2013

Embodied Properties of Semantic Knowledge Acquired From Natural Language

Kevin K. Durda
University of Windsor

Embodied Properties of Semantic Knowledge Acquired From Natural Language

by

Kevin Durda

A Dissertation
Submitted to the Faculty of Graduate Studies
through the School of Computer Science
in Partial Fulfillment of the Requirements for
the Degree of Doctor of Philosophy at the
University of Windsor

Windsor, Ontario, Canada

2013

EMBODIED PROPERTIES OF SEMANTIC KNOWLEDGE ACQUIRED FROM NATURAL LANGUAGE USAGE

by

Kevin Durda

APPROVED BY:

D. Inkpen
School of Electrical Engineering and Computer Science
University of Ottawa

T. Traynor
Mathematics & Statistics

R. Gras
Computer Science

L. Rueda
Computer Science

L. Buchanan
Psychology

R. Caron
Mathematics & Statistics

March 12^{th}, 2013

Author's Declaration of Originality

I hereby certify that I am the sole author of this thesis and that no part of this thesis has been published or submitted for publication.

I certify that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis and have included copies of such copyright clearances to my appendix.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

Abstract

The symbol interdependency hypothesis (Louwerse, 2007, 2008) posits that word meaning is dependent upon two sources of information: embodied or grounded knowledge, obtained from observation of and interaction with the physical world, and symbolic or co-occurrence information, gleaned from experience with how words are used together in written and spoken language. This theory assumes that embodied properties of objects influence the statistical structure of language to such an extent that the embodied properties become partially encoded within the structure of language.

The work presented in this dissertation provides support for the symbol interdependency hypothesis by demonstrating that grounded knowledge (in the form of physical and behavioural properties of living and non-living objects) can be identified by analyzing word usage in a large body of written text. An automated method of creating high-dimensional vector-based semantic representations is presented. Several demonstrations show that the representations capture word meaning in a way that aligns with intuition and are able to reproduce non-intuitive results of experiments from the psycholinguistic literature.

A feedforward neural network was trained to produce a list of physical and behavioural properties of an object in response to the object's high-dimensional vector representation. The resulting network was able to identify features of the concepts on which it was trained with near-perfect accuracy and was able to generalize this ability to novel concepts and identify properties of concepts to which it was not previously exposed. These results indicate that there is sufficient information in word usage to identify embodied properties of concepts, a finding that is consistent with the symbol interdependency hypothesis.

Acknowledgments

First and foremost, I would like to say thank you to my supervisors, Dr. Richard Caron and Dr. Lori Buchanan. I would not have been able to complete this without your motivation, expertise, insight, and support. Thank you both for the opportunity to work with you, for your guidance, both dissertation-related and otherwise, throughout the years, and for your patience while I meandered through graduate school. Thank you to my committee members, Dr. Diana Inkpen, Dr. Tim Traynor, Dr. Robin Gras, and Dr. Luis Rueda, for your critiques and comments on my research. For their unwavering support and encouragement, thank you to my parents, Pat and John, and my sister, Krystelle. Thank you to my friends and labmates, Scott Wisdom, Chris Domen, Courtney Heffernan, Tara McHugh, Karey Wilson, Gillian McDonald, Bruce King, and Darren Schmidt. Chris, thanks for all the times we spent studying, discussing research ideas, and the countless rounds of golf. Scott and Courtney, thank you for the late-night study sessions. I wouldn't have made it through grad school without you guys to work and relax with. Finally, thank you, Lindsay McNiff, for being supportive and inspiring, and for your patience, even while the Ancient Ones stirred threateningly in their slumber.

This work was made possible by the facilities of the Shared Hierarchical Academic Research Computing Network (SHARCNET: `http://www.sharcnet.ca/`) and Compute/Calcul Canada.

Table of Contents

List of Figures

List of Tables

1 Introduction

How people are able to communicate through written and spoken language is an open and difficult question. While much progress has been made, research in psycholinguistics and cognitive science has revealed little about the biological bases of language in the brain. Computer scientists in the field of natural language processing (NLP) have produced impressive results by applying formal language methods to natural languages, but these results are typically domain specific and operate on only a restricted subset of the language under consideration. Unfortunately, a general model of computational natural language processing has failed to emerge.

Many difficulties in NLP arise as a result of the syntactic and semantic ambiguities that are abundant in natural language. Contextual information and general semantic knowledge (that is, knowledge about words, the world, and the connections between the two) are essential to solving problems related to ambiguity. For example, the word sense discrimination task, in which the correct meaning of a word with multiple meanings must be selected (for example, determining the correct meaning of the written sentence *He could play the bass* requires identifying the word *bass* to refer to a musical instrument rather than a fish), can only be solved if contextual and semantic information are made available. This raises a difficult question: how can semantic knowledge be acquired, stored, and accessed by a computer? One approach to answering this question is to identify the means by which humans perform these tasks and attempt to simulate these means on a computer.

Quillian (1968; Collins & Quillian, 1969) proposed that semantic knowledge can be efficiently stored in and retrieved from a hierarchical structure. An example of such a hierarchy is shown in Figure 1.1. General concepts, such as LIVING-THING are stored near the top of the hierarchy while more specific concepts are stored deeper in the tree. For example, SPARROW is stored as a subordinate of the concept BIRD, which is in turn stored as a subordinate of the concept ANIMAL, and so on. Properties of concepts are attached at the shallowest possible level in the hierarchy and these properties are inherited by all subordinate concepts[1]. For example, the property ⟨breathes⟩ can be stored at the level of LIVING-THING, since all living things must breathe. Based on this theory of semantic organization, Collins and Quillian (1969) were able to make many detailed predictions about the amount of time it would take subjects to perform simple tasks, such as verifying whether a concept possesses a particular property (e.g., "A bird can

[1] Unless, of course, a subordinate concept has a property that explicitly contradicts an inherited property. For example, the property ⟨can-fly⟩ can be attached at the level of BIRD, and the concept PENGUIN can negate this inherited property with the property ⟨cannot-fly⟩.

fly") and verifying that one concept is an instance of some other concept (e.g., "A bird is an animal"). The amount of time required to perform such a task was posited to be determined by the number of links that must be traversed in the hierarchy, under the assumption that moving up or down a level in the hierarchy required a fixed amount of time and that accessing the properties attributed to some concept also required a fixed (but possibly different) amount of time. Based on these assumptions, this model was able to account for many behavioural results, although it was unable to account for the results of more complicated experiments (Rogers & McClelland, 2004, ch. 1).

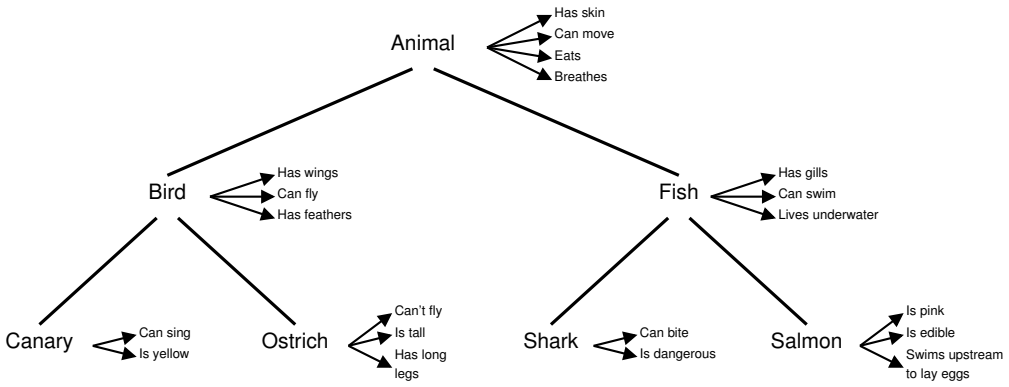

Figure 1.1: A semantic hierarchy of the type used to predict reaction times in linguistic tasks by Quillian (1968) and Collins and Quillian (1969).

Collins and Loftus (1975) extended this theory to accommodate results from several then-recent behavioural experiments. These extensions to the model focused on the mechanics of semantic processing within the model. The strict hierarchical organization was abandoned in favour of a more general semantic network, in which concepts are represented by nodes and knowledge about concepts is stored in various types of relational links connecting the nodes. Some semblance of a hierarchy is still maintained through the inclusion of IS-A links, used to describe the relationship where one concept is an instance of some other concept (e.g., a sparrow IS-A bird). The most important contribution of this revision of the model is the idea that semantic processing is achieved through means of spreading activation. Under this theory, when a concept is processed by an individual (i.e., when the individual engages in some task that requires semantic knowledge about that concept) activation accumulates at the node representing that concept. This activation spreads to other related nodes via the relational links. The amount of activation that spreads is proportional to the strength of the link but inversely proportional to the number of links. Access to the concept occurs when the accumulated activation surpasses some threshold level, at which time that concept's node becomes activated. Once the individual has completed the task and is no longer actively

processing the concept, activation attenuates until it reaches some resting level. Using this simple mechanism of accumulation and attenuation of activation in a semantic network, Collins and Loftus (1975) were able to account for the results of experiments that could not be explained within the framework of the Collins and Quillian (1969) model.

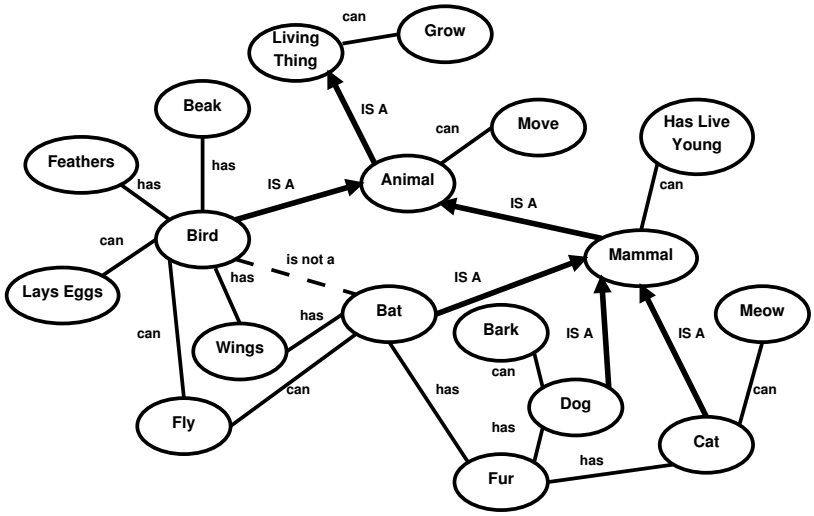

Figure 1.2: A semantic network. Concepts are represented by nodes and the relationships between concepts are indicated by various types of links between the concepts. The latent semantic hierarchy captured by the IS A links is drawn with heavier lines.

Although the model of Collins and Loftus (1975) is able to accommodate a large array of behavioural results, there are many shortcomings. Rogers and McClelland (2004, ch. 1) provide a survey of both the success and failures of the semantic network model. For example, the category inclusion relationship necessitated by the hierarchical nature of knowledge representation applies only to taxonomically organized categories, and even then only for those exemplars that are most typical of the category (Rips, Shoben, & Smith, 1973; Sloman, 1998; Steyvers & Tenenbaum, 2005). Additionally, there is no proposed mechanism by which a semantic network or hierarchy can be constructed, nor any way to determine where new knowledge should be placed in an existing network (Rogers & McClelland, 2004, p. 13).

Methods for constructing simple semantic networks have emerged. Steyvers and Tenenbaum (2005) analyzed the structure of semantic networks constructed from three sources of knowledge: subject produced word association norms (Nelson, McEvoy, & Schreiber, 1999), WordNet (Fellbaum, 1998), and Roget's Thesaurus (Roget, 1911). The resulting semantic networks each possessed both small-world and scale-free structures. That is, the average length of the shortest path between any two nodes (words) in the network was small, and the number of nodes with large degree (that is, nodes with a

large number of neighbours) is higher than expected when compared to random graphs. That these properties are not found together in randomly generated graphs nor in the graphs representing other scientific domains suggests that this combination of properties is an intrinsic feature of semantic organization.

Based on this observation, Steyvers and Tenenbaum (2005) developed an algorithm that employs the process of semantic differentiation, where new concepts added to the network are refinements of previously existing concepts, that could be used to construct a semantic network with the same small-world and scale-free structures as the networks constructed from linguistic resources. Their algorithm begins with a small complete graph of order M. Newly added nodes in the network serve to differentiate complex concepts, where complexity is measured by the degree of the node representing the concept; more complex concepts are more likely to be differentiated by the new node (this is referred to as preferential attachment). Once a node is selected for differentiation, a subset of M of its neighbours (i.e., the nodes that the concept is adjacent to in the network) are made neighbours of the new node by adding edges between the new node and the M randomly selected neighbours. This process is repeated until the order of the network reaches some predefined maximum. While the resulting networks do exhibit the same structural characteristics as the semantic networks constructed from linguistic resources, the networks produced using this algorithm were relatively small. In their experiments, Steyvers and Tenenbaum fixed the maximum order of a network at 5,018 to match the number of words found in the Nelson et al. (1999) association norms. However, Roget's Thesaurus, with nearly 30,000 entries, and WordNet, with over 120,000 entries, both contain much larger vocabularies than the algorithmically–generated networks. In addition, the semantic differentiation process was purely probabilistic, with nodes with higher degree more likely to become differentiated, and did not consider the true complexity of the concepts stored in the network. Further, the nodes in the network were arbitrary and did not correspond to words. This simplification allowed concepts to be differentiated and new edges to be added between nodes arbitrarily without the need for a mechanism to make decisions about whether or not these modifications to the network were consistent with the meanings of the concepts stored in the network. Despite these shortcomings, this early work was able to account for the effects of variables such as age of acquisition and frequency on performance in semantic tasks.

Lemaire and Denhière (2004) also proposed an algorithm to construct semantic networks. Their algorithm takes a large corpus of written text as input and produces a semantic network in which each concept is represented by a node and concepts are connected by weighted relational links. In a manner similar to Hebbian learning (Hebb,

1949), the strength of the relationship between two words is increased when the words occur together in text and is decreased when one of the words occurs without the other. Specifically, the corpus is analyzed using a small window which is centred on each word of the corpus in sequence. The window contains a few words preceding and a few words following the word it is centred upon. In the semantic network, the weights of the edges between the word at the centre of the window and each of the other words in the window are increased. In addition, the weights of the edges between the word at the centre of the window and each neighbour of each of the other words in the window are also increased. Finally, the weight of any edge between the word at the centre of the window and any of its neighbours in the network that do not appear in the window is reduced; if the weight of an edge falls below some threshold the edge is removed. When compared to association norms produced by children, the networks were found to be a good match to the behavioural data. It is unknown whether the networks produced by the algorithm of Lemaire and Denhière (2004) have the same small-world and scale-free properties observed in other semantic networks.

While these early models of semantic network construction have been able to account for some behavioural results and have been successful in constructing networks that share many properties with human semantic knowledge, the complexity of these networks does not approach that observed in human semantics. In addition, the networks work primarily at the level of the concept and give little attention to properties and behaviours of concepts, which are well represented in the semantic networks of Collins and Loftus (1975) and the hierarchies of Quillian (1968) and Collins and Quillian (1969).

Both the semantic hierarchies and the semantic networks describe above use localist representations to represent concepts. That is, each concept is represented by a single dedicated node in a hierarchy or network and these nodes have no innate relationship to the concepts which they represent; the nodes representing, say, BIRD and CAT are the same at their core with the differences residing in their connections to properties and other concepts.

An alternative method of representation is offered by distributed representations (Hinton, McClelland, & Rumelhart, 1986), in which each concept is represented as a unique pattern of activation across a common set of processing units. Empirical evidence suggests that distributed representations are used to represent semantic knowledge in the brain (see Saffran, 2000, for a review). For example, after damage to the brain from stroke or as a result of dementia, some patients exhibit differential processing of living and non-living concepts: living things may be processed with greater ease

than non-living concepts or vice versa (see Gianotti, Silveri, Daniele, & Guistolisi, 1995; Saffran & Schwartz, 1994; Saffran & Sholl, 1999, for reviews). These differences in performance are thought to be a result of localized damage to areas of the brain that store either perceptual knowledge, resulting in reduced capacity to process living objects, or functional knowledge, resulting in reduced performance for non-living objects. On a larger scale, distributed representation is an integral aspect of theories of grounded cognition, which posit that all knowledge is represented across different modalities of the brain and that language is closely tied to other cognitive systems, such as the perceptual, motor, and introspective systems (see Barsalou, 2008, for a review).

Distributed representations are often employed in neural network or connectionist models of cognition. These models aim to investigate the validity of theoretical explanations of cognitive processes by creating computational realizations of the theories that can then be tested and compared to human performance. With regards to the cognitive processes underlying language processing, connectionist models have used distributed representations for both phonological (i.e., aural) and orthographic (i.e., visual) knowledge (Plaut, McClelland, Seidenberg, & Patterson, 1996; Seidenberg & McClelland, 1989; Sibley, Kello, Plaut, & Elman, 2008), as well as semantic knowledge (Harm & Seidenberg, 2004; Hinton & Shallice, 1991; Rogers & McClelland, 2004). Binary feature vectors are the most commonly used method of representing semantic knowledge. In this form of representation, each component of a high-dimensional vector corresponds to a particular feature that can be used to describe a concept, such as ⟨is an animal⟩ or ⟨has legs⟩. A specific concept can be represented by activating all of the components of the vector corresponding to features possessed by the concept and deactivating all other features.

McRae, Cree, Seidenberg, and McNorgan (2005) and Vinson and Vigliocco (2008) provide feature production norms for a number of concepts. Unfortunately, obtaining reliable feature production norms is a time-consuming task even for a small number of concepts. The norms of McRae et al. (2005) provide data for only 541 living and nonliving objects, collected from approximately 725 participants. The norms provided by Vinson and Vigliocco (2008) describe 456 concepts and were collected from 280 subjects. The latter norms are unique in that they provide features produced for objects as well as both nouns and verbs referring to events, while previous sets of norms had provided features only for objects. These norms provide a valuable resource for those investigating the role of perceptual properties of concepts on word recognition and other linguistic tasks and are an excellent resource for creating distributed representations for use in computational models. However, the time consuming process required to obtain the norms poses a serious drawback. McRae et al., for example, began collecting their

norms in 1990 and did not publish them until 15 years later. In addition, features produced by subjects are unlikely to reflect the true nature of semantic memory. As noted by McRae et al. (2005, p. 549), when a participant is asked to provide a list of features describing a concept, they are accessing representations that are developed through experience and interactions with the concept in question; the representation of the concept stored in the subject's mind is not an explicit list of features.

The time-consuming norming process is often by-passed by researchers in favour of a small set of hand-selected features that are assumed to accurately represent the semantics of the concept (Hinton & Shallice, 1991; Plaut & Shallice, 1993). Others have constructed distributed representations from previously existing resources. Harm and Seidenberg (1999, 2001, 2004), for example, used feature vectors generated from the WordNet database (Fellbaum, 1998) However, this approach does not avoid the extensive time investment required to construct the norms, it merely exploits resources to which this time has already been dedicated. Some researchers have relied on similarity between randomly generated binary vectors (Plaut, 1995; Plaut & Booth, 2000; Rodd, Gaskell, & Marslen-Wilson, 2004). This approach is the least satisfying as the components of the randomly generated vectors do not correspond meaningfully to properties of the underlying concepts.

Lexical co-occurrence models offer an alternative method of constructing rich distributed representations of semantic knowledge (M. Andrews, Vigliocco, & Vinson, 2005, 2007, 2009; Burgess & Lund, 2000; Griffiths, Steyvers, & Tenenbaum, 2007; Jones & Mewhort, 2007; Landauer & Dumais, 1997; Lund & Burgess, 1996; Shaoul & Westbury, 2006). These models construct high-dimensional vector representations of semantic knowledge through the direct analysis of word usage in large bodies of written text. These word-usage statistics are referred to as "distributional information". A distinct advantage of lexical co-occurrence models over feature-based models is that the semantic representations are derived automatically from text. Once the corpus has been selected, the method can be allowed to run to completion with little or no human intervention. This stands in stark contrast to the extensive work required to obtain reliable results from subject-collected norms. Once a set of representational vectors has been constructed, similarity between word vectors can act as a surrogate measure of the similarity between the meanings of two words. Any number of similarity measures can be used to measure word similarity, such as the cosine of the angle between two vectors or the correlation between the components of two vectors. In addition, distance metrics such as Euclidean distance or city-block distance can be used to provide a measure of dissimilarity. In contrast to the feature-based vectors discussed above, the individual components of

the vectors constructed by lexical co-occurrence models often do not correspond to features or properties of the concepts they represent in any meaningful way: word meaning is represented in a distributed manner over all components of the vector.

