of each word in the vocabulary. The greater the cosine value is, the more strongly the given word is predicted.

With regard to the issue of systematicity, we examine the cosine value between the average of the vectors predicted following the restricted nouns, and the average of the vectors actually presented following the same restricted nouns in the test corpus. Recall that the training and the test corpora are generated with the same distribution of differing sentence types (e.g., simple sentences or embedded sentences) and with the same grammar and constraints, except that some words are precluded from appearing in certain grammatical positions in the training corpus, i.e. the restricted nouns. If the network does exhibit systematicity, the average of the vectors predicted to follow the restricted nouns should be very close to the average vector of those actually presented after those nouns in the test corpus. The cosine values for the unrestricted nouns and for all nouns are also calculated for comparison.

During the test phase, the two sub-networks make predictions separately. For normal sentences, we would expect that the result of the right sub-network to be better than the left one, because predictions from the right sub-network are supposed to follow the semantic constraints implied in the training corpus. If good subgroups can be formed in area A of HL1, the predictions will be better than for the left sub-network.

For grammatically correct but semantically anomalous sentences, we would expect that the right sub-network will not be able to make substantial predictions. This indicates that the right sub-network is capable of discerning a semantic anomaly. On the other hand, the left one can still predict a period at the right place, or make predictions according to the syntactic structure of the preceding sequence of words. This shows that the left sub-network is capable of recognizing a grammatical sentence (see the example in section 5.3). With the coherence reinforcement process we introduced previously, we would also expect that the better predictions between the two sub-networks will naturally be activated in the presentation layer.
Chapter 6 Experimental Results

In this chapter, we present the experimental results for three parts of the task, as explained in section 3.4. Briefly stated, the three parts concern (1) the degree of systematicity the network is capable of exhibiting, (2) the sensitivity of the network to semantic constraints for normal sentences, and (3) the response of the network when encountering grammatically correct but semantically anomalous sentences. Experimental results regarding the three parts are presented and discussed respectively in sections 6.1 to 6.3.

We have experimented on the network with different parameter settings. The results suggested that using parameters within a certain range of values would give similar results. Here we present the results of a parameter setting that yields good results.

During the training phase, 50,000 sentences were randomly selected from the training corpus described in the previous sections and presented to the network, with restrictions over the syntactic positions of some words. The parameter settings are shown in Table 6.1, below.
Chapter 6 Experimental Results

<table>
<thead>
<tr>
<th></th>
<th>Left Sub-network</th>
<th>Right Sub-network</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Learning rate</td>
<td>Constant c</td>
</tr>
<tr>
<td>HL1</td>
<td>0.5</td>
<td>1.0</td>
</tr>
<tr>
<td>HL2</td>
<td>0.001</td>
<td>0.1</td>
</tr>
<tr>
<td>Output layer</td>
<td>0.05</td>
<td>N.A.</td>
</tr>
</tbody>
</table>

Table 6.1: The parameter settings.

We stated in section 4.2 that a smaller learning rate and a smaller constant c results in a larger number of categories being developed by the competitive learning algorithm. Hence, a smaller learning rate and constant c were used in the right sub-network than in the left sub-network, so that area A of the left sub-network was trained to recognize general categories while area A of the right one was trained to recognize sub-categories. Also, during the test phase, the reinforcement threshold we used was 0.05.

6.1 Systematicity

In order to examine the network's capacity for generalizing the use of a word to a novel syntactic position, we divided all nouns in the vocabulary list into 3 groups: unrestricted nouns, NO nouns, which are precluded from appearing in the object position during training, and NS nouns, which are precluded from appearing in the subject position during training.

The way we evaluate the results regarding the issue of systematicity is, for each group of nouns, calculate the cosine value between the average of the vectors predicted to follow the nouns in that group, and the average of the vectors actually presented after the same nouns in the test corpus (Hadley at al, 2001). Notice that, according to the grammar defined in Figure 2.3, nouns can appear in different syntactic positions, such as a subject
or an object, or within embedded sentences or prepositional phrases. Also, the possible words following a noun in different syntactic positions are different. However, since the training corpus and the test corpus are generated according to the same grammar and semantic constraints, with the same distribution of sentences of different structures, the average of the predicted vectors after training and those actually presented in the test corpus should be very close to each other for unrestricted nouns. If the network does have the capacity of exhibiting strong systematicity, the same should also hold true for restricted nouns.

The results of five different training sessions are shown in Table 6.2. The table shows cosine values obtained following different groups of nouns, i.e. all nouns, NS nouns, NO nouns, and unrestricted nouns. For each session in a sub-network, the percentage of winning times is also shown in the last column of the table. The percentage of winning times is equal to the number of times for which the predictions of the given sub-network are activated in the presentation layer, over the total number of times the sub-network predicts during the test phase.
### Chapter 6 Experimental Results

**Table 6.2:** The cosine values obtained for different groups of nouns in the left sub-network, right-sub-network, and presentation layer.