The Hyperspace Analogue to Language (HAL; Lund & Burgess, 1996) was among the first co-occurrence models to be used to demonstrate that word meaning can be derived from distributional information. In this model, the number of times that each pair of words occur within five words of one another is counted. These counts are recorded in a matrix with one row and one column corresponding to each unique word in the corpus and only the columns of this matrix whose entries have the highest variance are retained (typically 100 or 300 columns are kept). The rows of the resulting matrix are used as the representations of the words. Latent Semantic Analysis (LSA; Landauer & Dumais, 1997) is another early co-occurrence model that has been widely used to simulate psycholinguistic tasks. In this model, the corpus is broken into small documents and the number of times that each word occurs in each document is recorded in a matrix that has one row for each unique word and one column for each document. This dimension of this matrix is reduced by using singular value decomposition (SVD) and using the left-singular vectors resulting from this process as the word meaning representations. The Bound Encoding of the Aggregate Language Environment (BEAGLE; Jones & Mewhort, 2007) model incorporates both co-occurrence and order information into its representations by using vector convolution methods. That is, this model incorporates information about the order in which words occur in text into its semantic representations directly. Further, BEAGLE constructs representations through an iterative algorithm and does not require any batch process to reduce the dimension of the vectors, as is required in HAL and LSA. This allows the quality of the vectors to be easily assessed at multiple times during the construction of the vectors, providing insight into the manner in which the model acquires semantic knowledge. Jones, Kintsch, and Mewhort (2006) compared the ability of HAL, LSA, and BEAGLE to simulate subject performance in an array of psycholinguistic tasks and found that BEAGLE demonstrated the strongest performance. Recent models have incorporated more sophisticated mathematical techniques. Turney (2012), for example, describes a dual-space vector model that represents domain similarity and functional similarity in separate spaces and combines the two types of similarity into a single measure. This model was used to simulate semantic relations and semantic compositions and showed strong performance. Van de Cruys, Rimell, Poibeau, and Korhonen (2012) present a method of acquiring verb sub-categorization frame and selectional preference information. The method employs a tensor factorization (a generalization of matrix factorization) to produce representa-

tions that are shown to contain syntactic, lexical, and semantic information. Shaoul and Westbury (2010) examined the impact of several model parameters, including the size of the window used to collect co-occurrence counts and the weight applied to each word in the window, on the the performance of a HAL-like model. They found that a common set of parameters produced the best performance in simulating both lexical decision and semantic decision tasks. Razavi, Matwin, Inkpen, and Kouznetsov (2009) present a model that weights co-occurrence counts according to the context in which two words appear in a sentence. A corpus consisting of a set of documents was analyzed one sentence at a time. Weighted co-occurrence counts are stored in a "closeness" matrix with one row and one column for each unique word in the corpus. For each pair of words that appear together in the same sentence, the corresponding entry in the closeness matrix is incremented by a real-valued weight based on the relationship between the two words in the sentence. For example, two words occurring adjacent to one another were weighted most heavily, and two words separated by a semi-colon were weighted much lower. The resulting co-occurrence matrix was transformed using the Dice coefficient (Dice, 1945). Second-order co-occurrence vectors were used to represent the corpus at different conceptual levels. These vectors were constructed for the sentence level by averaging the co-occurrence vectors representing each word in the sentence. Similarly, document representations were created by averaging the vectors representing each of the sentences in the document. A weighted combination of the word vectors and higher-level vectors was used as the final semantic representation, where the weight was adjusted according to the task being simulated. The resulting vectors were shown to be effective at rating the positivity or negativity of descriptions of dreams and were used to accurately classify abstracts of scientific papers.

Lexical co-occurrence models have been successful at modeling a number of empirical results from the psycholinguistic literature and have proved useful in many tasks in computational linguistics. Schütze (1992, 1998) used an early lexical co-occurrence model together with clustering algorithms to identify the correct meaning of an ambiguous word in a word sense discrimination task. Distributional information has proven useful in several methods for measuring word ambiguity and automatic thesaurus generation (Lin, 1998; McDonald & Shillcock, 2001; Pantel & Lin, 2002; Pereira, Tishby, & Lee, 1993). Durda, Buchanan, and Caron (2009) used representations from a co-occurrence model together with graph clustering techniques and the entropy measure (Shannon, 1948) to provide a measure of word ambiguity. The resulting process was an automated version of the work done by Twilley, Dixon, Taylor, and Clark (1994), eliminating the substantial time commitment required to collect judgments from subjects.

Multidimensional scaling (MDS; a technique which reduces the dimension of a set of points while retaining the pairwise distance between points as best as possible) was used to show that the vectors produced by lexical co-occurrence models contain categorical and grammatical information (Lund & Burgess, 1996; Burgess & Lund, 2000, 1997a; Burgess, 1998). MDS was also used to show that lexical co-occurrence models can differentiate between common and proper nouns, as well as differentiate female names from male names and famous names from common names (Burgess & Conley, 1998a, 1998b). Louwerse, Cai, Hu, Ventura, and Jeuniaux (2006) used MDS to show that representations from the Latent Semantic Analysis model (LSA; Landauer & Dumais, 1997) contain knowledge about the relative order of events that can be used to correctly order the days of the week and the months of the year and contain knowledge about the relative magnitude of units of times and the relative distances between geographical locations.

The representations produced by lexical co-occurrence models have been able to reproduce the differences between associative and semantic priming effects (Burgess & Lund, 1997b; Jones et al., 2006; Lund & Burgess, 1996) and can account for the subtle effects found in mediated priming experiments (Jones et al., 2006; Livesay & Burgess, 1997). These models also perform similarly to humans in synonym selection tasks, such as those found on the TOEFL exam (Landauer & Dumais, 1997; Turney, 2001b).

Lexical co-occurrence has been shown to provide adequate information for children to acquire word meaning (Li, Burgess, & Lund, 2000) and uses statistical techniques that appear to be available to children at very early ages (see Kuhl, 2004). Further, the vector representations produced by lexical co-occurrence models have been shown to be similar to those produced by simple recurrent neural networks (Elman, 1990), which develop internal representations by exploiting information present in the temporal structure of language. Elman (1990) trained a simple recurrent neural network to predict the upcoming word in a stream of text. This network was trained using a small corpus containing 10,000 two– and three–word sentences following 15 different sentence forms with a vocabulary of 29 words. The resulting internal representations (that is, the pattern of activation on the network's hidden units in response to an input pattern) differentiated between verbs and nouns, and within each of these categories were able to differentiate between different forms (for example, within the category of verbs, the network differentiated between verbs that require a direct object, verbs that are intransitive, and verbs for which a direct object is optional). Burgess and Lund (2000) trained the Hyperspace Analogue to Language (HAL; Lund & Burgess, 1996) model on the same corpus as was used by Elman and found that the representations developed by HAL demonstrated a

similar pattern when hierarchical clustering was performed on the resulting representations. While these two approaches acquire semantic knowledge through vastly different mechanisms, both methods use contextual information to guide the learning process and the resulting representations appear to contain similar information. Burgess and Lund (2000) suggest that one advantage of the lexical co-occurrence approach over simple recurrent networks is the ability for lexical co-occurrence models to scale easily to much larger vocabularies; it is not uncommon for co-occurrence models to include representations for 100,000 words or more.

Despite their successes, lexical co-occurrence models have been met with criticism. Early criticisms often focused on the influence of word frequency on the vectors produced by these models. Word frequency has been shown to be a strong predictor of performance in many language-related tasks (Taft & Russell, 1992; Forster & Chambers, 1973; Fredrickson & Kroll, 1976; Monsell, Doyle, & Haggard, 1989; Whaley, 1978) and has been shown to mask the effects of other variables, such as variations in a word's visual appearance, sound, or meaning (S. Andrews, 1982, 1992; Glushko, 1979; D. Jared & Seidenberg, 1990; F. Jared, McRae, & Seidenberg, 1990; Peereman & Content, 1995; Sears, Hino, & Lupker, 1995; Westbury & Buchanan, 2002). Shaoul and Westbury (2006) showed that word frequency varies greatly between corpora, particularly among mid- to low-frequency words, which make up the majority of the words in the English lexicon. Hence, it is important to remove these influences of frequency on the vectors produced by a lexical co-occurrence model. HAL has been criticized for the presence of frequency both within its representations and in the distances between word vectors (Durda & Buchanan, 2008; Lowe, 2000; Shaoul & Westbury, 2006). Durda and Buchanan (2008) showed that frequency effects also exist within the representations produced by the Latent Semantic Analysis model (LSA; Landauer & Dumais, 1997; Landauer, Foltz, & Laham, 1998). To demonstrate this, two lists of words were constructed. The first list contained 40 target words with frequencies ranging between 0.63 and 64,356 per million words. The second list contained 470 test words with frequencies ranging from 1 to 1,383 per million words. For each target word, the cosine similarity[2] with each test word was calculated. The correlation between these similarities and the logarithm of the frequency of the test words was then calculated, resulting in a sample of 40 correlations. These values ranged between 0.12 and 0.75 ($M = 0.39, SD = 0.17$). Typically, higher frequency target words were associated with stronger correlations between frequency and similarity. However, researchers have applied heuristic and statistical tech-

[2]Cosine similarities for the LSA model were calculated using the "general reading up to 1st year college" corpus with 300 factors, using term-term space, available at `http://lsa.colorado.edu`.

niques to produce co-occurrence models whose representations exhibit little sensitivity to word frequency (see Durda and Buchanan (2008) and Shaoul and Westbury (2006) for examples).

A much stronger criticism of lexical co-occurrence models focuses on the lack of groundedness in the representations. Glenberg and Robertson (2000) claim that lexical co-occurrence models are insufficient for representing human semantics and present three experiments to support this claim. Their experiments demonstrate that the LSA model (Landauer & Dumais, 1997) is unable to distinguish between sensible sentences and non-sensible sentences, while subjects were easily able to differentiate between the two. Glenberg and Robertson suggest that these deficits of the LSA model arise because the representations constructed by the model are ungrounded. That is, there is no relationship between the representation of a concept produced by the model and its referent in the real world; the vector representing CAT, for example, has no innate property of "catness". French and Labiouse (2002) make similar criticisms of lexical co-occurrence models, focusing on the absence of essential world knowledge in the vector representations. They posit that this knowledge is not available from word co-occurrence data alone. They demonstrate that the PMI-IR model (Turney, 2001a, 2001b), which scored well on the synonym section of the TOEFL and ESL (English as a Second Language) exams, was unable to differentiate between the suitability of male and female proper names for the name of a father. Similarly, despite an undeclared war between Israel and Palestine at the time of Turney's work, PMI-IR was unable to differentiate between the suitability of traditional Jewish and traditional Arab names for Israeli or Palestinian ministers. Both of these inadequacies are attributed to the absence of grounded knowledge in PMI-IR.

The studies cited above provide evidence indicating that lexical co-occurrence models fall victim to the symbol grounding problem (Harnad, 1990). This problem is well illustrated by the Chinese Room thought experiment of Searle (1980). Consider Searle, a non-speaker of Chinese, sitting in a room. Outside of the room, native speakers of Chinese write questions on slips of paper and slide them under the door. Upon receiving a slip of paper, Searle looks up the Chinese characters on the paper in a large book and, following instructions written in English, transcribes other Chinese characters onto another piece of paper and slides it under the door. If the answer produced in response to each question is reasonable, then to the native speaker of Chinese it would appear that the room understands Chinese. However, from the perspective of Searle, the process of answering the question was merely a tedious exercise in symbol manipulation: in response to a specific sequence of symbols that enter the room, a corresponding se-

quence of symbols is transcribed and slipped back under the door. It is obvious that Searle does not understand Chinese and, even after decades or centuries of performing this task, would remain ignorant of the meanings of the Chinese characters he is manipulating. The goal of Searle's thought experiment was to demonstrate that even if a computer program can behave in a manner that simulates intelligent human conversation, (such as is required to pass the Turing Test; Turing, 1950), the program does not understand the conversation it is having. Both Searle (1980) and Harnad (1990) argue that observing only the relationships between symbols (i.e., words) is insufficient to acquire the meaning of the symbols; the knowledge contained in the relationships between abstract symbols is only sufficient to produce more abstract symbols.

By dint of their construction, lexical co-occurrence models simply observe and record the relationships between words (symbols) as they are used in language. Thus, lexical co-occurrence models are subject to the symbol grounding problem and are unable to acquire word meaning. It is worth noting, however, that these models are capable of analyzing word usage on a scale that was unachievable before sufficient computing power was available; a model can realistically process 10 billion words of text or more in only a few hours. Regardless of the scale of the experience of symbolic models, and despite the many demonstrations that the representations contain knowledge that is typically considered to be semantic, the resulting representations are merely abstract (though complex) symbols.

In addition to these direct criticisms of lexical co-occurrence models, a large number of studies have shown that featural properties of concepts affect subjects' performance in a range of language related tasks, and are thought to play a role in the basic organization of semantic memory (Martin & Chao, 2001; McRae, Cree, Westmacott, & de Sa, 1999; McRae, de Sa, & Seidenberg, 1997; Pexman, Holyk, & Monfils, 2003; Sitnikova, West, Kuperberg, & Holcomb, 2006; Vigliocco et al., 2006). Research in grounded cognition has produced many results that demonstrate that embodied knowledge is strongly linked to performance in many language-related tasks. For example, subjects are faster at recognizing a word when the word is preceded by a related gesture (Krauss, 1998) and subjects showed higher performance when the action required to respond was consistent with the stimuli (e.g., if the response action was to pull a lever, subjects responded more quickly to the word PULL than to PUSH; Glenberg & Kaschak, 2003). Appropriate motor areas of the brain are activated when a subject reads action-related words (e.g., reading the word KICK causes activation in the leg areas of the motor system; Pulvermüller, 2005). Thus, it appears that embodied knowledge is essential to the acquisition of and later access to word meaning, and is represented in semantic memory.

Burgess and colleagues (Burgess, 1998, 2000; Burgess & Lund, 2000) argue that some co-occurrence models are, by nature of their construction, grounded. In HAL, each component of a word's semantic representation measures the relationship between that word and some other word from the corpus. This produces a form of grounding in the linguistic environment. In addition, because meaning of abstract concepts is acquired and stored in the same way as for concrete concepts, abstract concepts are also grounded. Unfortunately, this argument closely mirrors that used by Glenberg and Robertson (2000), as well as Searle (1980) and Harnad (1990), against symbolic cognition: that relationships between abstract symbols cannot give rise to meaning, but only to more abstract symbols. Burgess (2000) also rejects the criticisms of Glenberg and Robertson (2000) on the grounds that lexical co-occurrence models are purely representational models, but the tasks in the experiments of Glenberg and Robertson largely depend on linguistic processing. If research focuses primarily on representational issues, or if there is a close relationship between representation and processing, as in semantic priming, then lexical co-occurrence models provide a useful tool. However, in tasks that require extensive processing by the cognitive system, lexical co-occurrence models fall short. As Burgess states, "it is simply not reasonable to plop LSA or HAL vectors into a similarity comparison and pretend that it is reflecting the active comprehension process" (Burgess, 2000, p. 404). Regardless of whether one is able to look past the methodological shortcomings of Glenberg and Robertson, it is clear that lexical co-occurrence models suffer from the symbol grounding problem as described by Harnad and exemplified in Searle's Chinese Room.

Other researchers have attempted to overcome these shortcomings of lexical co-occurrence models by integrating both co-occurrence information and featural data obtained through feature norms collected from subjects. M. Andrews et al. (2005, 2007, 2009) present a model that treats distributional and featural information as a joint distribution to be learned by a Bayesian model. They demonstrated that the representations produced by this model are better able to reproduce behavioural data than are models that include only one of the two sources of information or those that treat the two sources as independent distributions. Further, Andrews et al. posit that only some concepts in their model become grounded and that treating distributional and featural information as a single joint distribution allows for chains of inference, where embodied knowledge about one concept is generalized to other concepts. Howell, Jankowicz, and Becker (2005) trained a simple recurrent neural network, augmented with noun– and verb–feature units, to predict the next word in a stream of language. When the model was provided with correct featural data in addition to linguistic data during training, the

resulting network was better able to predict the upcoming word than when randomized featural data was provided. This demonstrates that sensorimotor data improves the network's ability to learn the statistical structure of language.

Riordan and Jones (2011) compared featural and co-occurrence-based representations in a variety of semantic clustering tasks. These experiments revealed that both types of representation performed similarly on clustering tasks involving both concrete nouns and action verbs, however, the type of semantic knowledge employed by each model was often very different. This suggests that semantic knowledge is, to a large extent, encoded redundantly in both embodied and linguistic sources, and that these two sources of information act to compliment one another.

Louwerse (2007, 2008) argues that language is both embodied and symbolic and that the meaning of a word is reliant upon both that word's embodied properties and its relationships to other words. Louwerse (2007) calls this the *symbol interdependency hypothesis* and proposes that, while symbols can always be grounded, language operates largely upon symbolic representations and that the grounded representations of words are not necessarily accessed during comprehension and communication (although they may be partially activated). This theory is similar to the chains of inference in the joint distributional model of M. Andrews et al. (2005, 2007, 2009). The symbol interdependency hypothesis also posits that symbolic representations of words are "built onto" embodied representations and that a large amount of information about the meaning of words is available in the distributional information present in language usage. Thus, under this theory, grounded knowledge about words is still necessary to acquire true meaning of words, but distributional information is adequate to provide a large portion of meaning. Based on studies that examine iconicity and word frequency, Louwerse (2008) proposes that embodied relationships have a strong influence over the statistical structure of language. This influence is prevalent in language usage to the extent that embodied properties of words actually become encoded in language usage.

The work in this dissertation provides support for the symbol interdependency hypothesis of Louwerse (2007, 2008) by demonstrating that embodied information is at least partially encoded in the statistical structure of language. Specifically, a lexical co-occurrence model, described in the next chapter, is used to create vector-based semantic representations for a large number of words. Chapter 3 contains simulations and experiments that show that this model is able to reproduce a large number of empirical results from behavioural experiments and that the representations contain information that is typically semantic in nature. Chapter 4 contains experiments demonstrating that these representations contain information that can be used to identify embodied prop-

erties of the concepts that the vectors represent. This is achieved through the use of a feedforward neural network that is trained using backpropagation. Data demonstrating that the neural network is able to generalize this ability and identify embodied properties of novel concepts is provided. That is, given the vector representation constructed from the co-occurrence data for a concept on which the network was not trained, the network is able to correctly identify embodied properties of that concept with accuracy greater than would be expected by chance. These results are interpreted to provide direct support for Louwerse's theory that embodied knowledge is embedded in the structure of language.

2 Creating Co-occurrence Representations

This chapter describes the method used to derive word-meaning, or semantic representations. A description of the corpus used in the experiments in Chapters 3 and 4 is provided in Section 2.2. Section 2.3 describes two implementations of the method described in this chapter.