<table>
<thead>
<tr>
<th>Session</th>
<th>All (cosines)</th>
<th>NS (cosines)</th>
<th>NO (cosines)</th>
<th>Unrestricted (cosines)</th>
<th>Percentage of winning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Left Sub Network</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0.985242</td>
<td>0.980224</td>
<td>0.957048</td>
<td>0.974358</td>
<td>31.03%</td>
</tr>
<tr>
<td>2</td>
<td>0.973371</td>
<td>0.978231</td>
<td>0.92891</td>
<td>0.968372</td>
<td>38.39%</td>
</tr>
<tr>
<td>3</td>
<td>0.977763</td>
<td>0.976381</td>
<td>0.966672</td>
<td>0.95902</td>
<td>37.47%</td>
</tr>
<tr>
<td>4</td>
<td>0.989005</td>
<td>0.980031</td>
<td>0.965502</td>
<td>0.97611</td>
<td>25.28%</td>
</tr>
<tr>
<td>5</td>
<td>0.974209</td>
<td>0.920609</td>
<td>0.947157</td>
<td>0.973542</td>
<td>29.83%</td>
</tr>
<tr>
<td>Min</td>
<td>0.973371</td>
<td>0.920609</td>
<td>0.92891</td>
<td>0.95902</td>
<td>25.28%</td>
</tr>
<tr>
<td>Ave</td>
<td>0.979918</td>
<td>0.9670952</td>
<td>0.9530578</td>
<td>0.9702804</td>
<td>32.40%</td>
</tr>
<tr>
<td>Max</td>
<td>0.989005</td>
<td>0.980224</td>
<td>0.966672</td>
<td>0.97611</td>
<td>38.39%</td>
</tr>
<tr>
<td>Std.dev.</td>
<td>0.006910471</td>
<td>0.02603315</td>
<td>0.01560498</td>
<td>0.00692257</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Session</th>
<th>All (cosines)</th>
<th>NS (cosines)</th>
<th>NO (cosines)</th>
<th>Unrestricted (cosines)</th>
<th>Percentage of winning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Right Sub Network</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0.984908</td>
<td>0.984033</td>
<td>0.988902</td>
<td>0.983159</td>
<td>68.97%</td>
</tr>
<tr>
<td>2</td>
<td>0.97833</td>
<td>0.986272</td>
<td>0.979529</td>
<td>0.971736</td>
<td>61.61%</td>
</tr>
<tr>
<td>3</td>
<td>0.986398</td>
<td>0.981267</td>
<td>0.988609</td>
<td>0.975674</td>
<td>62.53%</td>
</tr>
<tr>
<td>4</td>
<td>0.982285</td>
<td>0.984839</td>
<td>0.983903</td>
<td>0.97873</td>
<td>74.72%</td>
</tr>
<tr>
<td>5</td>
<td>0.976258</td>
<td>0.980796</td>
<td>0.975719</td>
<td>0.974518</td>
<td>70.17%</td>
</tr>
<tr>
<td>Min</td>
<td>0.976258</td>
<td>0.980796</td>
<td>0.975719</td>
<td>0.971736</td>
<td>61.61%</td>
</tr>
<tr>
<td>Ave</td>
<td>0.9816358</td>
<td>0.9834414</td>
<td>0.9833324</td>
<td>0.9767634</td>
<td>67.60%</td>
</tr>
<tr>
<td>Max</td>
<td>0.986398</td>
<td>0.986272</td>
<td>0.988902</td>
<td>0.983159</td>
<td>74.72%</td>
</tr>
<tr>
<td>Std.dev.</td>
<td>0.004291164</td>
<td>0.00234743</td>
<td>0.00573625</td>
<td>0.00436677</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Session</th>
<th>All (cosines)</th>
<th>NS (cosines)</th>
<th>NO (cosines)</th>
<th>Unrestricted (cosines)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Presentation Layer</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0.988986</td>
<td>0.988649</td>
<td>0.987688</td>
<td>0.985855</td>
</tr>
<tr>
<td>2</td>
<td>0.983709</td>
<td>0.984972</td>
<td>0.982056</td>
<td>0.983011</td>
</tr>
<tr>
<td>3</td>
<td>0.987747</td>
<td>0.987172</td>
<td>0.98391</td>
<td>0.987299</td>
</tr>
<tr>
<td>4</td>
<td>0.990253</td>
<td>0.990627</td>
<td>0.975212</td>
<td>0.990231</td>
</tr>
<tr>
<td>5</td>
<td>0.982009</td>
<td>0.984772</td>
<td>0.979751</td>
<td>0.981203</td>
</tr>
<tr>
<td>Min</td>
<td>0.982009</td>
<td>0.984772</td>
<td>0.975212</td>
<td>0.981203</td>
</tr>
<tr>
<td>Ave</td>
<td>0.9865408</td>
<td>0.9872384</td>
<td>0.9817234</td>
<td>0.9855198</td>
</tr>
<tr>
<td>Max</td>
<td>0.990253</td>
<td>0.990627</td>
<td>0.987688</td>
<td>0.990231</td>
</tr>
<tr>
<td>Std.dev.</td>
<td>0.003527416</td>
<td>0.00248478</td>
<td>0.00465717</td>
<td>0.00354972</td>
</tr>
</tbody>
</table>
Chapter 6 Experimental Results

As can be seen, all the cosines are above 0.9. Most are above 0.96 in the left sub-network, most are above 0.97 in the right sub-network, and most are above 0.98 in the presentation layer. This shows that the average of the predicted vectors and the average of the vectors that are actually presented during testing are close to each other, for both restricted and unrestricted nouns. The test corpus contains both simple sentences and embedded sentences, and all nouns are eligible to appear in embedded clauses during testing (see section 5.1). Hence, the network does exhibit strong systematicity, since it generalizes the use of the restricted nouns to novel syntactic positions, in both simple sentences and embedded sentences.

Theoretically, the results of the right sub-network should be better than the left sub-network, because predictions from the right sub-network are supposed to comply with the semantic constraints implied in the training corpus. If a good grouping, which can distinguish different subcategories in the semantic constrains defined in Figure 3.4, is formed in area A of HL1 in the right sub-network, the predictions of the right sub-network should be better than those of the left sub-network. As revealed in Table 6.2, the results support this argument, since the average cosine values obtained in the right sub-networks are all larger than in the left sub-network.

Notice that sometimes the cosine values for unrestricted nouns are slightly lower than for both NS and NO nouns. On closer examination, we found that in these cases, the difference of angles between the two cosines is within only one to three degrees. This shows that the performance on systematicity of NS nouns, NO nouns, and unrestricted nouns are very close to one another. This slight difference is due to the distribution of nouns in the training corpus, since the corpus has been randomly generated.

As regards the predictions in the presentation layer, in most cases the cosine values are greater than those obtained solely from any one of the sub-networks. If we examine all cosine values obtained in 5 sessions from the left sub-network, the right sub-network, and presentation layer, we can see that in 16 out of 20 times (80%) the cosine value from the presentation layer is higher than either the left or right sub-networks. It is
Chapter 6 Experimental Results

clear that the coherence reinforcement process does help the network get better predictions in the presentation layer.

On closer examination, we see that during the competition after the coherence reinforcement process, the right sub-network wins around two-thirds of the time on average. One-third of the time the left sub-network wins. Amongst the latter wins, most are predictions following a sequence of words that humans tend to make, requiring syntactic information. For example, humans tend to predict a noun, instead of a human noun or an animal noun, following a period (i.e. beginning of a sentence), and the left sub-network wins in this case. For some nouns, such as those in *Noun-Human*, that can be followed by almost every verb, the two sub-networks make similar predictions, and sometimes the left sub-network wins.

For example, the left sub-network sometimes wins for predictions following “*Noun-Human*” or “*Noun-Human that*”. Notice that the word following “*Noun-Human that*” can be a noun to introduce a new relative clause, and this noun can be an agent of a human patient (e.g., in “bats hit boys”, “bats” is the agent of the human patient “boys”). Since in the semantic constraints defined in Figure 3.4, “*Noun-Human*” can be a patient of all other nouns, *all* nouns are eligible to appear after “*Noun-Human that*”. Thus, both sub-networks make similar predictions.

We believe, although it has not been confirmed through sufficient psychological experiments, that humans would also tend to predict a noun, instead of a specific sub category of nouns, following “*Noun-Human that*”, since almost all nouns are eligible to appear in this position. It is probable that the coherence reinforcement process we introduce here does help the network get better predictions and provide a possible computational model to simulate human behavior, on the task of predicting the range of possible next words in a sentence.
6.2  Semantic Predictions

The training corpus was created according to the simple semantic constraints defined in Figure 3.4. On the task of predicting the type of the next word in a sentence, humans do exhibit knowledge of semantic constraints and make the predictions accordingly. For the model proposed here, we experimented with sentences with different semantic structures in order to examine whether the network possesses "knowledge" of semantic constraints. For example, the word following the verb "eat" should be an edible noun.

Recall that in the network proposed here, the left sub-network is trained to extract information about more general categories, and the right sub-network is trained to extract information about sub-categories. Hence, the left sub-network has a better capacity for recognizing the coarse-grained syntactic structure of a sentence and making predictions accordingly. On the other hand, the right sub-network has a better capacity for recognizing the fine-grained syntactic-semantic structure of a sentence, dealing with semantic constraints implied in the training corpus, and discerning a sentence with semantic anomalies (see section 5.3 for details).