2.1 Description of Model

The first step of developing semantic representations is to count the number of times that each pair of words occur near one another in a large corpus of written text. These are referred to as *co-occurrence counts*. To simplify this task, we first define a fixed set of words, $\mathscr{D}$, which we call the *dictionary*, as $\mathscr{D} = \{w_1, w_2, \ldots, w_{|\mathscr{D}|}\}$, where $|\cdot|$ denotes set cardinality. Each w_i in $\mathscr{D}$ denotes a unique word and we will often refer to some word from the dictionary as $w \in \mathscr{D}$ or simply w. We only count co-occurrences between pairs of words that are both in $\mathscr{D}$. Let $\mathscr{T}' = \left(t'_1, t'_2, \ldots, t'_{|\mathscr{T}'|}\right)$ be an ordered list of words, called the *corpus*, such that for every $t'_i \in \mathscr{T}'$ there is a $w_j \in \mathscr{D}$ with $t'_i = w_j$. We can think of $\mathscr{T}'$ as the concatenation of a large number of documents, where each document is written using words from $\mathscr{D}$. If $\mathscr{D}$ contains every word from the English language, then we could represent any written English work as a sequence of words from $\mathscr{D}$. However, natural language is fluid and new words are added to the dictionary of a language continuously. If $\mathscr{D}$ is fixed today, it will quickly become outdated as new words enter common usage and gain status as words. Further, different geographical regions often use different dialects of the same language and what is considered a word in one area of the world may be gibberish in another. To avoid these problems, a fixed dictionary, which we still refer to as $\mathscr{D}$, that contains only a subset of English words is used and only co-occurrences between pairs of words both found in $\mathscr{D}$ are counted. Thus, $\mathscr{T}'$ may contain many words that do not appear in the dictionary. To alleviate this, a special token denoted by $w^\star$ is added to $\mathscr{D}$ and any t'_i in $\mathscr{T}'$ that does not appear in $\mathscr{D}$ is replaced with $w^\star$. This modified corpus is used as input to the model and is denoted $\mathscr{T} = \left(t_1, t_2, \ldots, t_{|\mathscr{T}|}\right)$. Note that $|\mathscr{T}| = |\mathscr{T}'|$ and that $t_i = t'_i$ if $t'_i \in \mathscr{D}$; otherwise $t_i = w^\star$. Co-occurrences of other words with $w^\star$ are not counted. It will be shown later that, provided the most frequently used words are included in the dictionary, nearly all tokens in the corpus will be found in even a small dictionary[3]. That is, by excluding only low-frequency words from the dictionary, we can drastically reduce its size (since English contains a very large number of infre-

[3] In this context, "small" means somewhere below 100,000 entries. There are at least a quarter-million words in the English language, depending on how words are counted.

quently used words) while still recognizing most of the words in the corpus as words in the dictionary, $\mathscr{D}$.

We define the *frequency* of a word $w \in \mathscr{D}$ by

$$f(w) = |\{k : t_k = w, t_k \in \mathscr{T}\}|.$$

That is, a word's frequency is the number of times it appears in the corpus. We assume that each word in the dictionary occurs at least once in the corpus, so $f(w) \geq 1, \forall w \in \mathscr{D}$.

Given an ordered pair of words, $(w_i, w) \in \mathscr{D} \times \mathscr{D}$, we define the n^{th} *co-frequency* of w_i given w by

$$f^n(w_i \mid w) = |\{k : t_k = w \text{ and } t_{k+n} = w_i\}|.$$

The value of $f^n(w_i \mid w)$ is the number of times that w_i occurs exactly $-n$ words before w if $n < 0$, or n words after w if $n > 0$. The word w can be considered a "target" word and w_i an "associate" word that occurs near the target word. To illustrate the above definitions, consider the following sentence:

THE BIG BLACK BEETLE BIT THE BIG BLACK BUG.

The frequency of THE is $f(THE) = 2$. The word THE appears immediately before the word BIG two times, so $f^{-1}(THE \mid BIG) = 2$, and BLACK appears twice immediately after the word BIG, so $f^1(BLACK \mid BIG) = 2$.

Given two integers, $n_0 \leq 0$ and $n_1 \geq 0$, with $n_0 \neq n_1$, we can record the co-frequency data for each pair of words and each n, $n_0 \leq n \leq n_1, n \neq 0$, in a three-dimensional array, N, where N has order $|\mathscr{D}| \times |\mathscr{D}| \times (n_1 - n_0)$ and is indexed by target word (w), associate word (w_i) and position (n). It is helpful to think of n_0 and n_1 as defining a small window of the corpus containing the $|n_0|$ words preceding and n_1 words following some instance of w in the corpus:

$$\left(t_{i+n_0}, t_{i+n_0+1}, \ldots, t_{i-1}, t_i = w, t_{i+1}, \ldots, t_{i+n_1-1}, t_{i+n_1}\right).$$

This window is passed sequentially over each word in $\mathscr{T}$, accumulating the co-frequency values incrementally in the co-frequency array N. As the window is passed over each word in $\mathscr{T}$, the co-occurrence data is recorded in N. For example, if a particular window centred on some instance of the word w_i contains the word w_j exactly three words after t, then the value of $\mathsf{N}[i][j][3]$ is incremented to record to co-occurrence.

We define the *co-occurrence frequency* of w_i given w as the sum of the co-frequencies

across all values of $n, n_0 \leq n \leq n_1, n \neq 0$:

$$f(w_i \mid w) = \sum_{\substack{n=n_0 \\ n\neq 0}}^{n_1} f^n(w_i \mid w).$$

The value of $f(w_i \mid w)$ is the number of times that w_i appeared $|n_0|$ or fewer words before w or n_1 or fewer words after w in the corpus. That is, $f(w_i \mid w)$ is the number of times that w_i appeared in a window centred on an instance of w in the corpus[4]. The co-occurrence frequencies can be stored in a $|\mathscr{D}| \times |\mathscr{D}|$ matrix, M, which is called the co-occurrence matrix. If the co-frequency data has been stored in a three-dimensional array N, as described above, then the co-occurrence matrix M can be calculated from N by simply summing across all values of $n, n_0 \leq n \leq n_1$[5]. That is,

$$\mathsf{M}_{i,j} = \sum_{\substack{n=n_0 \\ n\neq 0}}^{n_1} \mathsf{N}_{i,j,n} = f(w_j \mid w_i).$$

Note that this method of constructing the co-occurrence matrix results in equal weighting of each window position; no advantage is given to words that appear closer to the target word within a window. Due to the distribution of words in language, both the co-frequency array and the co-occurrence matrix are extremely sparse (that is, the matrices have very few non-zero entries). Examples of this will be given in the next section.

The frequencies of w_i and w have a strong influence on the value of $f^n(w_i \mid w)$, with higher frequency words more likely to occur together merely by chance than low frequency words. As a result of this, a co-occurrence of a word with a high-frequency word is not as informative of the relationship between the words as is a co-occurrence with a low-frequency word.

For example, in the one billion word corpus used in the experiments in Chapters 3 and 4 (see Section 2.2 for a description of the corpus) the word CAT appears 30,617 times.

[4]Note that unless $n_0 = -n_1$, w will not appear in the centre of the window. However, the term "centered" will be used to refer to the situation where a window contains n_0 words before and instance of w and n_1 words following an instance of w, regardless of whether w actually appears at the true centre of the window.

[5]The reason for making a distinction between co-frequency and co-occurrence values and their associated matrices, N and M is purely practical. Collecting the co-frequency and co-occurrence matrix is time consuming and requires both a large amount of memory and a large amount of temporary disk space. To make experimentation with different values of n_0 and n_1 feasible, the co-frequency data can be collected by using a large window defined by two values, m_0 and m_1, defined analogously to n_0 and n_1. Once this is complete, the values of n_0 and n_1 can be fixed, with $m_0 \leq n_0 \leq 0$ and $0 \leq n_1 \leq m_1$, and the co-frequency data can be summed across all values of $n, n_0 \leq n \leq n_1, n \neq 0$ to obtain the co-occurrence counts. This allows for comparing results using different window sizes without the computational burden of collecting new co-occurrence data for each different window size.

Using a window that contains ten words preceding and five words following the word on which it is centred, DOG occurs with CAT 1,040 times (that is, $f(DOG \mid CAT) = 1{,}040$). The word MEOW occurs with CAT only 37 times when the same size window is used (that is, $f(CAT \mid MEOW) = 37$). However, when no regard is given to context, DOG occurs a total of 49,391 times in the corpus, while MEOW occurs only 308 times. Thus, the proportion of occurrences of MEOW that occur in the presence of an instance of CAT is higher than the proportion of occurrences of DOG that occur in the presence of CAT. This seems to suggest that a co-occurrence of CAT with the lower-frequency word MEOW is more informative of the meaning of CAT than is a co-occurrence of CAT with the higher-frequency word DOG, and that the total number of co-occurrences between two words must be weighted according to the frequencies of the words. Further, closed-class words such as THE appear with high frequency near all other words and will thus produce high co-occurrence counts with almost all words. The number of times that THE occurs near CAT is 34,615 (that is, $f(THE \mid CAT) = 34{,}615$), which is greater than the number of occurrences of CAT (this situation arises because THE sometimes appears more than once in a single window centred on an instance of CAT). However, given that the total number of occurrences of THE in the corpus is 71,936,637, which is about 7.5% or one out of every 13 words in the corpus, it is hardly surprising that CAT and THE occur together with such high frequency. It is clear that the word THE contributes little to the meaning of any word, so it is important to weight the number of co-occurrences with such words to account for their high frequency.

Word frequency has also been shown to be a strong predictor of performance on many psycholinguistic tasks: subjects perform more quickly and more accurately in tasks involving high frequency words than in the same task with low frequency words (Taft & Russell, 1992; Forster & Chambers, 1973; Fredrickson & Kroll, 1976; Monsell et al., 1989; Whaley, 1978). This frequency effect in psycholinguistic tasks can mask other, more subtle effects, such as true semantic effects (S. Andrews, 1982, 1992; Glushko, 1979; D. Jared & Seidenberg, 1990; F. Jared et al., 1990; Peereman & Content, 1995; Sears et al., 1995; Westbury & Buchanan, 2002)

Finally, Shaoul and Westbury (2006) have shown that word frequency varies greatly between corpora, particularly among low frequency words. Of course, because each corpus acts as an analogue to an individual's experience with language, they will each produce different sets of semantic vectors. However, there should remain strong similarities between the internal structures of semantic spaces constructed from different corpora, regardless of differences in the frequencies of particular words. Thus, it is important to remove these influences of word frequency from the co-occurrence counts if

we wish to capture the semantic characteristics of the words in the dictionary.

To reduce frequency effects, we use the log-relative frequency ratio (Damerau, 1993). This measure was originally used to identify a vocabulary of words related to a particular subject by comparing a word's usage in a general corpus of text to that same word's usage in a subject-specific corpus. Here, we use the same technique to compare a word's usage in the presence of some other word to that same word's usage without regard to context. To calculate the log-relative frequency ratio, we define two probabilities. The probability that some word in a window centred on an instance of w is w_i is given by

$$P_w(w_i) = \frac{f(w_i \mid w)}{(n_1 - n_0) f(w)}. \tag{2.1}$$

The denominator in Equation 2.1 is the total number of co-occurrences counted around the word w. The probability that a word randomly selected from $\mathscr{T}$ is w_i is given by

$$P(w_i) = \frac{f(w_i)}{|\mathscr{T}|}.$$

Since each word appears at least once in the corpus, $P(w_i) > 0$ for all $w_i \in \mathscr{D}$. However, since some words occur only infrequently, this probability may be very small for some words. The log-relative frequency ratio is calculated as

$$R(w_i \mid w) = \log\left(\frac{P_w(w_i)}{P(w_i)}\right). \tag{2.2}$$

If w_i is more likely to appear in the presence of w than when context is ignored then $R(w_i \mid w)$ is positive. If w_i occurs with smaller probability in the presence of w, then $R(w_i \mid w)$ is negative. Equation 2.2 is applied to each element of the matrix M to produce a $|\mathscr{D}| \times |\mathscr{D}|$ matrix $\mathsf{R} = (r_{ij})$, where $r_{ij} = R(w_j \mid w_i)$.

Consider the examples given above as a demonstration of how the log-relative frequency ratio can remove frequency effects. Recall that the corpus contained one billion words (to be exact, there were 962,070,534 words) and co-occurrence counts were collected using a window that contained ten words preceding the target ($n_0 = -10$) and five words following the target ($n_1 = 5$). Substituting these values into (2.2), we obtain $R(DOG \mid CAT) = 3.78669$ and $R(MEOW \mid CAT) = 5.52806$. Further, for the high-frequency word THE, which should contribute little to the meaning of CAT, the log-relative frequency ratio is $R(THE \mid CAT) = 0.00798$. Recall the co-frequencies given earlier: $f(DOG \mid CAT) = 1{,}040$, $f(MEOW \mid CAT) = 37$, and $f(THE \mid CAT) = 34{,}615$. In the co-frequency data, THE occurs most frequently with CAT, followed by DOG and

MEOW with the fewest occurrences with CAT. After applying (2.2), this pattern reverses: the word MEOW has the strongest association to CAT and the high-frequency word THE shows little relationship to CAT. This example suggests that the log-relative frequency ratio weights co-occurrence counts according to word frequency in an intuitively appealing way. One advantage of using this method to remove word frequency effects is that the influence of function words on the semantic representations is drastically reduced without the need for any preprocessing of the input corpus to tag each word for part-of-speech or to maintain a list of stop words. This allows the algorithm to be implemented with relative ease without the need for any specialized techniques.

Finally, the dimension of the matrix R is reduced by using singular value decomposition (SVD)[6]. We can write R as the product of three matrices,

$$\mathsf{R} = \mathsf{U}\Sigma\mathsf{V}^\top, \tag{2.3}$$

where U is a $|\mathscr{D}| \times |\mathscr{D}|$ orthogonal matrix whose columns are eigenvectors of $\mathsf{R}\mathsf{R}^\top$ (called the left singular vectors of R), V is a $|\mathscr{D}| \times |\mathscr{D}|$ orthogonal matrix whose columns are eigenvectors of $\mathsf{R}^\top\mathsf{R}$ (called the right singular vectors of R), and Σ is a $|\mathscr{D}| \times |\mathscr{D}|$ a diagonal matrix whose diagonal elements are called the singular values of R and appear in decreasing order. An approximation to R can be obtained by fixing a positive integer k and setting all but the k largest singular values to 0 and retaining only the first k columns of both U and V. Thus, we have

$$\tilde{\mathsf{R}} = \tilde{\mathsf{U}}\tilde{\Sigma}\tilde{\mathsf{V}}^\top, \tag{2.4}$$

where both $\tilde{\mathsf{U}}$ and $\tilde{\mathsf{V}}$ have order $|\mathscr{D}| \times k$ and orthogonal columns, and $\tilde{\Sigma}$ is diagonal with order $k \times k$. The matrix $\tilde{\mathsf{R}}$ is the best approximation (in the least squares sense) to R having rank k. The rows of $\tilde{\mathsf{U}}$ are used as the representations of the words. That is, the i^{th} word of the dictionary, $w_i \in \mathscr{D}$, is represented by the i^{th} row of $\tilde{\mathsf{U}}$. The word "representation" in this context is used to mean an abstract symbol that is intended to capture properties of the semantic content of a word. In the model presented above, these representations take the form of k-dimensional vectors. In Sections 3 and 4 it will be shown that the vectors produced using this method contain semantic information that captures many properties of human semantic memory and that the vectors contain information about embodied properties of objects.

Given two words, w_a and w_b, with corresponding vectors $\boldsymbol{w_a} = (w_{a1}, w_{a2}, \ldots, w_{ak})$ and

[6]The SVD was computed using Doug Rodhe's SVDLIBC, available at `http://tedlab.mit.edu/~dr/SVDLIBC/`

$\boldsymbol{w_b} = (w_{b1}, w_{b2}, \ldots, w_{bk})$, the similarity between w_a and w_b, denoted $\sigma(w_a, w_b)$, is given by the cosine of the angle between their vectors,

$$\sigma(w_a, w_b) = \frac{\sum_{i=0}^{k} w_{ai} w_{bi}}{\|\boldsymbol{w_a}\| \|\boldsymbol{w_b}\|}, \tag{2.5}$$

where $\|\cdot\|$ denotes the Euclidean norm of a vector[7]. Higher values of $\sigma(w_1, w_2)$ correspond to higher similarity between w_1 and w_2 and negative values are interpreted as low similarity rather than opposition in meaning. Recall that the vectors $\boldsymbol{w_a}$ and $\boldsymbol{w_b}$ are given by the a^{th} and b^{th} rows of $\tilde{\mathsf{U}}$, respectively.

2.2 Input Corpus

For the experiments in Chapters 3 and 4, representations were created from a corpus containing all articles from Wikipedia that contain over 2,000 words, provided by Shaoul and Westbury (2009). A dictionary of the 100,000 most frequent words was used. There were over 962 million words in the corpus, of which over 931 million, or 96.8%, appeared in the dictionary. The corpus contained 3,035,070 articles. A window containing ten words preceding and five words following the word around which the window was centred was used to collect co-occurrence data from the corpus (that is, $n_0 = -10$ and $n_1 = 5$). This window size has been found to work well by other researchers (Shaoul & Westbury, 2008). Shaoul and Westbury (2010) showed that a flat weighting of the window positions, as used in the current method, works well when simulating an array of behavioural results. As the goal is not to provide the "best" co-occurrence model, but rather to produce a model that captures semantic knowledge effectively, the space of model parameters will not be explored further and the window size will be set based on the prior work of Shaoul and Westbury.

The co-occurrence matrix produced from the corpus, M, was extremely sparse, with only 4.09% of the entries non-zero. The dimension of this matrix was reduced to only 300 dimensions using SVD ($k = 300$, in the notation of the previous section). The matrix $\tilde{\mathsf{R}}$ resulting from the SVD is dense. Note that in all experiments in Chapter 3 where a measure of similarity between pairs of words was required, the cosine between the words' vectors, given in Equation 2.5, was used.

[7]The Euclidean norm of a vector $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ is given by $\|\boldsymbol{x}\| = \sqrt{\sum_{i=1}^{n} x_i^2}$.

2.3 Implementation Details

Two implementations of the above algorithm were written in C++. The first was designed to run on a relatively modern desktop PC and the other to run on a computer with a very large amount of memory (i.e., 30GB or more).

Both implementations analyzed the corpus in two stages. In the first stage, the frequency of each unique token in the corpus was counted. From this data, a list of the 100,000 most common tokens was determined; this list served as the dictionary during the second pass through the corpus. During this pass, the co-occurrence frequency of each pair of words was calculated and saved to a file on disk. Finally, a SVD was performed on the co-occurrence data to obtain the semantic representations.

Common to both implementations was the requirement to efficiently parse a large corpus of text into words. This was accomplished via a lexer written in GNU flex (an alternative to the Lex tool). This lexer was used to scan the corpus a single token at a time. During the first pass through the corpus, the frequency of each token was counted and the 100,000 most frequent words were identified. Each word in the 100,000 word dictionary was assigned a unique integer between 0 and 99,999. These integers were used as unique identifiers for each word. As each word was read from the corpus, its unique identifier was determined and used in all further processing. As the corpora processed by the program were on the order of one billion words, it was essential that this identifier look-up could be performed as efficiently as possible. Even a perfectly balanced binary search tree would require, on average, 16.6 look-ups to find a word (although some savings could be gained by storing words at non-leaf nodes). Further, the comparison required at each node during the search through the tree is a string comparison, whose complexity is determined by the length of the string. A trie (Fredkin, 1960) is a natural alternative to the binary search tree for string-based look-ups. This data structure can perform look-ups in $O(\ell)$ time, where ℓ is the length of the string. Further, the comparison performed at each step of the search is a simple character comparison that can be performed in constant time. In the particular corpus used in the experiments in the following chapter, the average number of comparisons performed by the trie for each look-up was 4.86. Given the size of the corpus, the use of the trie for dictionary look-ups produced a substantial gain in efficiency.

Given the size of the dictionary and the number of unique word-pairs observed in the corpus, the co-occurrence data consumed a large amount of memory and could not be stored in its entirety in the RAM of a desktop PC. Thus, only a subset of the co-occurrence matrix was stored in memory at any time. This subset contained entries only for those word pairs and windows positions that were most recently observed in

the corpus. This matrix was stored in an AVL tree whose nodes were ordered by target word, associated word, and window position and contained the number of times that the combination of words and window position were observed. Once this tree grew beyond a specified number of nodes, its contents were written to a file on disk and the tree was emptied. This process was continued until the entire corpus was scanned. The files on disk were then merged into a single file using merge sort. This merged file corresponds to the matrix N described above. Given a minimum and maximum window position, a co-occurrence matrix M was extracted from this file and all non-zero entries were saved to a file on disk.

This version of the program worked well for moderately large corpora, but was inefficient for very large corpora due to the requirement to write a large amount of data to disk. However, the incremental nature of the construction of the co-occurrence matrix allows a co-occurrence matrix to be constructed from a smaller corpus and later updated in stages to include data from increasingly larger corpora. This property is also convenient for examining how representations change with exposure to greater amounts of text. A further drawback of this implementation was the large amount of temporary disk space required to store the intermediate results of the algorithm.

The implementation of the algorithm intended for machines with a large amount of memory was straightforward. The co-occurrence data was stored in a 100,000 by 100,000 two-dimensional array allocated in memory. As each word was read from the corpus, the co-occurrence data could be updated directly in memory. One deviation from the method described above was that the co-occurrence data was not split by distance from the target; co-occurrences in all window positions were recorded in the same matrix. That is, the matrix M was constructed directly, rather than constructing N and summing across window positions to calculate M. This reduced the amount of memory required to hold the co-occurrence matrix in memory. For example, when using a windows with ten preceding words and five following words, as in the experiments in this dissertation, the amount of memory required to store N is 15 times the amount of memory required to store M. Constructing M directly allowed all operations to be performed in memory without any caching of results to disk. Once the entire corpus was processed, any non-zero entries in the co-occurrence matrix were written to a file on disk, using the same file structure as the variation of the program intended for use on a desktop PC.

In both implementations, the SVD was performed using Doug Rodhe's SVDLIBC library, available at `http://tedlab.mit.edu/~dr/SVDLIBC/`. This library is based on SVDPACKC (Berry, Do, O'Brien, Krishna, & Varadhan, 1993), a library of methods for calculating the SVD of large, sparse matrices.

3 Demonstrations of Model

This chapter contains several demonstrations that the vectors produced by the model above capture many characteristics of human semantic memory in a way that is independent of the frequency of the words in the corpus.

3.1 Independence of Frequency

The preceding chapter described a technique that can be used to reduce the influence of word frequency during the construction of semantic vectors. To demonstrate that the resulting semantic vectors produce similarity measurements that are independent of frequency, a random sample of 10,000 pairs of words with frequency ranging between two and 78,457 per million words of written text was selected. The similarity between each pair of words was calculated. No correlation was found between similarity and the frequency of the first word in the pair[8], $r(9998) = -0.034$), or between similarity and the frequency of the second word, $r(9998) = -0.031$. Although both of these correlations are significant (both p's < 0.01), this is a consequence of the large sample size rather than any meaningful relationship between frequency and similarity. Indeed, Figure 3.1 shows that there is little relationship between the two variables.

3.2 Multidimensional Scaling Results

In this section, multidimensional scaling (MDS) is used to demonstrate that the vector representations capture categorical information. MDS is also used to show that the representations capture information about the temporal order of events (i.e., the days of the week or the months of the year) and the relative magnitude of units of time and measurement. MDS reduces the dimension of a set of data points in a way that best retains the pairwise distances between points (see Borg & Groenen, 2005, for example).

3.2.1 Categorical Information

Burgess and Lund (1997a) used MDS to demonstrate that representations produced by a co-occurrence model latently encode categorical information. Exemplars from four

[8]The results of statistical tests are reported using notation that generally agrees with the APA standard. Each statistic is reported with the degrees of freedom in parentheses and the significance of the statistic following. For analysis of variance (ANOVA) analyses, the F-statistic is reported with two degrees of freedom (e.g., $F(1,23) = 12.2, p < 0.001$). t-tests are reported similarly (e.g., $t(12) = 8.21, p < 0.001$), as are correlations (e.g., $r(99) = 0.34, p = 0.012$. Generally, statistics are reported in line; however, means and standard deviations are reported in parentheses using the format ($M = 63.3$, $SD = 12.3$).

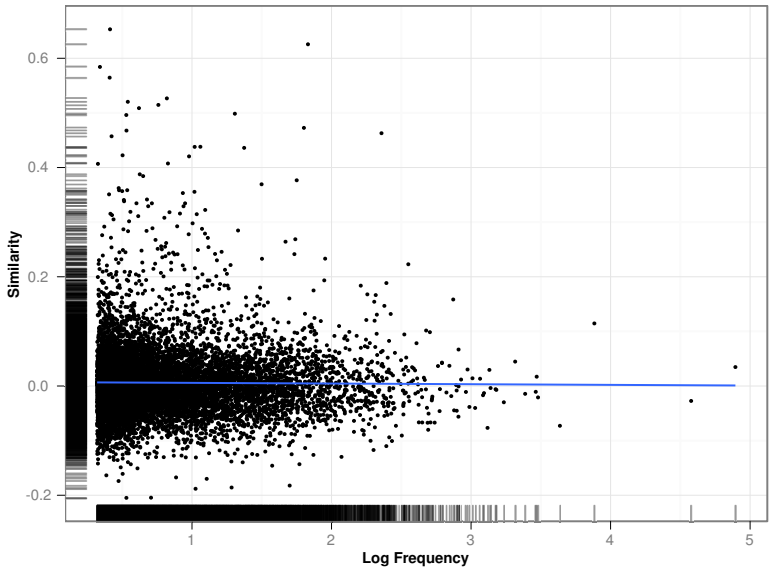

Figure 3.1: Scatter plot of word similarity against word log-frequency. The nearly horizontal best fit line demonstrates that there is little relationship between frequency and similarity.

categories (Animals, Body Parts, Geographical Locations, and Cities) were selected and scaled to two dimensions using MDS. Concepts from a common category grouped together in the lower-
dimensional space, suggesting the representations capture knowledge about category membership without explicit exposure to this information. The present model is able to reproduce the results of Burgess and Lund (1997a). Several other experiments examining categorical information are also presented.

Figure 3.2 shows the results of an MDS performed on the same set of concepts as were used by Burgess and Lund (1997a). Words from each category cluster together in the two-dimensional space. While Burgess and Lund observed some overlap between similar categories (some Cities clustered with the Geographical Locations and vice versa, and some Body Parts clustered with the Animals), this does not occur in the current results.

To verify this clustering, within- and between-category similarities were analyzed in a one-way ANOVA. For each category, all pairwise similarities between pairs of words in the category were calculated (the within-category distances), and the similarity between each word in the category and each word in each of the other categories was calculated (the between-category distances). For example, for the category Animals, the similarity between each pair of words from the category Animals was calculated to provide the

Figure 3.2: Multidimensional scaling of animals, body parts, geographical locations, and cities. Concepts from a common category cluster closer together than concepts from different categories.

within-category similarities. In addition, the similarity between each animal and each concept from the other categories (Body Parts, Geographical Locations, and Cities) was calculated to provide the between-category similarities. Since the data are not independent, a separate ANOVA was performed for each category. This analysis revealed that the Animals were more similar to one another than to concepts from other categories, $F(1,257) = 242.44, p < 0.001$. The same pattern was observed for the category of Body Parts, with higher within-category similarities than between-category similarities, $F(1,553) = 352.63, p < 0.001$, as well as for Cities, $F(1,220) = 138.26, p < 0.001$, and Geographical Locations, $F(1,331) = 305.35, p < 0.001$. Since Cities and Geographical Locations are highly similar categories, an additional analysis was performed to compare the within- and between-group similarities for only these two categories. Again, Cities were differentiated from Geographical Locations, $F(1,88) = 31.686, p < 0.001$, and Geographical Locations were differentiated from Cities, $F(1,133) = 51.413, p < 0.001$. Note that all of these analyses were performed on the vectors in the original high-dimensional space and not on the results of the MDS[9]. Hence, this analysis suggests that the distinction be-

[9]This is true for all analyses presented in this section. Any statistics calculated on the similarity values use similarities calculated in the original 300-dimensional space. The MDS results provide a projection of the high-dimensional space to a lower-dimensional space for convenient visualization.

tween categories observed in the results of the MDS exist in the high-dimensional space and can not be dismissed as an artifact of the MDS procedure. Figure 3.3 illustrates the higher similarity between concepts within the same category than between concepts from different categories.

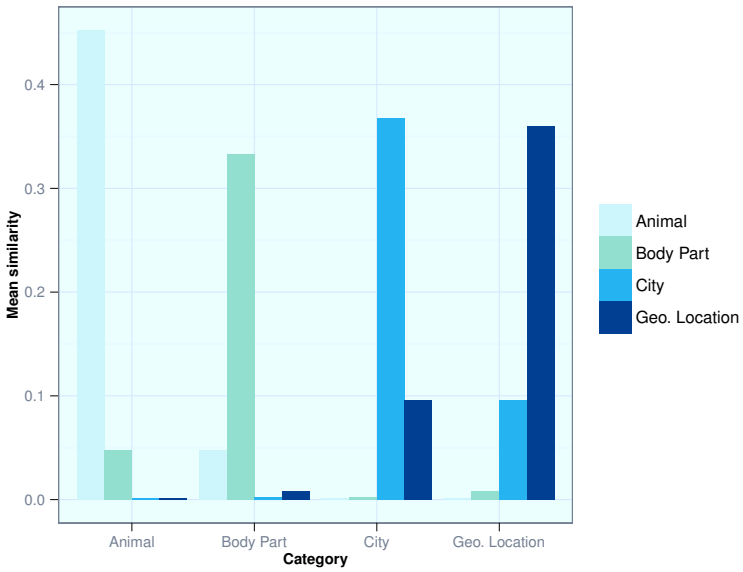

Figure 3.3: Mean similarity between items in the categories animals, body parts, geographical locations, and cities. Similarity between words from the same category are higher than those between words from different categories.

Rosch (1975) provides category norms for concepts from ten categories. These norms are used to provide stimuli for further exploration of the extent to which categorical information is captured by the vector representations. Note that Rosch's norms contain a range of category exemplars that includes exemplars that are both central to the category (e.g., AUTOMOBILE as a member of the category Vehicles) and peripheral to the category (e.g., ELEVATOR as a member of the category Vehicles). Thus, only the exemplars from each category that were ranked by subjects to be representative of the category are used. Specifically, the top ten exemplars from each category that do not appear in any other category and also appear in the model's dictionary are used as stimuli. These words are listed in Table A.1 in Appendix A. The results of an MDS performed on these stimuli are shown in Figure 3.4. As seen in the previous MDS, concepts from the same category tend to cluster together in the lower-dimensional space. While the clusters are not as distinct as in the previous MDS, which contained concepts from only four categories, given the number of concepts (100) and categories (10), and that the representations are projected from 300 to only two dimensions, resulting in a large loss

of information, the results are still quite good. The categories of tools, weapons, and toys seem to be especially dispersed among the other concepts. The categories of birds, sports, vehicles, fruits and vegetables produce particularly strong clustering. Some concepts, while grouped closest to a different category, are still highly similar to the concepts they clustered with. For example, WAGON is clustered with the Vehicles. While WAGON is considered to be a member of the category Toys in Rosch's norms, it seems reasonable that it should cluster with the other members of the category Vehicles, as it shares both properties and function with these items. The word KITE was categorized as a toy in the Rosch norms but was clustered with the birds in the MDS. However, the word KITE also refers to a family of birds of prey. This concept's positioning in the MDS results suggest that the bird meaning of KITE was better represented in the corpus from which the representations were constructed than the children's toy meaning.

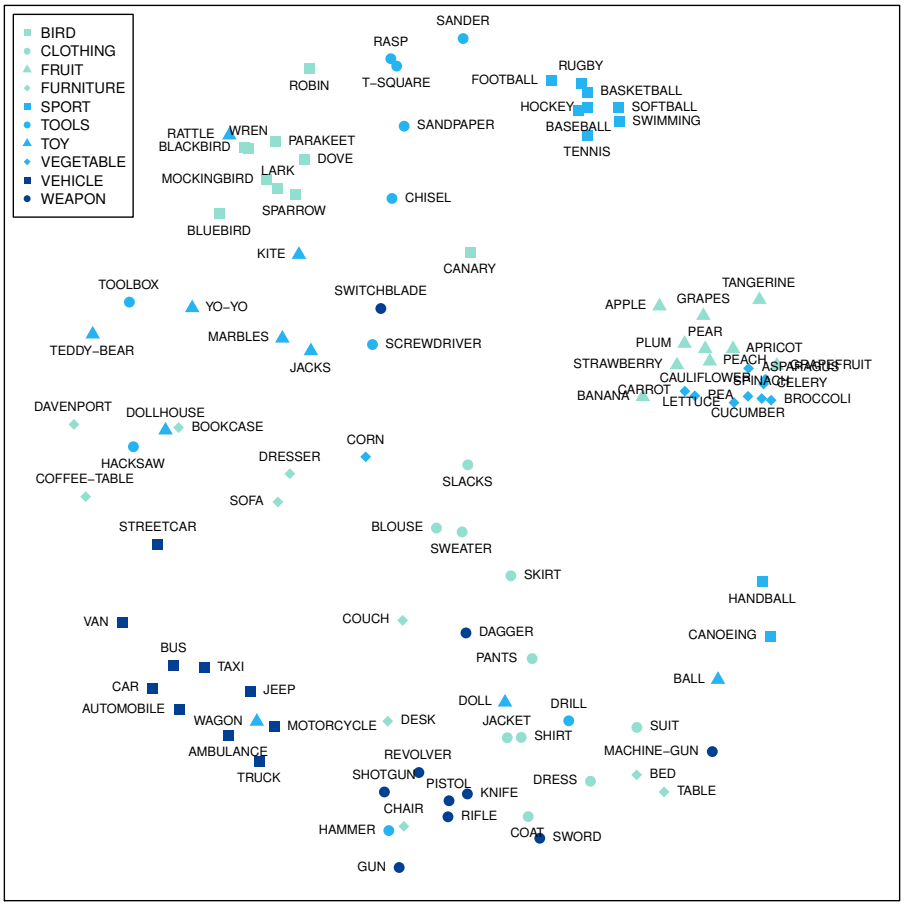

Figure 3.4: Multidimensional scaling of concepts from Rosch (1975) norms. Words from the same category tend to be located closer together in the plane. For example, fruits and vegetables are located in a large cluster on the right side of the plot. Within this cluster, vegetables are differentiated from fruits.

Figure 3.5 shows the mean similarity between concepts in each pair of categories. Each square in this figure represents the similarity between two categories, with white representing low similarity and darker colours representing higher similarity. The lower-left square, for example, represents the similarity of the category Bird with itself. The square to the right of this represents the similarity between the categories of Birds and Clothing. Note that each category is most similar to itself, as indicated by the darkly-shaded diagonal extending from the lower-left corner to the upper-right corner of Figure 3.5. Most categories have low similarity with all other categories, with the notable exception of Fruits and Vegetables, which are very similar categories.

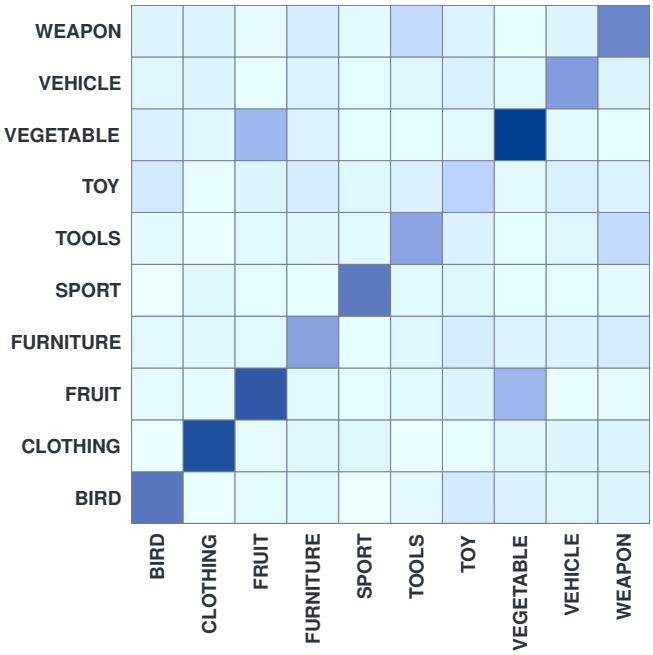

Figure 3.5: Mean similarity between items in each category from Rosch (1975) category norms, with darker colour indicating higher similarity. Note that mean similarity between words in the same category is higher than mean similar between words from different categories.

Table 3.1 shows the mean similarity of items within each category and the mean similarity between categories by category, as well as the results of ANOVAs performed for each category, comparing within- and between-category distances. All results were significant with $p < 0.001$, showing that items from the same category are more similar than items from different categories. A separate analysis comparing only the highly-similar categories of Fruits and Vegetables was performed. This revealed that Vegetables were more similar to one another, $M = 0.512$, $SD = 0.274$, than to Fruits, $M = 0.277$, $SD = 0.141$, $F(1,198) = 58.33$, $p < 0.001$. In addition, Fruits were more similar to one another,

$M = 0.364, SD = 0.246$, than to Vegetables, $M = 0.277$, $SD = 0.141$, $F(1,198) = 9.41$, $p = 0.002$. The category of Toys, whose items were the most dispersed in the MDS results, has the lowest similarity with itself ($M = 0.131$), while the category of Vegetables has the highest within-category similarity ($M = 0.512$).

	Mean similarity		
Category	Between	Within	F-Test
BIRD	0.0093	0.3296	921.45
CLOTHING	0.0233	0.4141	1195.70
FRUIT	0.0321	0.3639	579.97
FURNITURE	0.0206	0.1965	204.12
SPORT	0.0062	0.3750	1204.40
TOOLS	0.0153	0.2036	250.00
TOY	0.0283	0.1311	65.69
VEGETABLE	0.0334	0.5121	1118.50
VEHICLE	0.0174	0.2868	573.32
WEAPON	0.0300	0.3371	580.47

Table 3.1: Results of ANOVAs comparing within- and between-category distances for stimuli from Rosch (1975) norms. All tests were performed using (1, 998) degrees of freedom. All tests were significant at the 0.001 level.

These results demonstrate quite strongly that the representations produced by the model above have captured some notion of categorical information: concepts from the same category have more similar vectors than concepts from different categories.

3.2.2 Proper Names

Burgess and Conley (1998a, 1998b) examined how proper names are represented in HAL. They found differential representation of common and proper nouns (that is, common and proper nouns inhabited different areas of the high-dimensional semantic space) and posit that a denser semantic space for proper nouns than common nouns is a source of difficulty in proper noun retrieval (Cohen & Faulkner, 1986). In this section, the representation of proper names in the new model is explored. Figure 3.6 shows the results of an MDS performed on words that refer to common nouns and proper nouns that are typically male, typically female, or a surname. Words from each category clustered together in the plane. Further, within the common noun and surname categories, the items are further grouped into subcategories. The surnames show a distinct cluster of Spanish surnames. Within the category of common nouns most categories are concentrated in a small region in the plane. For example, fruits and vegetables are found in the upper right of the plot, while clothes are found near the lower right.

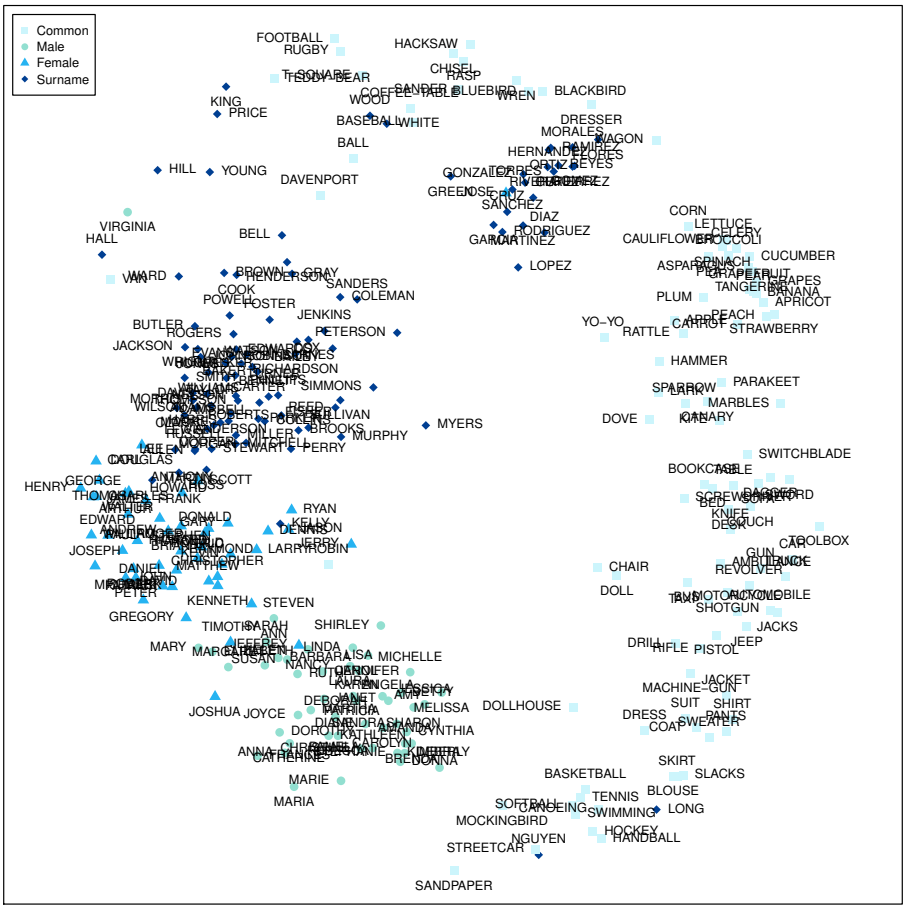

Figure 3.6: Multidimensional scaling of common nouns, male and female given names, and surnames. Common nouns were separated from proper nouns with higher accuracy, and female names, male names, and surnames each occupy a distinct region of the plane.