6.2.1  Simple Sentences

The simple sentences under examination here have the structure Noun Verb Noun. In this section, we are going to examine the predictions following a sequence of input words with the forms Noun (in the subject position), Noun Verb, and Noun Verb Noun in both left and right sub-networks.

Figure 6.1 shows predictions following a noun appearing in the subject position of a sentence in the left sub-network. The x-axis displays the words in the network's vocabulary; the y-axis displays the cosine values of the angle between the average predicted vector following a given input and the vector of each word in the vocabulary.
Chapter 6 Experimental Results

Figure 6.1: Predictions following a noun appearing in the subject position of a sentence for the left sub-network.

As can be seen, all verbs are strongly predicted, and the word that also receives significant prediction. During the test phase, the same winner is selected in area A of HL1 for all the input vectors of nouns. Consequently, all nouns are categorized into the same category, and the same predictions are generated in the output layer. This explains why the left sub-network possesses the capacity of recognizing the coarse-grained syntactic structure of a sentence. Some weak predictions for the word "girls" and "women" can be observed. This is because these two words share the common feature emotive with the verbs bite and love. This semantic feature has been assigned for experimental purposes.

Figure 6.2 shows predictions following a sequence of words having the form Noun Verb for the left sub-network. As expected, all the nouns are strongly predicted. The predictions also suggest that the left sub-network is capable of generalizing the use of words to novel syntactic positions within simple sentences, since all restricted nouns are predicted as strongly as unrestricted nouns. The slightly weaker predictions for the
Chapter 6 Experimental Results

nouns in *Noun-Food*, such as *cookies*, may be due to their lower frequency of appearance during the training phase. The very weak predictions appear for *bite* and *love* are due to the sharing of the common feature *emotive* with the nouns *girls* and *women*.

![Figure 6.2: Predictions following a sequence of words having the form N V for the left sub-network.](image)

Figure 6.2: Predictions following a sequence of words having the form N V for the left sub-network. It is clear that the period has the strongest prediction. This may be because half of the training sentences are simple sentences. The relative pronoun *that* is also strongly predicted. The prepositions *with* and *from* are also moderately predicted.

Figure 6.3 shows predictions following a sequence of words having the form *Noun Verb Noun* for the left sub-network. It is clear that the period has the strongest prediction. This may be because half of the training sentences are simple sentences. The relative pronoun *that* is also strongly predicted. The prepositions *with* and *from* are also moderately predicted.
Chapter 6 Experimental Results

As revealed in Figures 6.1 to 6.3, the left sub-network is able to make predictions according to the syntactic structure of a sentence for simple sentences generated by the given grammar. This suggests that the left sub-network possesses the capacity of discerning the coarse-grained syntactic structure of a sentence and making predictions accordingly. Also, the left sub-network exhibits systematicity since all restricted nouns are predicted as strongly as unrestricted nouns in a position where a noun is expected.
Chapter 6 Experimental Results

In contrast to the left sub-network, the right sub-network is trained to learn the sub-categories of the vocabulary and deal with semantic constraints. It is intended to have a better capacity than the left sub-network for making predictions in compliance with the semantic constraints implied in the training corpus. For example, predictions following a sentence with the pattern *Noun-Human Verb-Eat*, such as *boys eat*, should be the features of food.

Figure 6.4 shows predictions following a *Noun-Human* in the subject position of a sentence for the right sub-network. A similar distribution of predictions, as shown in Figure 6.5, can also be found when the current input is a *Noun-Anim*, since *Noun-Anim* appears in almost exactly the same context as *Noun-Human*, given the semantic constraints. Both figures show that all verbs are strongly predicted, and the relative pronoun *that* is also predicted. The words *hit* and *bump*, however, have weaker predictions than all the other verbs. The predictions comply with the semantic constraints defined in Figure 3.4, since according to the constraints, these two words are not eligible to appear after a *Noun-Human*. There is a strong probability that the predictions for *hit* and *bump* are due to the sharing of some common features with the words *move* and *break*.

---

9 For experimental purposes, the semantic constraints used in generating the corpora contain some simplifying assumptions that are not in compliance with English semantics. For example, *Noun-Human* can be the subject of the verb *hit* in English, but that is not the case in our semantic constraints.
Chapter 6 Experimental Results

Figure 6.4: Predictions following a Noun Human in the subject position of a sentence for the right sub-network.

Figure 6.5: Predictions following a Noun-Anim in the subject position of a sentence for the right sub-network.
Chapter 6 Experimental Results

Figure 6.6 below shows predictions following a Noun-Inanim in the subject position of a sentence for the right sub-network. As revealed, the words *bump* and *hit* are strongly predicted. The semantic constraints state that when a Noun-Inanim appears in the subject position of a sentence, the next word should be a Verb-Hit. Hence, the predictions comply with the semantic constraints. It is very likely that the weaker predictions for other verbs are due to the sharing of some common features with *bump* and *hit*.

![Figure 6.6: Predictions following a Noun-Inanim in the subject position of a sentence for the right sub-network.](image)

Figure 6.6: Predictions following a Noun-Inanim in the subject position of a sentence for the right sub-network.

Figure 6.7 shows predictions following a sequence of words having the form Noun-Human Verb-Eat for the right sub-network. As can be seen, words belonging to Noun-Food are all strongly predicted. The semantic constraints state that the words following this pattern should belong to Noun-Food. An example of such a sentence is “boys eat cookies”. By examining the predictions further, we can observe that only those noun features belonging to Noun-Food are predicted. It is almost certain that the weaker predictions for other nouns are due to the sharing of common features with these nouns belonging to Noun-Food. For example, words belonging to Noun-Inanim, i.e. books,
rocks, bats, and chairs, have higher predictions than the words in Noun-Human or Noun-Anim, since they have more common features with Noun-Food. For example, the feature human-made is shared by chairs, bats, and all words in Noun-Food, and the feature flat is shared by books and coookies.

Figure 6.7: Predictions following a sequence of words having the form Noun-Human Verb-Eat for the right sub-network.

Figure 6.8 shows predictions following a sequence of words having the form Noun-Human Verb-Percept or Noun-Human Verb-Tran for the right sub-network. Examples of such patterns are “boys sniff” and “boys chase”. It is clear that all nouns except those in Noun-Food are strongly predicted. The semantic constraints, which are not intended to correspond closely to English constraints, state that all nouns in the vocabulary are eligible to appear after a Verb-Percept and a Verb-Tran, except those words in Noun-Food. The weaker predictions for the nouns in Noun-Food are due to the sharing of many common features with other nouns. It is clear that systematicity arises from this sharing of common features among words. For example, a sentence with the form Noun-Human Verb-Precept Noun-Food such as “boys sniff cookies” does make
Chapter 6 Experimental Results

sense, although it never appears during training. The weak predictions suggest that there is chance for the predicted word to appear in a reasonable sentence with the preceding words. This also reveals the capacity of the right sub-network to exhibit systematicity and predict according to the semantic constraints at the same time.