Figure 3.7 illustrates the mean similarity between each pair of noun categories. Each category shows highest similarity to itself. Each proper noun category showed higher similarity to the other proper noun categories than to common nouns. The category of proper nouns showed the lowest similarity to itself. However, it should be noted that items within this category came from a number of subcategories. As shown in Table 3.1, common nouns from different subcategories show very low similarity to one another. Since these intra-category comparisons were included in the means shown in Figure 3.7, it is expected that the mean similarity between common nouns would be lower than between pairs of, say, surnames, which all fall under a natural superordinate category. To further emphasize this point, consider whether a cucumber is more similar to a basketball or a person named Larry.

Table 3.2 shows the mean similarity between items within the same category and

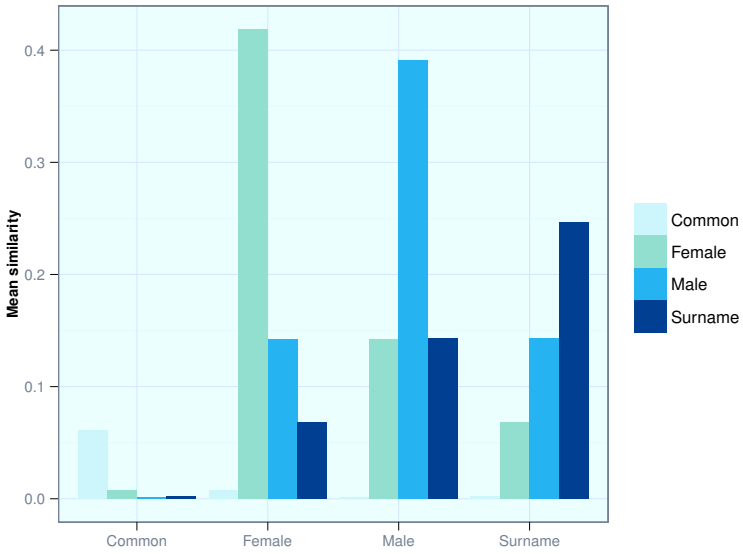

Figure 3.7: Mean similarity between a sample of common and proper nouns. Proper nouns were typically male, typically female, or surnames. Similarity between words from the same category are higher than those between words from different categories.

between different categories, as well as the results of ANOVAs comparing these means for each noun type. Note that due to the large number of degrees of freedom, a random sample of 50 items from the same category and 50 items from a different category was taken for each noun category and used in the ANOVAs. For example, for common nouns, the similarity between 50 pairs of common nouns and the similarity between 50 pairs in which one word is a common noun and the other word is a male, female, or surname were analyzed using ANOVA. In all comparisons, the similarity between items in the same category was higher than the similarity between items from different categories. This pattern is less pronounced for common nouns due to the issue described in the previous paragraph.

The data presented in this section suggest, as observed in HAL, that there is differential representation of common and proper nouns in the representations produced by the model of Chapter 2.

3.2.3 Parts of Speech

Burgess and Lund (1997a) examined the extent to which grammatical knowledge was encoded in the vectors produced by the HAL co-occurrence model by applying MDS to 35 words from four grammatical classes (nouns, verbs, determiners, and prepositions).

Noun Type	Mean similarity		F-Test	
	Between	Within		
Common	0.0033	0.0609	4.56	*
Proper (Male)	0.0861	0.3911	147.55	**
Proper (Female)	0.0584	0.4186	60.31	**
Proper (Surname)	0.0539	0.2461	24.73	**

Table 3.2: Results of ANOVAs comparing within- and between-category distances for common and proper nouns. All tests were performed using (1, 98) degrees of freedom. Tests marked with * were significant at the 0.05 level; tests marked with ** were significant at the 0.001 level.

They found that the vectors contained sufficient knowledge to identify the grammatical class of words and posit that the source of the knowledge is the substitutability of words from the same grammatical class in different contexts. In this section, MDS is used to explore whether the representations produced by the model encode grammatical information by examining how words from different part-of-speech categories are represented. The stimuli[10] included 88 adjectives, 76 nouns, 64 verbs, and 40 adverbs, and are listed in Tables B.1 through B.4 in Appendix B.

Although the results of the MDS, shown in Figure 3.8, are not as clear as those obtained using stimuli from different semantic categories, the words still have a tendency to cluster into groups based on part-of-speech. This is particularly evident for adverbs and nouns and less obvious for adjectives. The verb stimuli seem to be distributed relatively uniformly among the other items. An interesting observation is that one factor contributing to the moderate quality of the clustering results appears to be the semantic nature of the vectors. The goal of the model is to capture information related to the meaning of words. While part-of-speech information is certainly central to a word's identity, it is not particularly informative of the words meaning. Consider any grammatical sentence and replace all nouns with different nouns; the resulting sentence is still grammatical, though it may be non-sensical from a semantic perspective. As demonstrated, the vectors produced by the model capture a large quantity of semantic information. This information appears to interfere with any grammatical information contained within the vectors, as can be seen in Figure 3.8. The lower-left contains a large cluster of animals while the lower-right contains a small cluster of vehicles. In the lower-centre of the plot, there is a cluster of words that are largely sports-related. The verbs KICK, THROW, JUMP, and CATCH are clustered near the noun BALLS. The upper-right portion of the figure contains a cluster of cardinal numbers ranging from TWO to

[10]Items used in this experiment were taken from `http://www.k-3teacherresources.com/vocabulary_flashcards.html`

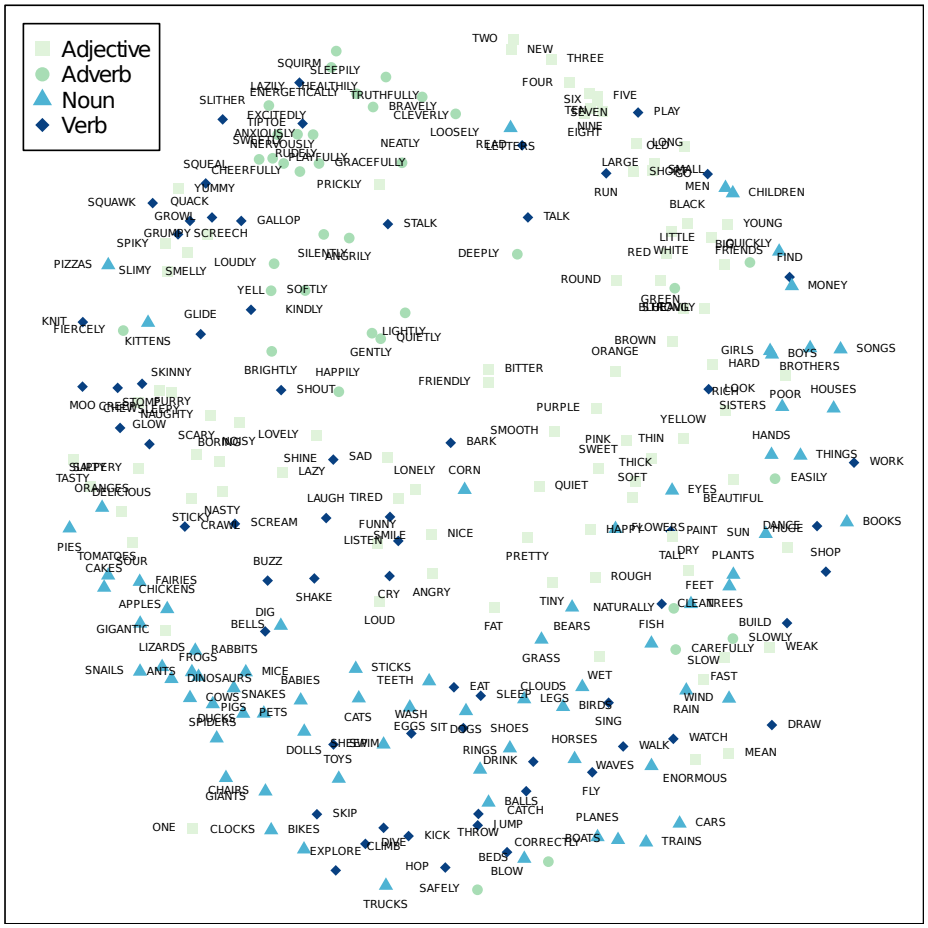

Figure 3.8: Results of MDS performed on words from four part-of-speech categories.

NINE. Interestingly, ONE was placed in the lower-left corner, far from the other cardinal numbers. Overall, it appears that there is some competition between the semantic and grammatical information contained within the vectors. An interesting exercise, which is delegated to the realm of future work, is to attempt to identify those components of the vector representations that best capture grammatical information.

3.2.4 Temporal Information and Relative Magnitudes

Louwerse et al. (2006) used MDS to explore the extent to which LSA (Landauer & Dumais, 1997) was able to capture information about the relative magnitudes of measurement-related words and time periods. By performing a one-dimensional MDS, a rank ordering of concepts could be obtained from the LSA vectors. This rank ordering matched closely with the natural orderings of the days of the week, the months, and words representing periods of time: the rank ordering produced by the model correlated significantly with

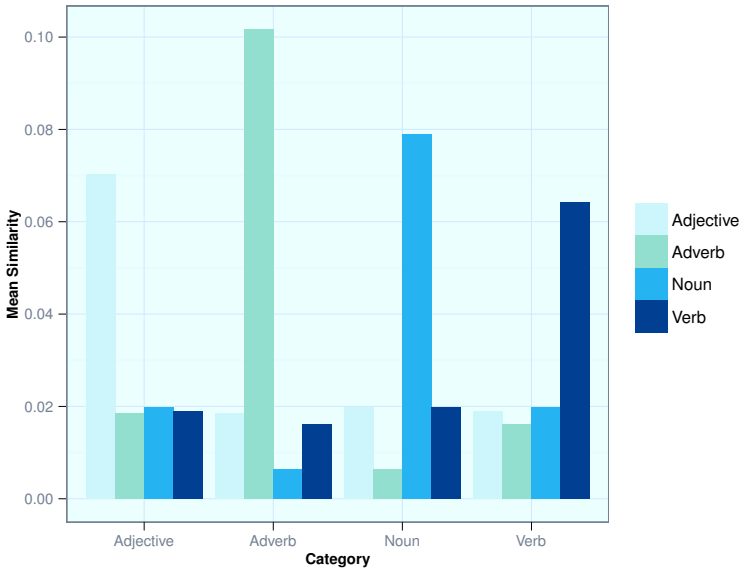

Figure 3.9: Mean similarity between items in each part of speech category. Similarity between words from the same part of speech are higher than those between words from different parts of speech.

the natural ordering for the days of the week, the months, and time periods.

The same holds true for the model describe in Chapter 2. When weekday names are projected into a single dimensional space, the model produces the ordering *Tuesday*, *Wednesday*, *Thursday*, *Monday*, *Friday*, *Saturday*, and *Sunday*, a close match to the correct order of the days of the week. Only *Monday* is misplaced and placed near *Friday*, a word that is a close associate due to the common phrase "Monday to Friday". Spearman correlation showed that the ordering produced by the model is similar to the natural ordering, $\rho(5) = 0.7568, p = 0.024$. A similar analysis performed on the names of the months showed a marginal relationship between the model's ordering and the natural ordering, $\rho(10) = 0.4406, p = 0.076$.

One-dimensional MDS was also used to determine if the model was able to capture relevant information about the relative magnitudes of different units of time. The vector representations of the modifiers AGO and LATER were combined with the vector representations of each of the words YEAR, MONTH, WEEK, DAY, HOUR, MINUTE, and SECOND by simply adding the two vectors together to produce 14 exemplars of time periods either in the past or future. The ranking produced by the model was strongly related to the natural ordering of these time periods (from furthest in the past to furthest in the future), $\rho(12) = 0.7538, p < 0.001$.

As an additional test of the model's ability to glean information about relative mag-

nitude from language usage, MDS was applied to metric units of distance (NANOMETERS, MICROMETERS, MILLIMETERS, CENTIMETERS, METERS, and KILOMETERS). The model was able to very accurately reproduce the natural ordering of the units of distance, as revealed using a Spearman correlation[11], $\rho(4) = -0.9429, p = 0.002$. When units of distance from the United States customary system, (INCHES, FEET, YARDS, and MILES) were added, the MDS results remain very strong, $\rho(8) = -0.8545, p < 0.001$.

These experiments suggest that the model is able to successfully order concepts by sequential or relative temporal information. In addition, the model was able to capture information about the relative magnitude of units of distance, even when the units were taken from different measurement systems.

3.3 Simulations of Behavioural Results

In this section, simulations of a number of psycholinguistic experiments are presented. These experiments tested subjects' performance in many language-related tasks and their simulation using the model of the previous chapter serves as demonstration that the representations produced by the model capture many characteristics of human semantic memory. Many of these experiments were used by Jones et al. (2006) as a means of comparing co-occurrence models.

3.3.1 Semantic and Associative Priming

Before the simulations are presented, a little background in necessary. Lexical decision (Meyer & Schvaneveldt, 1971) is an experimental paradigm commonly used to measure the effects of linguistic variables on performance in language-related tasks. In a typical lexical decision experiment, a subject is seated in front of a computer while a series of letter strings are displayed on the screen, one string at a time. The series of strings contains both English words and random strings of letters, called pseudo-words or non-words. In response to each letter string, the subject is asked to indicate, as quickly and accurately as possible, whether or not the letter string forms an English word by pressing an appropriate key on the keyboard. Both the accuracy of the response and the time elapsed between presentation of the letter string and the subject's response, referred to as the reaction time (RT), are recorded. Lexical decision has been used to observe many properties of language processing. For example, words that occur more frequently are recognized more quickly than rarely encountered words (Taft & Russell, 1992; Forster &

[11] Note that the coordinate system used by MDS is arbitrary, so the magnitude of the correlation is what is relevant, not the sign.

Chambers, 1973; Fredrickson & Kroll, 1976; Monsell et al., 1989; Whaley, 1978). As another example, words with multiple distinct meanings are recognized more slowly than words with only a single meaning, while words with multiple related meanings (called the *senses* of the word), are recognized more quickly than those with only a single unambiguous meaning (Klepousniotou, 2002; Rodd, 2004; Rodd, Gaskell, & Marslen-Wilson, 2002, 2004). This second example hints at the complex nature of the way word meaning is stored and accessed in the brain. Durda (2006) and Durda, Caron, and Buchanan (2010) showed that co-occurrence representations could be used to identify the various senses of a word and that ambiguity measures calculated from co-occurrence models predicted subjects' reaction time in lexical decision experiments.

A variation of the lexical decision task is the priming task. In a priming task, a letter string (called the "prime") is presented to the subject only briefly, then replaced by a second letter string (referred to as the "target"). The subject is asked to respond in the same manner as in a lexical decision task, but must only respond to the second string of letters. The relationship between the target and the prime modulates the subject's speed of response to the target. For example, if the prime and the target have similar visual forms, then subjects are faster to identify the target than if the prime is not visually similar to the target (S. Andrews, 1992; Sears, Siakaluk, Chow, & Buchanan, 2008); this effect is called orthographic priming. In many experiments, the presentation of the prime is so brief that the subject is unaware of its presence, yet the relationship between the prime-target pair still affects the subject's performance. A specific form of priming task is the semantic priming task. In this experiment, the relationship between the prime and the target is semantic in nature. That is, the relationship is based on word meaning. The pair may refer to words whose meanings are related through association (e.g., CAR-STREET), or through shared properties (e.g., CAR-TRUCK). The latter relationship is referred to as "semantic similarity." Many experiments have shown that stronger relationships between the prime and target lead to shorter reaction times from subjects than are produced in response to unrelated prime-target pairs. This effect is called a priming effect, and it's magnitude is measured by subtracting the mean response time for the related pairs from the mean response time for the unrelated pairs.

In the remainder of this section, simulations of several semantic priming experiments are presented. In each simulation, the similarity between each prime-target pair was calculated as the cosine between their corresponding vectors. An unrelated condition was simulated by using the same target words, but shuffling the primes so that no prime appeared with its original target. The magnitude of priming can then be calculated as the difference between the related condition and the unrelated condition.

Chiarello, Burgess, Richards and Pollok (1990) Chiarello et al. (1990) examined the effects of different types of relatedness on reaction time in a priming experiment. Their results demonstrated a priming effect for semantically related pairs and an increase in this effect for pairs that were both semantically and associatively related. No priming effect was found for pairs that were only associated and not semantically related. A simulation of this experiment produced similar, though not identical, results. The mean similarity between word pairs in each condition is shown in Table 3.3 and depicted graphically in Figure 3.10.

Pair Type	Related	Unrelated
Associated	0.2142 (0.1735)	0.0545 (0.1109)
Similar	0.2617 (0.0616)	0.0133 (0.0595)
Both	0.3518 (0.2147)	0.0196 (0.0778)

Table 3.3: Means and standard deviations (in parenthesis) from simulation of Chiarello et al. (1990)

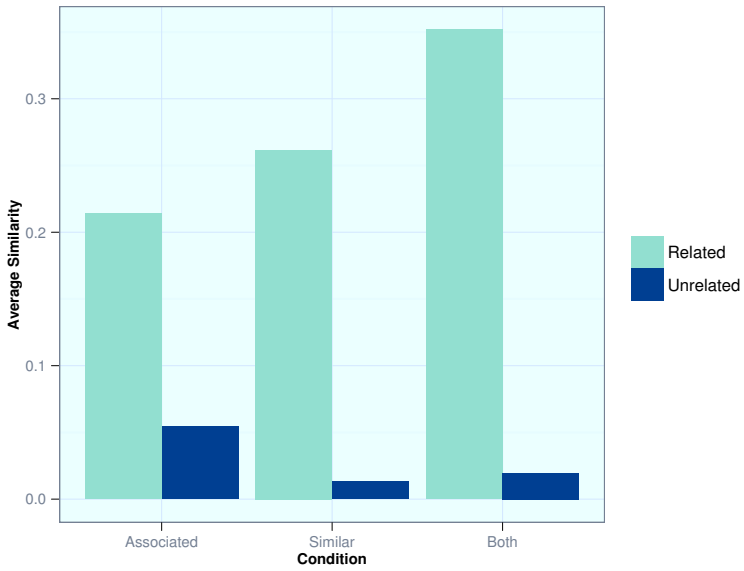

Figure 3.10: Results of simulation of Chiarello, et al. 1990. The results obtained by the model closely match the behavioural data.

The data were analyzed in a 2×3 between-subjects ANOVA. There was a main effect of prime type [related or unrelated], $F(1,282) = 212.41, p < 0.001$, and pair type [associated, similar, or both], $F(2,282) = 3.85, p = 0.022$, as well as an interaction between the two, $F(2,282) = 8.64$, $p < 0.001$. Priming was found in each condition of the simulation. For the associated only pairs, the mean difference in similarity was

$0.1598, t(47) = 5.66, p < 0.001$. The mean difference in the semantic only condition was $0.2484, t(47) = 9.85, p < 0.001$, and in the combined condition the difference was $0.3321, t(47) = 11.39, p < 0.001$. Although Chiarello et al. (1990) found no priming in the associative only condition, several other experiments have found a robust priming effect between associated pairs (see, for example, Ferrand and New (2003) in the next section). Thus, it seems reasonable that the model produces a priming effect for associated only pairs, despite the absence of this effect in the behavioural data. Post-hoc analysis with a Bonferroni correction revealed the source of the interaction: In the related condition, there was a difference between the mean similarity of associated only pairs and the combined pairs, $p < 0.001$, as well as between the semantically similar pairs and the combined pairs, $p = 0.035$. There was no difference between the associated only pairs and semantic pairs, and no differences were found between any of the pair types in the unrelated condition, all p's > 0.05. The general pattern of results found are similar to those found by Chiarello et al. (1990). Although there are some inconsistencies between the simulation and the behavioural data, these may be a result of the quality of stimuli used in the behavioural experiment.