Figure 6.8: Predictions following a sequence of words having the form *Noun-Human Verb-Percept* or *Noun-Human Verb-Tran* for the right sub-network.

Figure 6.9 below displays predictions following a sequence of words having the form *Noun-Human Verb-Stren* for the right sub-network. An example of such a pattern is “boys break”. It shows that all words belonging to *Noun-Inanim* are strongly predicted, and further observation shows that only those noun features belonging to *Noun-Inanim* are predicted. This complies with the semantic constraints, which state that the word completing this pattern should belong to *Noun-Inanim*, such as *tables* and *bats*. 
Chapter 6 Experimental Results

Figure 6.9: Predictions following a sequence of words having the form Noun-Human Verb-Stren for the right sub-network.

So far, we have seen predictions following a sequence of words having the form Noun-Human Verb for the right sub-network. A similar distribution of predictions can also be found when the current input pattern is Noun-Anim Verb. This may be because that Noun-Anim appears in almost exactly the same context as Noun-Human in the semantic constraints and the training corpus.

Figure 6.10 shows predictions following a sequence of words having the form Noun-Inanim Verb-Hit for the right sub-network. As revealed, all nouns except those in Noun-Food are strongly predicted. The semantic constraints state that all words belonging to Noun-Human, Noun-Anim, or Noun-Inanim are eligible to appear after this pattern. Hence, the predictions comply with the constraints.
Chapter 6 Experimental Results

Figure 6.10: Predictions following a sequence of words having the form *Noun-Inanim Verb-Hit* for the right sub-network.

Figures 6.11 to 6.15 show predictions following the word pattern *Noun-Human Verb Noun* for the right sub-network. Predictions following the form *Noun-Human Verb-Eat Noun-Food* are provided in Figure 6.11. It is clear that the period is strongly predicted. Weak predictions for the word *with* and *that* can also be observed. These predictions comply with the semantic constraints.
Chapter 6 Experimental Results

Figure 6.11: Predictions following a sequence of words with a form Noun-Human Verb-Eat Noun-Food for the right sub-network.

The very weak prediction for the relative pronoun that may be due to the infrequent appearance of an object relative pronoun in the training corpus. Since Noun-Food never appears in the subject position in the semantic constraints, a relative pronoun following this pattern can only be an object relative pronoun. This seems to be weakening of the systematicity effect. However, this weaker prediction is due to compliance with the semantic constraints defined in Figure 3.4. If the noun in the object position is the word that is eligible to appear in subject position as well, the prediction for that will be stronger than this case (see Figure 6.12, for example).

Other predictions following the form Noun-Human Verb Noun, as shown in Figures 6.12 to 6.15, are also in compliance with the semantic constraints. Notice that the absence of predictions for from is because, according to the constraints, it never appears following this word pattern.
Chapter 6 Experimental Results

Figure 6.12: Predictions following a sequence of words having the form Noun-Human Verb-Percept Noun-Inanim for the right sub-network.

Figure 6.13: Predictions following a sequence of words having the form Noun-Human Verb-Tran Noun-Human for the right sub-network.
Chapter 6 Experimental Results

Figure 6.14: Predictions following a sequence of words having the form *Noun-Human Verb-Tran Noun-Human* for the right sub-network.

Figure 6.15: Predictions following a sequence of words having the form *Noun-Human Verb-Stren Noun-Inanim* for the right sub-network.
Chapter 6 Experimental Results

A similar distribution of predictions can also be found following a sequence of words having the form *Noun-Anim Verb Noun*. This is because in the semantic constrains, *Noun-Anim* appears in almost exactly the same contexts as *Noun-Human*.

Figure 6.16 shows predictions following a sequence of words having the form *Noun-Inanim Verb-Hit Noun-Human*. As can be seen, the predictions include the word “from”, which only appears following the sentence pattern *Noun-Inanim Verb Noun* in the semantic constraints. The predictions for a sequence of words having the form *Noun-Inanim Verb-Hit Noun-Inanim* or *Noun-Inanim Verb-Hit Noun-Anim* also have a similar distribution as Figure 6.16.

As revealed by Figures 6.4 to 6.16, the right sub-network is able to make predictions in compliance with the semantic constraints implied in the training corpus, and at the same time exhibit systematicity, for simple sentences generated by the given grammar and the semantic constraints. Notice that the predictions are also consistent with
Chapter 6 Experimental Results

the grammar rules. This shows that the right sub-network can discern the syntactic-
semantic structure of preceding words, and deal with semantic constraints implied in the
training corpus.

6.2.2 Embedded Sentences

The embedded sentences under examination here are those containing clauses that begin
with the relative pronoun *that*, including sentences containing the pattern *Noun that* and
*Noun Verb Noun that*. In this section, we examine predictions following word patterns
having the forms *Noun that*, *Noun that Verb*, *Noun that Noun*, *Noun that Verb Noun*, and
*Noun that Noun Verb* in both left and right sub-networks.

Figure 6.17 shows predictions for a sequence of words having the form *Noun that*
for the left sub-network. It is clear that all nouns and verbs are predicted. The syntactic
constraints state that the relative clause following the word *that* can start with either a
noun or a verb. The predictions thus comply with the constraints. This also suggests that
the left sub-network is capable of generalizing the use of words to novel syntactic
positions, within embedded sentences, since NS nouns, which never appear following
*Noun that* during training, are also predicted as strongly as unrestricted nouns.
Chapter 6 Experimental Results

The weaker predictions for the nouns, as opposed to the verbs, are due to the less frequent appearance of the pattern *Noun that Noun* than *Noun that Verb* during the training phase. Among the nouns, those words in *Noun-Food* have even weaker predictions. This is because in the semantic constraints, the words in *Noun-Food* never appear in the subject position of a sentence and hence never appear after the word pattern *Noun Verb Noun that*. It is very probable that the predictions for words in *Noun-Food* are due to the sharing of common features with other nouns. Similarly as in this case, the distribution of predictions among the verbs appears to reflect the frequency of their appearance after the given pattern in the training corpus.

Figure 6.17 below shows predictions following a sequence of words having the form *Noun that* for the left sub-network. All nouns are strongly predicted to follow the appearance of a verb in that context. Figure 6.19 is analogous to Figure 6.18, except that in the pattern of the preceding words, the word after *that* is a noun. This shows that all verbs are strongly predicted to follow the appearance of a noun in that context. It again reveals that the left sub-network is capable of generalizing the use of words to...
Chapter 6 Experimental Results

novel syntactic positions, since both NO nouns and unrestricted nouns are strongly predicted.

Figure 6.18: Predictions following a sequence of words having the form Noun that Verb for the left sub-network.

Figure 6.19: Predictions following a sequence of words having the form Noun that Noun for the left sub-network.
Chapter 6 Experimental Results

Recall that strong systematicity, defined by Hadley (Hadley, 1994a), requires a network to be able to generalize a substantial proportion of words in its vocabulary to novel syntactic positions, within both simple sentences and embedded sentences. In section 6.2.1 we have shown that the left sub-network is capable of this generalization within simple sentences. Combined with the result shown here, we can conclude that the left sub-network is capable of exhibiting strong systematicity.