Ferrand and New (2003) Ferrand and New (2003) performed a similar experiment; however, the combined condition was omitted and only associated and semantic pairs were used. Further, Ferrand and New were much more careful to control for association and semantic similarity. To select stimuli for the associated condition, pairs of words that were strongly associated in the Ferrand and Alario (1998) French association norms were rated for semantic similarity by subjects, and those with the lowest average similarity were retained as stimuli. For the semantic condition, pairs with low association in the Ferrand and Alario (1998) association norms were rated by subjects and those with the highest average similarity were used as stimuli. Although Ferrand and New (2003) used French stimuli, English translations were used in the simulation.

Ferrand and New observed a robust priming effect among both associated and semantic pairs. Semantic pairs produced a greater priming effect than associated pairs, but these conditions were not directly compared so there is no way to know if this difference was significant. Mean similarity and standard deviations for each condition in the simulation are given in Table 3.4 and are depicted graphically in Figure 3.11.

Analysis in a between-subjects ANOVA revealed main effects of pair type [related or unrelated], $F(1,158) = 73.68, p < 0.001$, and prime type [associated or semantic], $F(1,158) = 21.16$, $p < 0.001$, and a significant interaction, $F(1,158) = 14.06$, $p < 0.001$. Priming was found in both conditions. In the associated condition, the difference in

Pair Type	Related	Unrelated
Associated	0.1089 (0.1491)	0.0012 (0.0700)
Similar	0.2899 (0.2107)	0.0196 (0.0778)

Table 3.4: Means and standard deviations (in parenthesis) from simulation of Ferrand and New (2003)

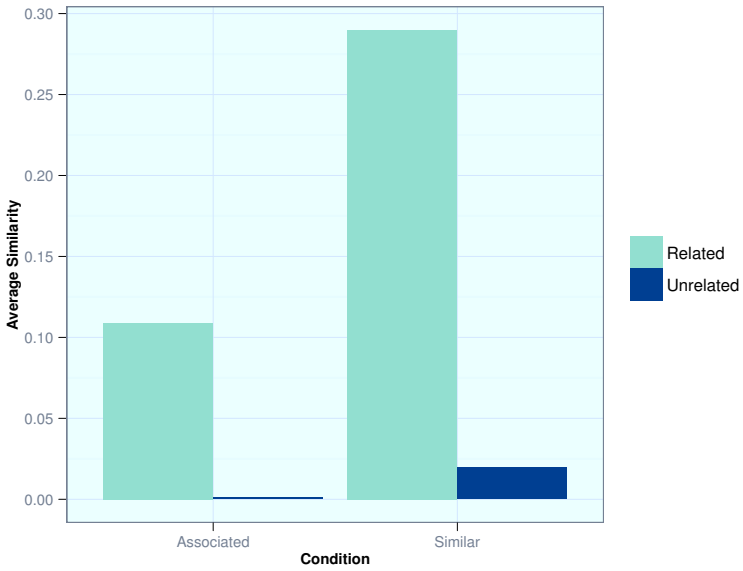

Figure 3.11: Results of simulation of Ferrand and New, 2003. The results obtained by the model closely match the behavioural data.

mean similarity between the related and unrelated pairs was 0.1077, $t(41) = 4.35$, $p < 0.001$. The difference between related and unrelated pairs in the semantic condition was 0.2703, $t(38) = 7.30$, $p < 0.001$. There was a difference in mean similarity between associated and semantic pairs in the unrelated condition, $p < 0.001$. No difference between pair types was found in the unrelated condition, $p > 0.05$. The simulation results match those found in the experimental results and suggest that the model has captured the subtle differences between associative relationships and semantic relationships.

Williams (1996) Williams (1996) compared the priming effects produced by four types of prime-target relationships: (1) semantically similar pairs (e.g., CAR-TRUCK), (2) category coordinates, which are words that are from the same category (e.g., BOWL-PLATE), (3) collocates, which are words that commonly occur together in conjunctive phrases (e.g., MILK-SUGAR, NIGHT-DAY), and (4) associates, which are pairs with high association strength but that do not appear in common phrases (e.g., CAR-STREET). Stimuli were presented both intact as black text against a white background, and in a visually

degraded condition where the stimuli were superimposed over a rectangle of random dots. The simulation performed here was only compared with the intact condition. Williams found shorter response times for related pairs than for unrelated pairs for all prime types. Pairs of collocates produced a greater priming effect than the other three types of pairs, among which there was no difference in effect size. Williams analyzed the collocate and associates data in a separate ANOVA and found an effect of relatedness but no significant interaction between relatedness and type of relationship. He concludes that, while there is no significant difference, the data are "certainly suggestive" (pp. 133).

The mean similarities and standard deviations from a simulation of the Williams (1996) experiment are provided in Table 3.5.

Pair Type	Related	Unrelated
Semantic	0.2463 (0.1456)	0.0222 (0.0797)
Category Coordinate	0.2812 (0.1739)	0.0076 (0.0569)
Collocates	0.4704 (0.2353)	0.0383 (0.1131)
Associates	0.2905 (0.2320)	0.0422 (0.0641)

Table 3.5: Means and standard deviations (in parenthesis) from simulation of Williams (1996)

Analysis of the data in a between-subjects ANOVA showed a main effect of pair type [related or unrelated], $F(1,132) = 129.41, p < 0.001$, and prime type [semantic, category coordinate, collocate, associative], $F(3,132) = 4.51, p < 0.01$, and an interaction, $F(3,132) = 3.33, p = 0.02$. Priming was found in each condition in the simulation. For the semantic pairs, the mean difference in similarity between related and unrelated pairs was 0.2483, $t(15) = 4.09$, $p < 0.001$. In the category coordinate condition, the mean difference between related and unrelated was $0.2242, t(21) = 5.85, p < 0.001$. In the collocate condition, this difference was $0.4321, t(15) = 7.13, p < 0.001$, and in the associated condition the difference was $0.2736, t(15) = 6.53, p < 0.001$. Post-hoc analysis with a Bonferroni correction revealed that, within the related condition, there was a difference between mean similarity for collocates and associates, $p = 0.026$, between collocates and category coordinates, $p = 0.014$, and between collocates and semantic pairs, $p < 0.001$. No differences were found between associated pairs, category coordinates, and semantically similar pairs in the related condition, all p's > 0.1. No differences were found among the different prime types in the unrelated condition, all p's > 0.1. An independent comparison of similarity between collocates and associated pairs was performed using an independent samples t-test. This analysis revealed a difference between the two conditions, $t(30) = -2.18, p = 0.037$, with collocates having higher average similarity than associated pairs that were not collocates.

The results of this simulation are consistent with the findings of Williams (1996) and suggest that the model is capturing the different types of similarity that may appear between pairs of words in natural language. It is particularly interesting to note that the behavioural data demonstrated a larger priming effect for pairs of words that frequently appear together as part of a phrase. By dint of its construction, the vector-based model used in the simulation captures relationships between words that are often used together in written (and presumably spoken) text. A successful simulation of Williams's experiment suggests that analyzing co-occurrence data collected from written text is a reasonable way to capture natural semantic information.

Moss, Ostrin, Tyler, and Marslen-Wilson (1995) Moss et al. (1995) explored the types of semantic information that are automatically retrieved when hearing a word in an auditory priming task (in their Experiment 1). Prime target pairs were chosen to have categorical or functional relationships. Type of semantic relationship was crossed with association strength (high or low association strength). Within the categorically related pairs, words were selected from both natural and artificial categories. Within the functionally related stimuli, pairs were selected to have instrument relations or script relations. This produced eight types of prime-target pairs, which are shown with examples in Table 3.6. Moss et al. observed priming for both categorical and functional relationships in both the presence and absence of an associative relationship. In addition, an associative boost was found, with associated pairs producing shorter reaction times than non-associated pairs.

	Associated	Non-associated
Category Coordinates		
Natural	THUNDER-LIGHTNING	COW-GOAT
Artifact	BAT-BALL	KITE-BALLOON
Functional Relations		
Script	BEACH-SAND	CASTLE-DUNGEON
Instrumental	BELT-TROUSERS	BROOM-FLOOR

Table 3.6: Examples of prime-target pairs for each condition in Experiment 1 of Moss et al. (1995).

The results of a simulation of this experiment are shown in Table 3.7. Priming was found for both category coordinates, $t(109) = 11.284, p < 0.001$, and functional relations, $t(109) = 5.485$, $p < 0.001$. For category coordinates, the mean similarity between pairs in natural categories, $M = 0.4619, SD = 0.2063$, was higher than the mean similarity between pairs from artificial categories, $M = 0.2763$, $SD = 0.1680$, $t(53) = 3.651$,

$p < 0.001$. In the functional relation condition, there was no difference between pairs with a script relation, $M = 0.2070, SD = 0.1604$, and those with an instrumental relation, $M = 0.1670, SD = 0.1466, t(54) = 0.9733, p = 0.33$. Although similarity between associated pairs is slightly higher than similarity between non-associated pairs, this difference was not significant for either category coordinates, $t(53) = 1.276$, $p = 0.21$, or functional relations, $t(54) = 0.5843, p = 0.56$. In summary, the model was able to simulate priming in all conditions, but did not reproduce the associative boost observed in the behavioural data (the model did, however, demonstrate an associative boost in the simulation of Chiarello et al. (1990), presented earlier in this section).

		Category coordinates		Functional Relation	
		Natural	Artifact	Script	Instrumental
Assoc.	Rel.	0.5169 (0.22)	0.2889 (0.17)	0.2198 (0.13)	0.1783 (0.16)
	Unrel.	0.0083 (0.09)	0.0549 (0.03)	0.0489 (0.10)	0.0973 (0.12)
Non-assoc.	Rel.	0.4068 (0.18)	0.2645 (0.17)	0.1942 (0.19)	0.1558 (0.13)
	Unrel.	0.0280 (0.10)	0.0219 (0.06)	0.0531 (0.11)	0.0004 (0.04)

Table 3.7: Means and standard deviations (in parenthesis) from simulation of Balota and Lorch (1986).

3.3.2 Mediated Priming

In a mediated priming task, the prime and target pair are not directly related, but are instead related through a third concept to which both are related. For example, the prime-target pair STRIPES-LION are related through the concept TIGER. The effects of mediated relationships on subjects' performance are more subtle than the effects of direct semantic and associative relationships.

Balota and Lorch (1986) Balota and Lorch (1986) compared the priming effects produced by pairs of words that are directly related to those produced by pairs of words that are related indirectly through some other concept. Stimuli consisted of word triads in which the first and second word were directly related, the second and third word were directly related, but the first and third words were related only through their common relationship to the second word (e.g., COAL-BLACK-WHITE). An unrelated condition was also included, in which the the targets were paired with primes from a different triad. Rather than make a lexical decision, subjects were asked to read the target word aloud as quickly as possible. Balota and Lorch found an advantage for mediated pairs over unrelated pairs, and for related pairs over mediated pairs.

The mean and standard deviations from a simulation of this experiment are shown in Table 3.8. A one-way between-subjects ANOVA showed a main effect of pair type, $F(2,141) = 11.79$, $p < 0.001$. Post-hoc analysis using a Bonferroni correction revealed higher similarity between related pairs than unrelated pairs, $p < 0.001$, and higher similarity between mediated pairs than unrelated pairs, $p = 0.039$. Further, related pairs were more similar than mediated pairs, $p = 0.062$. The results of this simulation mirror those found in the behavioural data.

Pair Type	Similarity
Related	0.1591 (0.1654)
Mediated	0.0972 (0.1315)
Unrelated	0.0307 (0.0756)

Table 3.8: Means and standard deviations (in parenthesis) from simulation of Balota and Lorch (1986).

McNamara and Altariba (1988) McNamara and Altarriba (1988) explored mediated priming effects using stimuli derived from the Balota and Lorch (1986) stimuli set. The mediator from each stimulus item in the Balota and Lorch data set was used as the target word, and a new word that was related only to the original prime but not to the mediator or the original target was used as the prime. For example, for the triad LION-TIGER-STRIPES, TIGER was taken as the new target and MANE, which is related to LION but not to TIGER or STRIPES, was used as the prime. Subjects were presented with the prime and target simultaneously and asked to make a lexical decision to the pair, producing a positive response only if both the prime and target were English words. The results obtained were consistent with those of Balota and Lorch's naming experiment: mediated pairs produced shorter response times than unrelated pairs.

In a simulation of this experiment, the mean similarity between mediated pairs was 0.1040 ($SD = 0.1373$); the mean similarity for unrelated pairs was 0.0026 ($SD = 0.0615$). A paired-samples t-test showed a difference between conditions, $t(32) = 4.24$, $p < 0.001$. These results agree with those observed by McNamara and Altarriba (1988).

McNamara (1992) McNamara (1992) examined long-distance mediated priming effects. Subjects made lexical decisions in response to sequentially presented prime-target pairs in which the prime and target were related by a chain of two concepts. The prime-target pairs were based on the McNamara and Altarriba (1988) stimuli, which in turn were based on the stimuli from Balota and Lorch (1986). The original targets used by

Balota and Lorch served as the target words in this experiment, and the primes were provided by the additional words added for the McNamara and Altarriba (1988) experiment. For example, from the four-word sequence MANE-LION-TIGER-STRIPES, the prime-target pair MANE-STRIPES was used. Although the relationship between the prime and the target appears to be weak, McNamara observed a reliable 10 ms advantage for mediated pairs over unrelated pairs.

The results of this experiment were reproduced by the simulation: the similarity between mediated pairs ($M = 0.0662, SD = 0.1047$) was higher than between unrelated pairs ($M = 0.0181$, $SD = 0.0830$), and a paired-sampled t-test confirmed that there was a true difference between similarity in the two conditions, $t(32) = 2.54, p = 0.016$. Given the weak relationships between the primes and targets, and the subtlety of the effect in the subject data, it is surprising that the model is able to accurately reproduce the results of this experiment. A successful simulation demonstrates that the model is able to capture finely-grained differences in word similarity.

de Groot (1983) In an experiment similar to that of Balota and Lorch (1986), de Groot (1983) examined priming effects for prime-target pairs mediated by a single concept in a naming experiment. The prime-target pairs were constructed from Dutch association norms (de Groot, 1980). Priming was found for both related and mediated pairs. Related pairs produced shorter response times than mediated pairs, which, in turn, produced shorter response times than unrelated pairs. de Groot's stimuli were presented in Dutch; English translations are used in the simulation.

The means and standard deviations from the simulation are shown in Table 3.9. The data were analyzed using a one-way ANOVA, which showed a main effect of pair-type, $F(2,81) = 24.25$, $p < 0.001$. The mean difference between similarity for related pairs and unrelated pairs was $0.2350, t(55) = 6.44, p < 0.001$, and the difference between mediated and unrelated pairs was $0.0882, t(52) = 3.48, p = 0.001$. Similarity between related pairs was also higher than similarity between mediated pairs, with a mean difference of $0.1468, t(55) = 3.75, p < 0.001$. Again, the results of the simulation agree with the results observed in the subject data.

Pair Type	Similarity
Related	0.2525 (0.1761)
Mediated	0.0882 (0.1092)
Unrelated	0.0175 (0.0734)

Table 3.9: Means and standard deviations (in parenthesis) from simulation of de Groot (1983).

McKoon and Ratcliff (1992) In contrast to the spreading activation theory advanced by Collins and Loftus (1975), McKoon and Ratcliff (1992) argue that semantic priming effects occur not because of the spread of activation through mediating links between the prime and the target, but because the prime and the target form a compound cue that arises from the simultaneous presence of both the prime and target in short-term memory. The magnitude of the priming effect is mediated by the familiarity of the particular compound formed by the prime and target. In their Experiment 3, McKoon and Ratcliff used co-occurrence frequency within a six-word window in the six-million word Associated Press news-wire corpus to estimate the familiarity of prime-target compound cues. Each target word was matched with four primes. Two primes were selected based on the familiarity estimates: one prime was selected to have a high probability of co-occurrence with the target (called the *high-t* condition), and the other was selected to have a low, but higher than chance, probability of occurring with the target (called the *low-t* condition). In addition, a third prime known to have a strong association with the target was selected from published free-association norms. Finally, each target word was paired with an unrelated prime to produce a total of four prime-target pairs for each target word. The behavioural data revealed an effect of prime type, with free-association primes producing the shortest response times, followed by high-t primes, low-t primes, and unrelated primes producing the longest response times. Both free-association and high-t primes produced shorter response times than the unrelated condition. No difference was found between response times in the low-t and unrelated conditions.

Table 3.10 shows the means and standard deviations of the prime-target similarities in each condition produced by a simulation of the McKoon and Ratcliff (1992) experiment. Analysis in a one-way ANOVA showed an effect of prime type on mean similarity, $F(3,132) = 17.283$, $p < 0.001$. Post-hoc analysis using a Bonferroni correction showed a difference between similarity for free-association primes and for unrelated primes, $p < 0.001$, as well as both high-t primes, $p < 0.001$ and low-t primes, $p < 0.001$. There was a reliable difference between similarity for high-t primes and unrelated primes, $p = 0.079$, but no difference was found between the high-t and low-t conditions, $p = 1.00$. No difference was found between low-t and unrelated primes, $p = 0.254$. Once again, the results of the simulation replicate those found in the behavioural data.

3.3.3 Category Norms

The last test of the model's ability to obtain semantic knowledge from text uses the category norms of Rosch (1975), which were previously used in the MDS experiments in

Prime Type	Similarity
Associated	0.3056 (0.2078)
High-t	0.1405 (0.1657)
Low-t	0.1273 (0.1777)
Unrelated	0.0414 (0.0928)

Table 3.10: Means and standard deviations (in parenthesis) from simulation of McKoon and Ratcliff (1992), Experiment 3.

Section 3.2. Here, the norms are used to measure how well the model is able to identify the typicality of a category exemplar. As mentioned before, the Rosch norms include both prototypical exemplars of a category, such as CHAIR as an exemplar of the category FURNITURE, and those that are more peripheral to the category, such as TELEPHONE as FURNITURE. Rosch provides rankings of several exemplars for each of ten categories, as ranked by subjects.

To determine how well the model captured the graded nature of typicality, the similarity between each exemplar in Rosch's norms and the name of the category was calculated. These were then ranked from highest similarity to lowest. Table 3.11 shows the correlations between the rankings in Rosch's norms and those produced by the model.

Category	Correlation	df	p	
BIRD	-0.2900	44	0.025	**
CLOTHING	-0.4502	48	< 0.001	**
FRUIT	-0.6719	38	< 0.001	**
FURNITURE	0.0040	45	0.489	
SPORT	-0.5147	48	< 0.001	**
TOOLS	-0.1832	52	0.090	*
TOY	-0.1508	43	0.161	
VEGETABLE	-0.3556	40	0.010	**
VEHICLE	-0.4934	44	< 0.001	**
WEAPON	-0.6895	55	< 0.001	**

Table 3.11: Spearman correlations between vector similarity and subject ranking of exemplar typicality. Marginal correlations are marked with * and significant correlations are marked with **.

Correlations between subject rankings and similarity between the exemplars and their category were found in all categories except for Toys and Furniture. This echoes the results observed in Figure 3.4 on page 30. In the MDS results based on the Rosch norms, concepts from the category Toys were dispersed through the plane and did not form a cohesive group as the concepts from the other categories did. As mentioned there, members of the category Toys can be exemplars of nearly any category (for example, a

KITE is also a Bird and a WAGON is also a Vehicle). The absence of a relationship between the model's ranking and the Rosch norms for the category Furniture may be due to the binary nature of inclusion in this category. The categories of Weapons and Fruit produced the strongest relationship with the subject norms. This is consistent with the results shown in Figure 3.4, where items from both of these categories grouped together closely in the plot.