Figure 6.20 shows predictions following a sequence of words having the form \textit{that Verb Noun} for the left sub-network. As can be seen, the relative pronoun \textit{that} and the period are strongly predicted. Weak predictions for all verbs can also be observed. Stronger predictions for the word \textit{that} and the period are due to the more frequent appearance of them following the pattern \textit{that Verb Noun} than any single verb in the training corpus\(^\text{10}\).

\(^{10}\) Since there are only three areas in HL1 in the current model, the network can only make predictions according the pattern \textit{that Verb Noun}. 

---

Figure 6.20: Predictions following a sequence of words has the form \textit{that Verb Noun} for the left sub-network.
Chapter 6 Experimental Results

Figure 6.21 is analogous to Figure 6.20. The role of the relative pronoun *that* becomes an object relative pronoun, instead of a subject relative pronoun. There is no prediction for the word *that*. This complies with the grammar, which states that the word *that* is not eligible to follow a verb. All verbs also receive stronger predictions due to this reason.

![Figure 6.21: Predictions following a sequence of words having the form *that* Noun Verb for the left sub-network.](image)

As revealed by Figures 6.17 to 6.21, the left sub-network is able to make predictions in compliance with the syntactic category of the preceding sequence of words for embedded sentences, and at the same time exhibit strong systematicity. This also suggests that the left sub-network possesses the capacity of discerning the syntactic structure of a sentence and making predictions accordingly.

Figure 6.22 displays predictions following a sequence of words having the form *Noun-Human that* for the right sub-network. It shows that all nouns and verbs are predicted. Among the nouns, all words in *Noun-Food* have weaker predictions than other nouns; among the verbs, all words in *Verb-Hit* have weaker predictions than other verbs.
Chapter 6 Experimental Results

This complies with the semantic constraints. Predictions following a sequence of words having the form Noun-Anim that have a similar distribution. This is because Noun-Anim appears in almost exactly the same context as Noun-Human in the semantic constraints.

![Graph showing predictions following a sequence of words having the form Noun-Human that for the right sub-network.](image)

Figure 6.22: Predictions following a sequence of words having the form Noun-Human that for the right sub-network.

Figure 6.23 shows predictions following a sequence of words having the form Noun-Inanim that. As revealed, among the verbs the words in Verb-Hit are strongly predicted; among the nouns, those words in Noun-Food have weaker predictions than any other nouns. These indicate that the predictions comply with the semantic constraints.
Figure 6.23: Predictions following a sequence of words having the form Noun-Inanim that for the right sub-network.

Also, the predictions reveal that the right sub-network is capable of generalizing the use of words to novel syntactic positions within embedded sentences, since NS nouns are predicted as strongly as unrestricted nouns.

Figures 6.24 to 6.26 show predictions following a sequence of words having the form Noun-Human that Verb for the right sub-network. Figure 6.24 shows predictions following Noun-Human that Verb-Eat. As predicted, all words belonging to Noun-Food are strongly predicted. Figure 6.25 shows predictions following Noun-Human that Verb-Percept or Noun-Human that Verb-Tran. The results show that all nouns except those in Noun-Food are strongly predicted. This complies with the semantic constraints, since Noun-Food does not appear following these patterns in the constraints. Figure 6.26 shows predictions following Noun-Human that Verb-Stren. As hoped, all words belonging to Noun-Inanim are strongly predicted. Also, a similar distribution of predictions can be found following a sequence of words having the form Noun-Anim that Verb. These predictions comply with the semantic constraints, since only Noun-Inanim is eligible to appear after these patterns.
Chapter 6 Experimental Results

Figure 6.24: Predictions following a sequence of words having the form Noun-Human that Verb-Eat for the right sub-network.

Figure 6.25: Predictions following a sequence of words having the form Noun-Human that Verb-Percept or Noun-Human that Verb-Tran for the right sub-network.
Chapter 6 Experimental Results

Figure 6.26: Predictions following a sequence of words having the form *Noun-Human that Verb-Stren* for the right sub-network.

Also, the predictions reveal that the right sub-network is capable of generalizing the use of words to novel syntactic positions within embedded sentences, since NO nouns are predicted as strongly as unrestricted nouns. And, in section 6.2.1 we have shown that the right sub-network is capable of doing this generalization within simple sentences. Combined with the result shown here, we can conclude that the right sub-network is capable of exhibiting strong systematicity.

Figure 6.27 shows predictions following a sequence of words having the form *that Verb-Eat Noun-Food*. As can be seen, the period and the relative pronoun *that* are strongly predicted. Significant predictions for all verbs can also be observed. A similar distribution can be found for all predictions following a sequence of words having the form *that Verb Noun*. It may be because the number of memory buffers in the current model is three, so that the network is only able to remember a pattern with three words at most. More memory buffers will be helpful for resolving long-distance dependency...
resulting from relative clauses. The same reasoning can be applied to explain the result in Figure 6.28, which shows predictions following a sequence of words having the form *that Noun Verb*.

Figure 6.27: Predictions following a sequence of words having the form *that Verb-Eat Noun-Food* for the right sub-network (The predictions following a sequence of words having the form *that Verb Noun* all have a similar distribution).
Chapter 6 Experimental Results

As revealed by Figures 6.22 to 6.28, in contrast to the left sub-network, the right sub-network makes predictions in compliance with the semantic constraints implied in the training corpus, within embedded sentences generated by the given grammar and the semantic constraints, and at the same time exhibits strong systematicity. Notice that the predictions are also in compliance with the grammar rules. It reveals again that the right sub-network possesses the capacity of discerning the syntactic-semantic structure of the sequence of input words implied in the training corpus.

6.2.3 Sentences with a Prepositional Phrase

Sentences with a prepositional phrase have the form Noun Verb Noun with Noun or Noun Verb Noun from Noun in the training corpus. Figures 6.29 and 6.30 show predictions following a sequence of words having the form Noun Verb Noun with or Noun Verb Noun from for the left sub-network. As can be seen, in the predictions following Noun Verb Noun with, all nouns are strongly predicted. The weaker predictions
Chapter 6 Experimental Results

for the words in *Noun-Food* are due to the lack of appearance of those words following this pattern in the training corpus. The same can explain the predictions following the pattern *Noun Verb Noun from*. For example, in Figure 6.30, the nouns in *Noun-Human*, *Noun-Anim*, and *Noun-Food* are more weakly predicted than *Noun-Inanim*. This is because, according to the semantic constraints, these nouns never appear after a sequence of words having the form *Noun Verb Noun from* in the training corpus.

Figure 6.29: Predictions following a sequence of words having the form *(Noun) Verb Noun with* for the left sub-network.
Chapter 6 Experimental Results

Figure 6.30: Predictions following a sequence of words having the form (Noun) Verb Noun from for the left sub-network.

Figure 6.31 shows predictions following a sequence of words having the form Verb-Eat Noun-Food with for the right sub-network. It is clear that the words in Noun-Human and Noun-Anim are strongly predicted, which complies with the semantic constraints. The weaker predictions for Noun-Inanim and Noun-Food may be due to the sharing of common features with Noun-Human and Noun-Anim. The same reasoning applies to the predictions following a sequence of words having the form Verb-Tran Noun-Human with or Verb-Tran Noun-Anim with, which have a similar distribution as Figure 6.31. Figure 6.32 shows predictions following a sequence of words having the form Verb-Stren Noun-Inanim with. As hoped, the words belonging to Noun-Human, Noun-Anim, and Noun-Inanim are all strongly predicted, which complies with the semantic constraints.
Chapter 6 Experimental Results

Figure 6.31: Predictions following a sequence of words having the form **Verb-Eat Noun-Food** with for the right sub-network (The predictions following a sequence of words having the form **Verb-Tran Noun-Human** with or **Verb-Tran Noun-Anim** with also have a similar distribution).

Figure 6.32: Predictions following a sequence of words having the form **Verb-Stren Noun-Inanim** with for the right sub-network.
Figure 6.33 displays predictions following a sequence of words having the form Verb-Hit Noun-Human with. It shows that these words in Noun-Inanim are strongly predicted. This complies with the semantic constraints, which state that the next word of this pattern can only be a Noun-Inanim. The same applies to the predictions following a sequence of words having the form Verb-Hit Noun-Anim with, Verb-Hit Noun-Inanim with, Verb-Hit Noun-Human from, Verb-Hit Noun-Anim from, or Verb-Hit Noun-Inanim from.

As revealed by Figures 6.29 to 6.33, regarding the sentences with a prepositional phrase in the training corpus, the right sub-network can recognize more fine-grained patterns, such as Noun-Human Verb-Eat Noun-Food with, in contrast to the pattern Noun Verb Noun with, which the left sub-network can recognize. This suggests that the right sub-network possesses a better capacity of discerning the syntactic-semantic structure of
Chapter 6 Experimental Results

the preceding input words, and dealing with the semantic constraints and sub-categories implied in the training corpus.

In conclusion, in this section on the task of examining the response of the two sub-networks to the semantic constraints, the left sub-network possesses a better capacity for discerning general categories and the syntactic structure of the preceding sequence of words and for making predictions accordingly. On the other hand, the right sub-network possesses a better capacity of discerning the syntactic-semantic structure and the sub-categories of the preceding input words, and for dealing with the semantic constraints implied in the training corpus. Also, from the evidence that in the same category (or sub-category for the right sub-network), predictions for restricted nouns are similar to unrestricted nouns, within both simple sentences and embedded sentences, we can conclude that both the left and right sub-network are capable of exhibiting strong systematicity, for the task of predicting the range of possible next words in a sentence.

6.3 Semantic Anomalies

For the task of examining the network’s capacity for dealing with semantically anomalous sentences, three different test sets are created as mentioned in section 5.1. The first set contains sentences involving a semantic anomaly occurring in the middle of the sentences. The second set of sentences involving a semantic anomaly occurring at the end. The third set contains sentences with semantic anomalies involving a prepositional phrase or an embedded sentence (See Table 6.3 below).
Chapter 6 Experimental Results

<table>
<thead>
<tr>
<th>Sentences involving an anomalous combination of agents and actions</th>
<th>Sentences involving an anomalous combination of actions and patients</th>
<th>Sentences involving a semantic anomaly in a prepositional phrase or an embedded sentence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boys hit books.</td>
<td>boys eat rocks.</td>
<td>girls eat rocks with sandwiches.</td>
</tr>
<tr>
<td>rocks eat boys.</td>
<td>rocks eat boys.</td>
<td>rocks eat girls from boys.</td>
</tr>
<tr>
<td>cookies eat boys.</td>
<td>cookies eat boys.</td>
<td>boys that hit books hit books.</td>
</tr>
</tbody>
</table>

Table 6.3: Three sets of semantically anomalous sentences involving semantic anomalies in different positions of a sentence.

The first set of sentences contains three sentences: "boys hit books", "rocks eat boys", and "cookies eat boys". They have an anomalous combination of agents and actions, such as "boys hit", "rocks eat", and "cookies eat". Figure 6.34 shows the activation of each feature in the output layer of the right sub-network following those combinations. The x-axis displays the 60 features in the output layer, and the y-axis displays the activation of each feature. As previously mentioned in section 3.3, the last 16 features are assigned to the 3 words "with", "from", "that", and the period ".". As can be seen, all features have equally weak activation as their initial state. This suggests that a novel semantic-syntactic pattern that has never been seen during training is formed in HL1. Consequently, a node that has never won during training is selected as the winner in HL2, and the links from the winner to the output layer have never been trained. Thus, no weight modification ever happens on these links.

The equally weak activation on each feature reveals the network's inability to make semantic predictions for the given input sentence. On the other hand, the left sub-network is still able to make predictions. Figure 6.35 shows predictions from the left sub-network following the same combinations. As revealed, all nouns are strongly predicted as normal sentences with a pattern "Noun Verb" shown in Figure 6.2. The same
Chapter 6 Experimental Results

predictions can be found in the presentation layer. This indicates that the left sub-network wins the competition after the coherence reinforcement process.

Figure 6.34: The activation of each feature in the output layer of the right sub-network when the current input is a semantically anomalous sentence with a novel semantic pattern.
Chapter 6 Experimental Results

The second set of sentences contains “boys eat rocks”, “rocks eat boys”, and “cookies eat boys”. They have an anomalous combination of the action and the patient, such as “eat rocks” or “eat boys”. As expected, the activation of each feature following those patterns in the right sub-network is the same as shown in Figure 6.34. This reveals the right sub-network’s inability to make predictions for the given input pattern.

Figure 6.36 displays predictions following a sequence of words “boys eat rocks”, “rocks eat boys”, or “cookies eat boys” for the left sub-network. As revealed, it has the same distribution of predictions as the predictions for normal sentences with a pattern “Noun Verb Noun” shown in Figure 6.3. The same predictions can also be found in the presentation layer. This indicates that the left sub-network wins the competition after the coherence reinforcement process.
Chapter 6 Experimental Results

Figure 6.36: The predictions following a sequence of words “boys eat rocks”, “rocks eat boys”, or “cookies eat boys” for the left sub-network. The same predictions can be also found in the presentation layer.

The set of sentences involving a semantic anomaly in a prepositional phrase or an embedded sentence contains “girls eat rocks with sandwiches”, “rocks eat girls from boys”, and “boys that hit books hit books”. Notice that the sentence “boys that hit books hit books” is not anomalous in English. However, according to the semantic constraints defined in Figure 3.4, it is anomalous in our system. We examine the prediction following the pattern “girls eat rocks with”, “rocks eat girls from”, “girls eat rocks with sandwiches”, “rocks eat girls from boys”, “boys that hit “, and “boys that hit books hit books”. The predictions following those patterns in the right sub-network, however, all have the same equally weak activation on every feature as shown in Figure 6.34.

On the other hand, the left sub-network is still able to make predictions for these patterns. Figure 6.37 shows predictions following a sequence of words “girls eat rocks with” for the left sub-network. It is clear that all nouns are strongly predicted. The predictions are identical to those for normal sentences having a pattern “Noun Verb Noun with” shown in Figure 6.29. The predictions in the presentation layer are also identical to
Chapter 6 Experimental Results

those for the left sub-network. This indicates that the left sub-network wins the competition after the coherence reinforcement process.