3.4 Discussion

The experiments above demonstrate that the representations produced by the model described in Chapter 2 can simulate many aspects of human semantic memory. Through MDS, categorical information was revealed to exist in the vectors. This technique also showed that the vectors contain part-of-speech information, although there is interference from the semantic information contained in the vectors. Categorical information was also revealed through comparison with subjects' typicality ratings of category exemplars: similarity between a category and its exemplars moderately correlated with typicality ratings produced by subjects. The results of several priming experiments were reliably reproduced by the model.

Note that the goal of the demonstrations provided in this chapter is not to show that the model described in Chapter 2 is superior to existing models, but rather to show that the vectors produced by the model capture properties of human semantic memory and are not strongly influenced by the frequency of the words in the input corpus. Jones et al. (2006) provides a comparison of HAL, LSA, and BEAGLE on a battery of tests that overlaps with the experiments simulated above and found that BEAGLE was best able to reproduce the pattern of performance shown by subjects in language-related tasks. The performance of the model presented in this dissertation is similar to that of BEAGLE.

4 Identifying Features in Co-occurrence Representations

This chapter describes a neural network model that is trained to produce a list of the physical and behavioural properties that describe an object in response to an abstract semantic vector representing the concept. The network contained three layers of processing units with one layer of $\ell = 300$ input units, one layer of $n = 824$ output units, and one layer of m hidden units. The number of hidden units was determined experimentally using a process that is described later in this chapter. Each output unit denoted a single "feature primitive" that a concept may or may not posses. These feature primitives described physical and behavioural properties, such as ⟨has 4 legs⟩ and ⟨made of metal⟩, of both living and non-living things and were taken from the feature-production norms of McRae et al. (2005). The inputs to the network were a set of semantic vectors derived from the co-occurrence vectors described in Chapter 2. The network was trained to activate the correct combination of features on the output units in response to a semantic representation presented on the input units. This was achieved using the backpropagation learning algorithm (Rumelhart, Hinton, & Williams, 1986). The remainder of this chapter provides further detail about the structure of the network, the input and output patterns that were used for training the network, results demonstrating that the network was able to produce the desired mapping from input to output vectors, and the results of experiments demonstrating the ability of the network to generalize this mapping to novel inputs.

4.1 Network Structure

The network contained three layers of processing units: one layer of $\ell = 300$ input units, one layer of m hidden units[12], and one layer of $n = 824$ output units. The layers of processing units were fully interconnected; each input unit sent its output to every hidden unit, and each hidden unit sent its output to every output unit. An additional "bias" input was included in the network. This unit's activation was always set to +1 and was passed to every processing unit in the hidden and output layers. Figure 4.1 below shows the structure of the network.

[12]The number of hidden units, as well as the learning rate and momentum parameters of the network, were determined experimentally as described in Section 4.6.

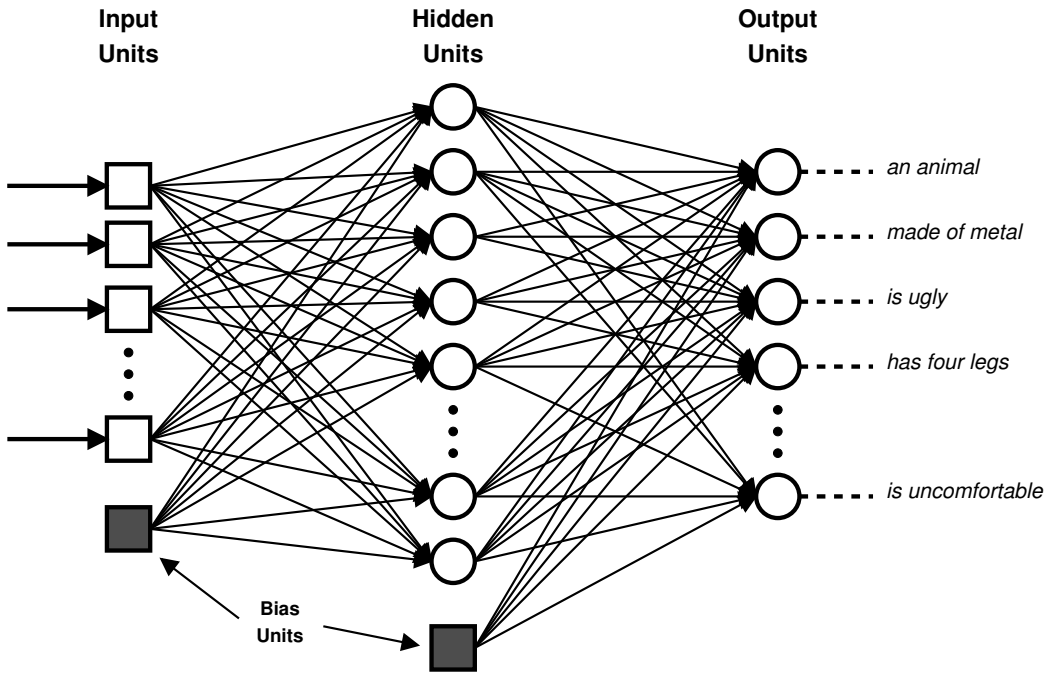

Figure 4.1: The neural network used in the simulations in Chapter 4. The bias units are shown as shaded squares. The network contained 300 input units, 4000 hidden units (as described in Section 4.6), and 824 output units, each corresponding to a feature or property that a concept may possess.

4.2 Training and Testing Data

The inputs provided to the network were 300-component bipolar vectors (i.e., each component was either -1 or $+1$). These vectors were derived from the co-occurrence vectors described in Chapter 2 by setting each component of the bipolar vector to $+1$ (active) if the corresponding component of the co-occurrence vector is positive and setting all other components of the bipolar vector to -1 (inactive). The average number of units active for a concept was 151.11 ($SD = 8.29$), or 50.37%. The minimum number of units active for a concept was 127 for the concept CHURCH and the maximum was 180 for the concept ANT.

The output patterns to be learned by the network were binary vectors constructed from the feature-production norms of McRae et al. (2005). McRae et al. provided subjects with lists of words and instructed the subjects to produce lists of properties describing each concept. These responses were then used to determine the probability that a subject would produce each property in response to a particular concept. From the complete list of 2,526 features, the resulting norms contained 824 features that were produced by subjects for two or more concepts[13]. For each of these features, an output

[13]Note that the each feature included in the network was produced in response to two or more of the

unit representing the feature was included in the network. While the McRae et al. norms provide probabilities that a feature is produced in response to a concept, the output patterns were simple binary vectors. To create the output patterns, each output unit that corresponded to a feature produced by one or more subjects was set to +1 (active). All other output units were set to 0 (inactive). On average, only 10.71 ($SD = 3.42$), or 1.30%, of the features were active. In comparison to the input vectors, in which 50% of the units were active on average, the output vectors are very sparse representations. The highest number of features was 23 for the concept LION, and there were 20 concepts with only 5 features. Although there is a correlation between the number of active input and output units ($r(463) = -0.10, p = 0.013$), the number of active input units accounts for less than 1.1% ($r^2 = 0.0107$) of the variance in the number of active output units. Figure 4.2 shows that there is no clear linear relationship between the two variables, indicating that the significance of the correlation is due to the large number of degrees of freedom.

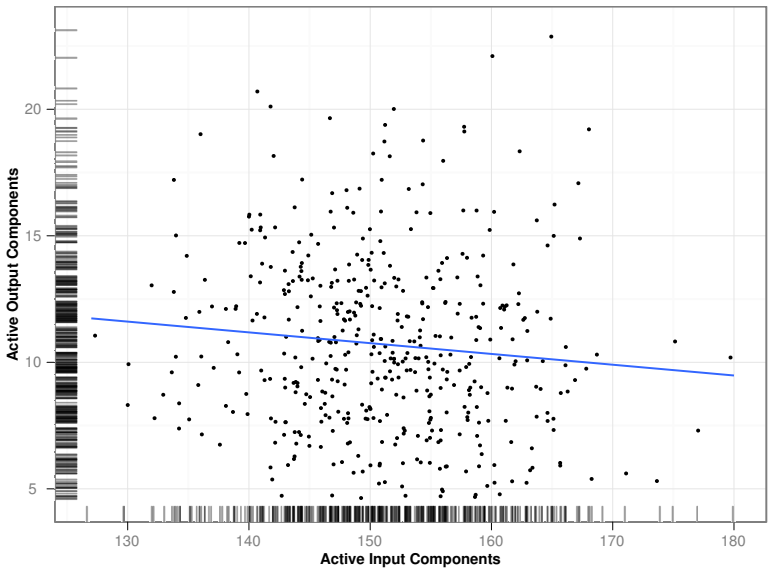

Figure 4.2: A scatter plot showing the relationship between number of active input units and number of active output units. Marginal distributions are shown along the axes.

From the McRae et al. norms, only those concepts that possessed five or more of the 824 features included in the network and that also appeared in the dictionary of the co-occurrence model of Chapter 2 were used in the experiments of this chapter. In total, 465 such words were found. To allow sufficient training data while allowing a large

concepts in the full set of norms; when the data set is restricted to only those concepts that appear in both the feature production norms and the vocabulary of the model developed in Chapter 2, some features are associated with only a single concept.

number of the available words to be used to test the network's performance, these words were split into ten training/testing pairs. Each training data set contained 445 words and the remaining 20 words were set aside for testing the network's ability to generalize. For the network to produce the best results on the testing items, the distribution of features between the testing and training data must be as similar as possible. Because many features occurred with few concepts, the 20 items used for testing in each data set needed to be chosen carefully. A genetic algorithm (Holland, 1992) was used to select a set of 20 items so that the distribution of features in this set most closely matched that found on the 445 training items[14]. The quality of the fit was measured by Kulback-Leibler divergence (Kullback & Leibler, 1951; a measure of the distance between two probability distributions). This process was then repeated to select a second set of 20 items, with the further restriction that no item that appeared in the first set could also appear in the second. This process was repeated until ten training/testing pairs were created with each test set excluding the words from all prior test sets. This resulted in a total test set containing 200 items spread across 10 testing sets.

In addition to the training and testing sets, an additional set of testing items was created. These items are referred to as randomized test items. For each training-testing set pair, the randomized items were the same as those used for testing. However, the components of the input patterns were shuffled randomly (a different random ordering was used to create each vector). The input vectors in these randomized patterns follow the same distribution of values as the original input vectors, but any relationship between specific components of the vectors has been removed, removing any regularities existing in the vectors. Thus, a failure of the network to be able to reliably produce features from these randomized patterns suggests that the network is exploiting regularities in the structure of the input vectors to identify features.

[14]A genetic algorithm is a heuristic search method that seeks a near-optimal solution to an optimization problem using methods inspired by the processes of natural selection and evolution. Initially, a large population of potential solutions to the problem are randomly generated and the quality of each of these solutions is assessed by a fitness function (for example, if the problem is to minimize some positive-valued multivariate function, the fitness function could simply be the value of the function evaluated at the potential solution). New solutions are generated by randomly selecting two parent solutions and combining their parameters (for example, by taking linear combinations of the parent solutions' parameters). The parameters of the newly generated solution may be changed by a small random amount. This process is repeated until the algorithm terminates after some specified number of solutions have been generated, or until no further increase in the quality of the solutions is observed.

4.3 Network Dynamics

Upon presentation of an input vector to the network, activation flowed from the input units (including the bias unit) through the weighted connections to the hidden units, where these weighted inputs were summed to produce the net input to each of the hidden units. The activation of each hidden unit was then calculated using the bipolar sigmoid function, given in (4.2). Before stating this more precisely, some notation is introduced. Let $\boldsymbol{x} \in \mathbb{R}^\ell$ be the input vector presented to the network, $\boldsymbol{\eta}^h \in \mathbb{R}^m$ be the vector of net inputs to the hidden units, and $\boldsymbol{y} \in \mathbb{R}^m$ be the vector of activations of the hidden units. Let $\mathsf{W}_1 = \left(w^1_{ji}\right)$ be the $m \times \ell$ matrix of weights connecting the input units to the hidden units, where w^1_{ji} is the weight of the connection from the i^{th} input unit to the j^{th} hidden unit, and let $\boldsymbol{\beta}^h \in \mathbb{R}^m$ be the vector of weights from the bias unit to each hidden unit, where β^h_j is the weight of the connection from the bias unit to the j^{th} hidden unit. Then the net input to the hidden units, denoted $\boldsymbol{\eta}^h \in \mathbb{R}^m$, is given by

$$\boldsymbol{\eta}^h = \mathsf{W}_1 \boldsymbol{x} + \boldsymbol{\beta}^h. \tag{4.1}$$

The activation of each output unit is calculated by applying the bipolar sigmoid function,
$g : \mathbb{R} \to [-1,1]$, given by

$$g(\eta) = \frac{2}{1+\exp(-\eta)} - 1, \tag{4.2}$$

to its net input, η. That is, $y_i = g(\eta^h_i)$ for $i = 1,2,\ldots,m$. For convenience, the notation

$$\boldsymbol{y} = g\left(\boldsymbol{\eta}^h\right) = \left(g\left(\eta^h_1\right), g\left(\eta^h_2\right), \ldots, g\left(\eta^h_m\right)\right)^\top$$

is used.

The activation of the output units was calculated similarly. Let $\boldsymbol{\eta}^o \in \mathbb{R}^n$ be the net input to the output units and $\boldsymbol{z} \in \mathbb{R}^n$ be the activation of the output units. Let $\mathsf{W}_2 = \left(w^2_{kj}\right)$ be the $n \times m$ matrix of weights connecting the hidden units to the output units, where w^2_{kj} is the weight of the connection from the j^{th} hidden unit to the k^{th} output unit, and let $\boldsymbol{\beta}^o \in \mathbb{R}^n$ be the vector of weights from the bias unit to the hidden units, where β^o_k is the weight on the connection from the bias unit to the k^{th} output unit. Analogous to (4.1), the net input to the output units is given by

$$\boldsymbol{\eta}^o = \mathsf{W}_2 \boldsymbol{y} + \boldsymbol{\beta}^o. \tag{4.3}$$

The activation of each output unit is calculated using the binary sigmoid function,

$f : \mathbb{R} \to [0,1]$, defined by

$$f(\eta) = \frac{1}{1+\exp(-\eta)}, \tag{4.4}$$

where η is the net input to the unit. Again, the notation

$$\boldsymbol{z} = f(\boldsymbol{\eta}^o) = \left(f\left(\eta_1^o\right), f\left(\eta_2^o\right), \ldots, f\left(\eta_m^o\right)\right)^\top \tag{4.5}$$

is used for convenience. The vector $\boldsymbol{z}$ is the network's output in response to the input vector $\boldsymbol{x}$.

Note that the output of the network can be written as a single equation:

$$\boldsymbol{z} = f\left(\mathsf{W}_2\left(g\left(\mathsf{W}_1\boldsymbol{x} + \boldsymbol{\beta}^h\right)\right) + \boldsymbol{\beta}^o\right). \tag{4.6}$$

The notation $\boldsymbol{z} = \mathfrak{N}(\boldsymbol{x})$ is used to denote the output, $\boldsymbol{z}$, produced by the network in response to the input vector $\boldsymbol{x}$.

Error on the output units was calculated using the cross-entropy error function. This error function is suitable for use with binary representations and produces larger error signals during training, potentially reducing the number of iterations required for the network to learn the training items. Let $\boldsymbol{t}$ be the binary output pattern to be learned in response to some bipolar input vector $\boldsymbol{x}$ and let $\boldsymbol{z}$ be the binary output actually produced by the network in response to $\boldsymbol{x}$. That is, $\boldsymbol{z} = \mathfrak{N}(\boldsymbol{x})$. Then the cross-entropy error for the pattern is given by

$$E(\boldsymbol{t}, \boldsymbol{z}) = -\sum_{i=1}^{n} t_i \log z_i + (1 - t_i)\log(1 - z_i). \tag{4.7}$$

Let $\boldsymbol{T} = \{(\boldsymbol{s}_1, \boldsymbol{t}_1), (\boldsymbol{s}_2, \boldsymbol{t}_2), \ldots, (\boldsymbol{s}_\tau, \boldsymbol{t}_\tau)\}$, where τ is the number of observations, be the set of all pairs of input and output observations used for training the network. The total error over the set of all input patterns is given by

$$E(\boldsymbol{T}) = \sum_{(\boldsymbol{s},\boldsymbol{t}) \in \boldsymbol{T}} E(\boldsymbol{t}, \mathfrak{N}(\boldsymbol{s})).$$

This is the quantity to be minimized during the network training process.

4.4 The Backpropagation Algorithm

The backpropagation algorithm (Rumelhart et al., 1986) is an iterative algorithm that minimizes $E(\boldsymbol{T})$ by using a gradient descent[15] procedure to adjust the parameters W_1, W_2, $\boldsymbol{\beta}^h$, and $\boldsymbol{\beta}^o$. This is achieved by taking a small step in the direction of the negative of the gradient of the error with respect to the parameters $\mathsf{W}_1, \mathsf{W}_2, \boldsymbol{\beta}^h$, and $\boldsymbol{\beta}^o$. The size of the step is controlled by the *learning rate* parameter of the backpropagation algorithm, denoted α. Consider a single training observation, $(\boldsymbol{s}, \boldsymbol{t}) \in \boldsymbol{T}$, and let $\boldsymbol{z} = \mathfrak{N}(\boldsymbol{s})$ be the output of the network in response to input vector $\boldsymbol{s}$. The value of $E(\boldsymbol{t}, \boldsymbol{z})$ depends on some element w^2_{kj} of W_2 only through the value of y_j. We have

$$\frac{\partial E}{\partial w^2_{kj}} = \sum_{\xi=1}^{n} \frac{\partial E}{\partial \eta^o_\xi} \frac{\partial \eta^o_\xi}{\partial w^2_{kj}} = \frac{\partial E}{\partial \eta^o_k} \frac{\partial \eta^o_k}{\partial w^2_{kj}} \tag{4.8}$$

since $\partial \eta^o_\xi / \partial w^2_{kj} = 0$ for any $\xi \neq k$. Let

$$\delta^o_k = \frac{\partial E}{\partial \eta^o_k}. \tag{4.9}$$

Then (4.8) can be written as

$$\frac{\partial E}{\partial w^2_{kj}} = \delta^o_k \frac{\partial \eta^o_k}{\partial w^2_{kj}}. \tag{4.10}$$

Now, δ^o_k can be written as

$$\delta^o_k = \frac{\partial E}{\partial z_k} \frac{\partial z_k}{\partial \eta^o_k}. \tag{4.11}$$

[15] Gradient descent is an algorithm for minimizing the value of a function $f : \mathbb{R}^n \to \mathbb{R}$. At the point $x_i \in \mathbb{R}^i$, the value of $f(x_0)$ increases most rapidly in the direction of the gradient of f, denoted $\nabla f(\boldsymbol{x}_i)$. Gradient descent searches for a point x_{i+1} by moving a small distance from x_i in the direction of the negative gradient. That is, $\boldsymbol{x}_{i+1} = \boldsymbol{x}_i - \alpha \nabla f(\boldsymbol{x}_i)$, where $\alpha \in (0, 1]$ is called the *step size*. For small enough α, $f(\boldsymbol{x}_{i+1}) \leq f(\boldsymbol{x}_i)$. Gradient descent produces a sequence $\{\boldsymbol{x}_i\}$ such that $f(\boldsymbol{x}_i)$ is non-increasing and converges to a local minimum, $f(\boldsymbol{x}^\star)$.

The first term in (4.11) is the partial derivative of the cross-entropy error function with respect to the k^{th} output unit:

$$\begin{aligned}\frac{\partial E}{\partial z_k} &= \frac{\partial E(\boldsymbol{t},\boldsymbol{z})}{\partial z_k} \qquad (4.12)\\ &= \frac{\partial}{\partial z_k}\left[-\sum_{i=1}^{824} t_i \log z_i + (1-t_i)\log(1-z_i)\right]\\ &= -\left(\frac{t_k}{z_k} - \frac{1-t_k}{1-z_k}\right)\\ &= -\left(\frac{t_k - z_k}{z_k(1-z_k)}\right). \qquad (4.13)\end{aligned}$$

The second term of (4.11) can be written as

$$\frac{\partial z_k}{\partial \eta_k^o} = \frac{\partial}{\partial \eta_k^o} f(\eta_k^o) = f'(\eta_k^o). \qquad (4.14)$$

Substituting the binary sigmoid function given in (4.4) into the previous equation, we have

$$\begin{aligned}\frac{\partial z_k}{\partial \eta_k^o} &= \frac{\exp(-\eta_k^o)}{\left(1+\exp(-\eta_k^o)\right)^2}\\ &= z_k\left(\frac{1+\exp(-\eta_k^o)-\exp(-\eta_k^o)}{1+\exp(-\eta_k^o)}\right)\\ &= z_k(1-z_k). \qquad (4.15)\end{aligned}$$

Thus,

$$\delta_k^o = -\left(\frac{t_k - z_k}{z_k(1-z_k)}\right)(z_k(1-z_k)) = -(t_k - z_k). \qquad (4.16)$$

These values can be arranged in the vector

$$\boldsymbol{\delta}^o = \left(\delta_1^o, \delta_2^o, \ldots, \delta_k^o\right)^\top = -(\boldsymbol{t}-\boldsymbol{z}).$$

This notation will become useful for writing the parameter update equations in a compact notation.