Figure 6.37: The predictions following a sequence of words "girls eat rocks with" for the left sub-network. The same predictions can be found in the presentation layer.

Figure 6.38: The predictions following a sequence of words "rocks eat girls from" for the left sub-network. The same predictions can be found in the presentation layer.
Chapter 6 Experimental Results

Figure 6.38 shows predictions following a sequence of words “rocks eat girls from” for the left sub-network. Similarly, the predictions are the same as those for normal sentences shown in Figure 6.30. The identical predictions found in the presentation layer indicate that the left sub-network wins the competition after the coherence reinforcement process.

Figure 6.39: The predictions following a sequence of words “girls eat rocks with sandwiches” or “rocks eat girls from boys” for the left sub-network. The same predictions can be found in the presentation layer. As can be seen, only the period is strongly predicted, which complies with the syntax of the training corpus.
Chapter 6 Experimental Results

Figure 6.40 shows predictions following a sequence of words “boys that hit” for the left sub-network. The same predictions can be found in the presentation layer. As revealed, all nouns are strongly predicted as normal sentences with a pattern “Noun that Verb” shown in Figure 6.18 (page 80).

Figure 6.41 displays predictions following a sequence of words “boys that hit books hit books” for the left sub-network. Identical predictions can be found in the presentation layer. This shows that all the nouns are strongly predicted as normal sentences with a pattern “Noun Verb Noun” shown in Figure 6.3 (page 66).
Chapter 6 Experimental Results

Figure 6.41: The predictions following a sequence of words “boys that hit books hit books” for the left sub-network. The same predictions can be found in the presentation layer.

As shown in Figures 6.34 to 6.41, when encountering a semantically anomalous sentence, the right sub-network fails to make significant predictions, whereas the left sub-network still discerns the coarse-grained syntactic structure and makes predictions accordingly. Since the right sub-network avoids making substantial predictions and leaves equally weak activation over every feature in the output layer, the left sub-network will always win the competition after the coherence reinforcement process and present its output to the presentation layer.

Thus, given a semantically anomalous but grammatically correct sentence, the network fails to make semantic predictions, but still recognizes it is a grammatical sentence and makes predictions accordingly. This suggests that some mechanism similar to this network might be found within a larger language acquisition system to explain how humans deal with semantically anomalous sentences.
In the foregoing, we have presented an experiment with a conventional SRN (see chapter 2). The purpose of this experiment is to duplicate Elman’s 1998 experiment (Elman, 1998), by using the same syntax for generating corpora, and the same encoding method (i.e. local representation). However, there were 4 restricted words in our experiment, instead of the solitary one in Elman’s experiment. The results showed that contrary to Elman’s results, there is no evidence that a SRN will ever display a capacity for strong generalization, when there is a substantial percentage of restricted words in the training corpus.

In view of this, we later (Chapter 3-6) presented a connectionist network that can exhibit strong systematicity, defined by Hadley (Hadley, 1994a), on the task of predicting the semantic features of the next word in a sentence. In addition, for normal sentences, the network is able to make predictions according to the semantic constraints implied in the training corpus. For semantically anomalous but grammatically correct sentences, it fails to predict according to the anomalous semantic pattern, but it is able to predict according to their syntactic structures instead.

It is the employment of two identical sub-networks, which are trained via different parameter settings during training to recognize different fineness of grains of categorization, that provides the network with the capacity for dealing with semantic
anomalies. Also, the network employs a coherence reinforcement mechanism on top of the two sub-networks to enable an information switch between them. The one whose predictions have greater coherence with the network's vocabulary will present its result as predictions of the global network (Section 5.4). We have shown that with this mechanism, for normal sentences, most predictions from the global network are closer to what is actually presented in the test corpus, than those from any of the sub-networks alone. For semantically anomalous sentences, the network can still make predictions according to the syntactic structure of preceding words, and ignore unreliable semantic predictions. Thus, the network has provided a possible computational model to simulate human behavior on predicting the range of possible next words in either a normal sentence or a semantically anomalous sentence.

As previously stressed, however, we do not claim that the network proposed here constitutes a real model of the human language acquisition process, or that it satisfies every definition of robust forms of systematicity. Our belief about systematicity in the human language acquisition process is that the sharing of common features, semantic or syntactic, plays an important role for a learning agent to exhibit robust forms of systematicity. We have, in chapter 2, presented an experiment on a conventional simple recurrent network, with local representations. The results suggested that simple recurrent networks display systematicity in a very limited context and for a small fraction of words involved, if local encoding is employed. However, as shown in (Hadley, 2001), a shift from local representation to distributed semantic representation can greatly ease the systematicity problem. Moreover, Hadley's network does exhibit robust forms of systematicity, as well as the network presented here.

Phillips (Phillips, 2000) argues that networks trained through the back-propagation algorithm are not able to fulfill the task of "cross-category" generalization. He gives an example of such generalization:

"John painted the chair red. What color is the chair?" (Target answer: Red)

"John painted the chair blue. What color is the chair?" (Target answer: Blue)
Chapter 7 Discussion and Future Directions

When a network is trained with these two sentences, and then given a test sentence: “John painted the chair mango. What color is the chair?”, it fails to answer “mango” as humans will typically answer. Admittedly, our network will also fail on this task, if “mango” has not been assigned overlapping semantic features with “red” and “blue”. However, as Phillips observes, humans solve this problem by making inferences. Moreover, as indicated in (Hadley, 2001), Phillips’s example presupposes considerable linguistic competence on the part of the human agent in question, and humans do not need to be trained to do this kind of instantaneous inferences.

In addition, to answer “mango” is not the only response that humans would have when they hear this kind of sentence. Humans are also able to recognize the semantic anomaly in “John painted the chair mango”, and at the same time anticipate an end of sentence. They can achieve this task by means of accumulated knowledge of the semantics of “paint” and the syntactic structure of a sentence. Our network demonstrates that this can be achieved through training. These observations suggest that it is probable that humans use different mechanisms to make inferences and to understand the semantic and syntactic structures of a sentence. Nevertheless, we will not pursue this issue here.

The issue in question here is the interaction between semantic and syntactic information in the human language acquisition process. We believe that making predictions for the next word in a sentence not only requires syntactic information, but also semantic information. Most previous works on the issue of systematicity have focused on the syntactic category that the network predicts (Christiansen and Chater, 1994; Elman, 1998). The authors usually trained the networks, with certain parameter settings, to be sensitive to only the syntactic structures of input patterns. Human behavior on predicting the range of possible next words in a sentence, however, is not only related to the syntactic structures of input patterns.

It is probable that humans switch back and forth between semantic and syntactic information to make good predictions. For example, when making predictions after the verb “eat”, we require semantic information for “eat” to predict a food noun. On the other hand, we require syntactic information to predict the appearance of functional words,
such as prepositions and relative pronouns. We hope the proposed network here can draw attention to the issue of the interaction between syntactic and semantic information.

A frequent complaint about connectionist networks is that the training of the network is like a “black box” -- it is difficult to understand how the network is capable of achieving certain task. As argued by Hadley (Hadley et al, 2001), the success of the network proposed here, which employs two sub-networks similar to Hadley’s Hebbian-competitive network, can be understand at a useful level of abstraction.

Hadley explained that the success of his network has two principle reasons: the successful categorization of input vectors according to their similarity of features in region A of HL1, and the strong pattern recognition capacity of HL2. The two sub-networks in our network are capable of exhibiting strong systematicity for the same reasons. In addition, taking a suggestion from Hadley, we have successfully trained the two sub-networks to be sensitive to different semantic-syntactic structures, by adjusting the learning rate and the constant $c$ in the winner selection rule. That is, the left sub-network is trained to recognize a coarse-grained semantic-syntactic structure of the preceding words, and the right sub-network is trained to recognize a fine-grained semantic-syntactic structure. Also, the coherence reinforcement process successfully switches the attention between the two sub-networks. Hence, the network can always focus on more coherent predictions, and more closely simulate human behavior, especially when encountering semantically anomalous sentences.

We can further compare the network’s behavior with that of human subjects through psychological experiments to see how closely the network is able to address psychological phenomena, especially how humans handle semantic anomalies. However, the human brain is also like a “black box” -- it is difficult to understand how cognitive processes happen in the brain directly. The main source for cognitive psychologists to understand human cognition is to explore the brain indirectly through the understanding of deficient cognition. The same applies to verifications of computational models, since any computational model of human cognitive processes is useless if it cannot address psychological phenomena. Thus, to further verify and challenge the proposed network on
Chapter 7 Discussion and Future Directions

the task of predicting the range of possible next words in a sentence, we can examine whether the proposed network can explain causes of deficits in language acquisition, such as language deficits in aphasia\textsuperscript{11}, or address phenomena on patients with deficits, such as aphasics' responses to semantic anomalies.

Also, as revealed in Chapter 2, Hadley's experiments (Hadley et al, 2001) showed that a shift from local representations to distributed representations could do much better with the problem of systematicity. Our results in Chapter 6 also suggest that the capacity of the proposed network for exhibiting systematicity is highly related to the overlapping of features among words in the same category. If we increase overlap of features among words in a category, it will also increase the predictions of restricted words in the same category when this category is predicted. Is this overlapping of features the only force to help the network exhibit strong systematicity? Further work can be done to verify whether appearances of different words in common contexts also help the propose network (or Hadley's network) do better with the problem of systematicity. For example, a separate vocabulary with every word only having few features (e.g., 10-20\%) overlapped with other words can be used for experimental purpose.

Another future direction is to include negated sentences in the training corpus. Current architecture of the network is not able to deal with the fact that the negation of a semantically anomalous sentences, such as "no boys eat rocks", is not semantically anomalous. However, this can be achieved by both expanding the number of areas in HL1, i.e., memory slides, and enlarging the training corpus with the negation of semantically anomalous sentences. Further experiments can also be done with a wider range of grammatical structures, including the use of different punctuations in a sentence and more varied uses of prepositions. As for semantic anomalies that reside in ambiguous sentences, they need genuine parsing to determine possible structures of the sentences.

\textsuperscript{11} Aphasia is a language disorder, which impairs the comprehension or production of speech, and affects the ability of reading or writing.
Chapter 7 Discussion and Future Directions

Sometimes some structures are semantically anomalous while others are not.\textsuperscript{12} Nevertheless, this is beyond scope of this thesis and beyond Elman’s approach.

As previously stated, we do not claim that the proposed model here provides a general language acquisition mechanism. The lack of biological evidence also means that we cannot be certain of a true computational model for human language acquisition processes in the brain. However, with the employment of both syntactic and semantic information, or rather, information about both fine-grained and coarse-grained syntactic-semantic categories, we believe that the proposed network has successfully provided a possible framework to deal with a subset of semantic anomalies within a connectionist network and raised the issue of the interaction between syntactic and semantic information.

\textsuperscript{12} Professor Raymond Jennings provides an example of such sentences: “I eat breakfast only if there is sufficient food”. This sentence has two possible structures, and one of them is a semantic anomaly.
Bibliography


Bibliography


Bibliography


Appendix A:

Assignment of Features to Words

Assignment of Noun Features to Nouns (no noun features are assigned to verbs):

<table>
<thead>
<tr>
<th></th>
<th>has</th>
<th>had</th>
<th>will</th>
<th>ill</th>
<th>am</th>
<th>of</th>
<th>to</th>
<th>four</th>
<th>is</th>
<th>for</th>
<th>in</th>
<th>in</th>
<th>of</th>
<th>an</th>
<th>the</th>
<th>be</th>
<th>is</th>
<th>on</th>
<th>are</th>
</tr>
</thead>
<tbody>
<tr>
<td>dogs</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>cats</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>birds</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>mice</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>books</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>rocks</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>bats</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>chairs</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>cookies</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>sandwiches</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>chocolate</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>noodles</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Assignment of the 16 Features to Functional Words:

<table>
<thead>
<tr>
<th>with</th>
<th>from</th>
<th>that</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
### Appendix A

**Assignment of Verb Features to Words:**

| Verb          | l | n | v | o | l | l | n | a | n | w | o | r | k | s | n | e | i | l | n | a |
|               | n | o | e | m | f | h | n | e | e | o | u | l | n | d | s | a | t | f | a | d |
|               | s | n | o | e | u | l | n | d | f | a | t | f | a | d | s | a | t | f | a | d |
|               | v | l | e | m | p | t | m | s | e | t | t | t | t | t | t | t | t | t | t | t | t |
|               | a | o | v | l | a | e | r | a | i | s | e | e | e | e | e | e | e | e | e | e | e |
|               | n | e | l | n | r | a | l | n | s | l | e | e | e | e | e | e | e | e | e | e | e |
|               | l | v | s | n | c | n | a | v | t | l | s | e | e | e | e | e | e | e | e | e | e |
|               | a | r | m | e | g | o | e | a | n | l | r | n | u | m | m | e | e | e | e | e | e |
|               | c | r | a | s | m | b | l | i | c | i | c | e | g | b | o | e | e | e | e | e | e |
|               | t | a | r | o | j | v | t | t | m | i | n | j | h | t | f | s | e | n | e | n | e |
|               | m | o | l | p | l | e | u | e | l | e | l | l | a | o | g | b | e | u | i | a | l | y |
|               | o | n | d | n | t | h | e | t | g | e | n | e | s | h | d | t | t | t | t | t | t |
|               | t | w | s | e |
| boys          | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| girls         | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| men           | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| women         | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| dogs          | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| cats          | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| birds         | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| mice          | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| books         | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| rocks         | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| bats          | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| chairs        | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| cookies       | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| sandwiches    | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| chocolate     | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| noodles       | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| eat           | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| bite          | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| chew          | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| swallow       | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| smell         | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| see           | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| watch         | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| sniff         | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| chase         | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| love          | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| follow        | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| move          | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| break         | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| hit           | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| bump          | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0