Finally, the second term of (4.11) is

$$\frac{\partial \eta_k^o}{\partial w_{kj}} = \frac{\partial}{\partial w_{kj}} \sum_{\xi=1}^{m} w_{k\xi}^2 y_\xi = y_j \qquad (4.17)$$

Substituting (4.16) and (4.17) into (4.11), we have

$$\frac{\partial E}{\partial w_{kj}^2} = \delta_{kj}^o y_j = -(t_k - z_k) y_j. \tag{4.18}$$

The partial derivatives of the error function with respect to the elements of W_2 can be arranged in a matrix. Let $\alpha \in \mathbb{R}$ with $0 < \alpha \leq 1$ be the *learning rate* parameter of the backpropagation algorithm, analogous to the step size parameter in the gradient descent method. The $n \times m$ matrix of weight updates, denoted $\Delta\mathsf{W}_2$, is given by

$$\Delta\mathsf{W}_2 = \left(-\alpha \frac{\partial E}{\partial w_{kj}^2} \right) = -\alpha \boldsymbol{\delta}^o \boldsymbol{z}^\top. \tag{4.19}$$

Similarly, the weight updates for the connections from the bias unit to the output units can be arranged in a vector, denoted $\Delta\boldsymbol{\beta}^h$, given by

$$\Delta\boldsymbol{\beta}^o = -\alpha \boldsymbol{\delta}^o, \tag{4.20}$$

since the activation of the bias unit is always $+1$.

Update equations for the weights connecting the input and hidden units and the weights from the bias unit to the hidden units were calculated similarly. Note that when calculating $\delta_j^h = \partial E / \partial \eta_j^h$, the multivariate chain rule must be used:

$$\delta_j^h = \frac{\partial E}{\partial \eta_j^h} = \sum_{\xi=1}^{n} \frac{\partial E}{\partial \eta_\xi^o} \frac{\partial \eta_\xi^o}{\partial \eta_j^h} = \sum_{\xi=1}^{n} \delta_\xi^o \frac{\partial \eta_\xi^o}{\partial \eta_j^h}. \tag{4.21}$$

The second term in the product can be written as

$$\begin{aligned} \frac{\partial \eta_\xi^o}{\partial \eta_j^h} &= \frac{\partial}{\partial \eta_j^h} \sum_{\varphi=1}^{k} w_{\xi\varphi}^2 y_\varphi \\ &= \frac{\partial}{\partial \eta_j^h} \sum_{\varphi=1}^{k} w_{\xi\varphi}^2 g(\eta_\varphi^h) \\ &= w_{\xi j}^2 g'(\eta_j^h). \end{aligned}$$

Thus,

$$\delta_j^h = \sum_{\xi=1}^{n} \delta_\xi^o \frac{\partial \eta_\xi^o}{\partial \eta_j^h} = g'(\eta_j^h) \sum_{\xi=1}^{n} w_{\xi j}^2 \delta_\xi^o. \tag{4.22}$$

This can be written in matrix form as

$$\boldsymbol{\delta}^h = \text{diag}\left(g'(\boldsymbol{\eta}^h)\right)\mathsf{W}_2^\top \boldsymbol{\delta}^o, \tag{4.23}$$

where $g'(\boldsymbol{\eta}^h) = \left(g'(\eta_1^h), g'(\eta_2^h), \ldots, g'(\eta_m^h)\right)^\top$ and $\text{diag}(g'\left(\boldsymbol{\eta}^h\right))$ is the diagonal matrix with the elements of $g'\left(\boldsymbol{\eta}^h\right)$ along the diagonal. For the bipolar sigmoid function, we have

$$\begin{aligned} g'(\eta) &= \frac{2e^{-\eta}}{(1+e^{-\eta})^2} \\ &= \frac{1}{2}\left(\frac{2}{1+e^{-\eta}}\right)\left(\frac{2e^{-\eta}}{1+e^{-\eta}}\right) \\ &= \frac{1}{2}g(\eta)\left(1-g(\eta)\right). \end{aligned}$$

The $m \times \ell$ matrix of weight updates for the weights connecting the input and hidden units is

$$\Delta\mathsf{W}_1 = \left(-\alpha\frac{\partial E}{\partial w_{ji}^1}\right) = -\alpha\boldsymbol{\delta}^h\boldsymbol{y}^\top, \tag{4.24}$$

and the vector of weight updates for the weights from the bias unit to the hidden units is

$$\Delta\boldsymbol{\beta}^h = -\alpha\boldsymbol{\delta}^h. \tag{4.25}$$

The weights of the network are updated using the equations

$$\mathsf{W}_1 = \mathsf{W}_1 + \Delta\mathsf{W}_1 \tag{4.26}$$

$$\boldsymbol{\beta}^h = \boldsymbol{\beta}^h + \Delta\boldsymbol{\beta}^h \tag{4.27}$$

$$\mathsf{W}_2 = \mathsf{W}_2 + \Delta\mathsf{W}_2 \tag{4.28}$$

$$\boldsymbol{\beta}^o = \boldsymbol{\beta}^o + \Delta\boldsymbol{\beta}^o. \tag{4.29}$$

Note that the above equations can only be applied after the full set of weight updates has been calculated. Once all elements of W_1, $\boldsymbol{\beta}^h$, W_2, and $\boldsymbol{\beta}^o$ have been calculated, all weights in the network can be updated in any order or simultaneously using the above equations.

An alternative weight update procedure that produces shorter learning times is called backpropagation with momentum. Let $\gamma \in \mathbb{R}$ with $0 < \gamma \leq 1$ be the momentum parameter. The matrix of weight updates in the n^{th} epoch, $\Delta\mathsf{W}_1^{[n]}$, is calculated as

$$\Delta\mathsf{W}_1^{[n]} = \alpha\boldsymbol{\delta}^h\boldsymbol{y}^\top + \gamma\Delta\mathsf{W}_1^{[n-1]}. \tag{4.30}$$

With similar adjustments made to the calculations of ΔW_2, $\Delta\beta^h$, and $\Delta\beta^h$, equations (4.26) through (4.29) can be used to calculate the new weights at each epoch.

4.5 Measuring Network Performance

The performance of the network was measured by the average precision and recall over all training and testing items. Let T^+ be the number of *true positives*, that is, the number features that were correctly activated by the model. Let T^- be the number of *true negatives*, that is, the number of features that were correctly set to inactive by the network. Let F^+ be the number of *false positives*, the number of features activated by the network which were incorrect, and let F^- be the number of *false negatives*, features which were set to inactive by the network but should have been activated.

Precision measures the number of correctly activated features from the set of all features activated by the network. This is given by

$$P = \frac{T^+}{T^+ + F^+}.$$

Recall measures the proportion of features correctly activated by the network from the set of all features associated with a concept. Recall is given by

$$R = \frac{T^+}{T^+ + F^-}.$$

These two measures can be combined into a single measure, termed the F-measure. This is given by

$$F_\beta = \frac{(1+\beta)^2 \cdot P \cdot R}{\beta^2 \cdot P + \cdot R},$$

where β is a non-negative real number. Values of β in the interval $[0,1)$ weight recall more heavily than precision. Values of β greater than 1 weight precision more heavily than recall. When $\beta = 1$, precision and recall are equally weighted. Regardless of the value of β, the value of F_β lies in the range $(0,1]$, with higher values indicating higher precision and recall. The maximum value of F_β is obtained when both precision and recall are perfect (i.e., both $P = 1$ and $R = 1$); F_β approaches 0 as either P or R approach 0. In the following analyses, the F_1-measure is used, and we denote $F = F_1$.

The *accuracy* of the network is defined as the portion of features correctly set by the network, whether active or not. This is given by

$$A = \frac{T^+ + T^-}{T^+ + T^- + F^+ + F^-}.$$

Accuracy is not an appropriate measure of network performance due to the sparse nature of the output representations. Simply setting all output units to inactive produces an average accuracy of 98.70% because on average only 1.3% of the features are active for each concept.

4.6 Determining Network Parameters

The learning rate (α) and momentum (γ) parameters and the number of hidden units were selected to maximize the network's performance on novel concepts. To determine these parameters, several combinations of values of these parameters were tested. The values tested are shown in Table 4.1. Note that all combinations of parameters were tested, resulting in a total of 252 different parameter sets. The ten best parameter combinations were then used to train five additional networks each (resulting in an additional 50 runs). The ten parameter combinations that produced the best performance on the testing data are shown in Table 4.2 below. Note that all parameter sets in this table used a value of 0.001 for the learning rate parameter.

Parameter	Values
Hidden Units	150, 250, 500, 750, 1000, 1500, 2000, 3000, 4000, 5000, 6000, 7500, 8500, 10000
Learning Rate	0.001, 0.0001, 0.00001
Momentum	0.0, 0.1, 0.25, 0.5, 0.75, 0.9

Table 4.1: Values tested for learning rate, momentum, and number of hidden units. Each combination of parameters was tested, resulting in a total of 252 different parameter sets.

Table 4.3 shows the mean value of F on the training data and the mean number of epochs until total error on the training data falls below one for each parameter combination shown in Table 4.2. Standard deviations are given in parentheses. From this, the parameter set that best maximized performance (e.g., the value of F) on the testing data set was chosen; these parameters are shown in Table 4.4.

4.7 Experiments

Each of the ten training/testing data sets was used to train ten networks, resulting in a total of 100 trained networks. Each of these networks was trained for 500 epochs. The average error on the training sets at the end of training was 0.0001 ($SD = 0.00$); the initial error was 1352.74 ($SD = 3.07$). Table 4.5 shows the average precision, recall, F-measure,

Parameters			Testing			Training		
Mom.	Hidden	Epochs	Prec.	Recall	F	Prec.	Recall	F
0.50	5000	110	0.4530	0.4674	0.4601	1.0000	1.0000	1.0000
0.90	5000	160	0.4675	0.4226	0.4439	0.9995	1.0000	0.9998
0.90	6000	220	0.3955	0.5014	0.4422	1.0000	1.0000	1.0000
0.50	7500	140	0.4470	0.4303	0.4385	1.0000	1.0000	1.0000
0.50	8500	130	0.4161	0.4189	0.4175	1.0000	1.0000	1.0000
0.50	10000	150	0.3673	0.4806	0.4164	1.0000	1.0000	1.0000
0.90	4000	150	0.3958	0.4357	0.4148	0.9998	1.0000	0.9999
0.20	6000	130	0.3895	0.4355	0.4112	1.0000	1.0000	1.0000
0.50	6000	120	0.3743	0.4560	0.4111	1.0000	1.0000	1.0000
0.90	8500	160	0.3321	0.5239	0.4065	0.9997	1.0000	0.9998

Table 4.2: The ten parameters combinations producing best performance when generalizing to novel concepts. In each parameter combination, a learning rate of 0.001 was used.

and cross-entropy error for training, testing, and randomized input vectors at the start of training, and Table 4.6 shows these values collected at the end of training. These data are summarized in Figure 4.3. Average precision, recall, and F throughout training are shown for training, testing, and randomized test items in Figures 4.4, 4.5, and 4.6, respectively. Figure 4.7 shows the average error on the training, testing, and randomized test sets throughout the 500 epochs of training[16]. The randomized test items produced a qualitatively different pattern for all measures of performance throughout training. This is discussed further below. Note that although each network was trained for 500 epochs, the following figures only show data for the first 200 epochs of training. By this time, the backpropagation algorithm had reduced total error across the training items to nearly zero, and the network showed only slight changes in performance after 200 epochs.

Precision, recall, F, and cross-entropy error were each analyzed using two-way analysis of variance with three levels of item type (training, testing, randomized) and two levels of training epoch (start, end). Data in the start training epoch condition were collected after the weights were randomly initialized but before any changes were made to the weights; data in the end training epoch condition were collected after 500 epochs of training were performed. Analyses revealed a main effect of item type on precision, $F(2,54) = 1659, p < 0.001$, recall, $F(2,54) = 508.7$, $p < 0.001$, F-measure, $F(2,54) = 4747$, $p < 0.001$, and error, $F(2,54) = 387.5, p < 0.001$. A main effect of training epoch was found for precision, $F(1,54) = 4917, p < 0.001$, recall, $F(1,54) = 877.8$, $p < 0.001$, F-measure, $F(1,54) = 12,783, p < 0.001$, and error $F(1,54) = 101.7, p < 0.001$. An inter-

[16]Note that precision, recall, F, and error were only calculated every five epochs during training.

Mom.	Hidden	F-measure	Epochs
0.9	4000	0.4094 (0.02)	162 (31.14)
0.5	10000	0.4021 (0.03)	162 (21.68)
0.9	5000	0.3990 (0.01)	152 (16.43)
0.9	6000	0.3963 (0.01)	176 (11.40)
0.5	6000	0.3784 (0.03)	122 (34.21)
0.5	7500	0.3778 (0.03)	130 (15.81)
0.9	8500	0.3747 (0.05)	182 (13.04)
0.5	5000	0.3671 (0.02)	112 (23.87)
0.5	8500	0.367o (0.04)	144 (21.91)
0.2	6000	0.3656 (0.03)	96 (15.17)

Table 4.3: Mean F and mean epochs to train for each combination of parameters shown in Table 4.2. Averages were taken over value obtained from training five networks with each combination of parameters.

Parameter	Value
Hidden Units	4000
Learning Rate	0.001
Momentum	0.9

Table 4.4: The optimal (in that sense that performance on novel stimuli is maximized) parameters set.

Type	Precision	Recall	F	Error
Training	0.0071 (0.0037)	0.3544 (0.1543)	0.0140 (0.0072)	1352.74 (61.94)
Testing	0.0081 (0.0039)	0.3543 (0.1434)	0.0158 (0.0075)	1352.29 (61.10)
Random	0.0080 (0.0037)	0.3519 (0.1427)	0.0156 (0.0072)	1353.46 (60.45)

Table 4.5: Precision, recall, F, and error at the onset of training.

Type	Precision	Recall	F	Error
Training	1.0000 (0.0000)	1.0000 (0.0000)	1.0000 (0.0000)	0.00 (0.00)
Testing	0.4330 (0.2367)	0.4020 (0.1970)	0.3760 (0.1712)	830.62 (585.70)
Random	0.0350 (0.0764)	0.4287 (0.3715)	0.0440 (0.0548)	2228.14 (1257.13)

Table 4.6: Precision, recall, F, and error at the completion of training.

action between item type and training epoch was found for precision, $F(2,54) = 1665$, $p < 0.001$, recall, $F(2,54) = 504.5, p < 0.001$, F-measure, $F(2,54) = 4783, p < 0.001$, and error, $F(2,54) = 386.9, p < 0.001$.

Post-hoc analyses using a Bonferroni correction were used to investigate the nature of the interaction between item type and training epoch. There was no difference in precision by item type at the start of training (all p's > 0.95). Upon the completion of train-

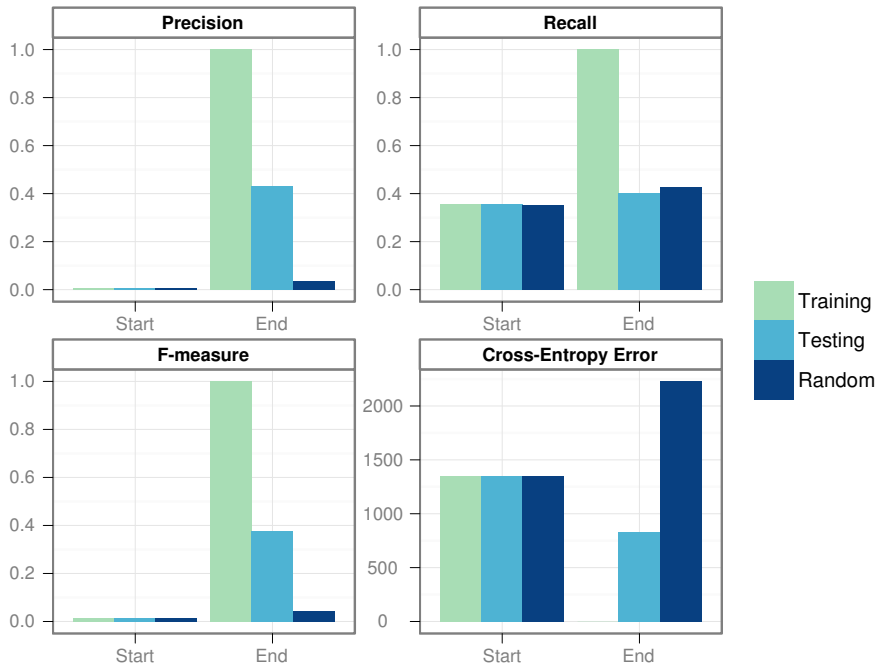

Figure 4.3: Precision, recall, F, and cross-entropy error by item type at the start and end of training. Training items show a reduction in error and an increase in precision, recall, and F during training. Testing items show a reduction in error and an increase in precision and F. Randomized items show an increase in error, and no or little improvement in precision, recall, and F.

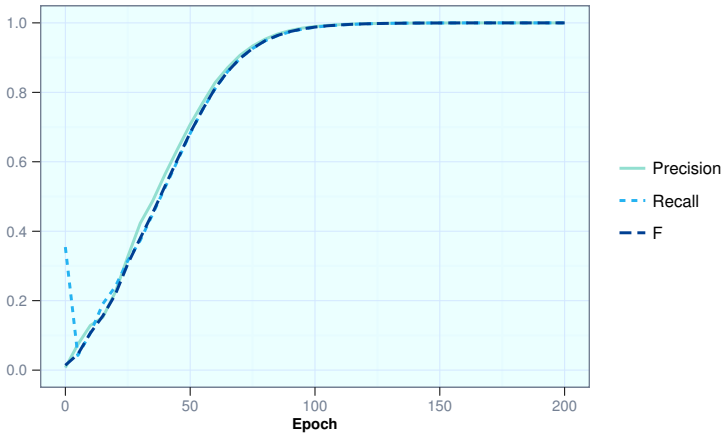

Figure 4.4: Precision, recall, and F for training items. Precision and F increase throughout training. Recall decreases during the earliest epochs of training, then increases.

ing, precision was higher for training items than test items, $t(9) = 28.34$, $p < 0.001$, and randomized test items, $t(9) = 196.4, p < 0.001$, and precision for test items was higher

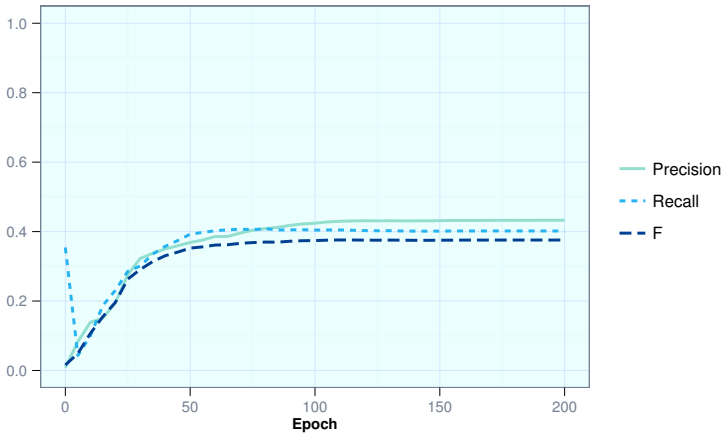

Figure 4.5: Precision, recall, and F for testing items. Although the performance on testing items is lower than that for training items, the pattern of performance throughout training is similar.

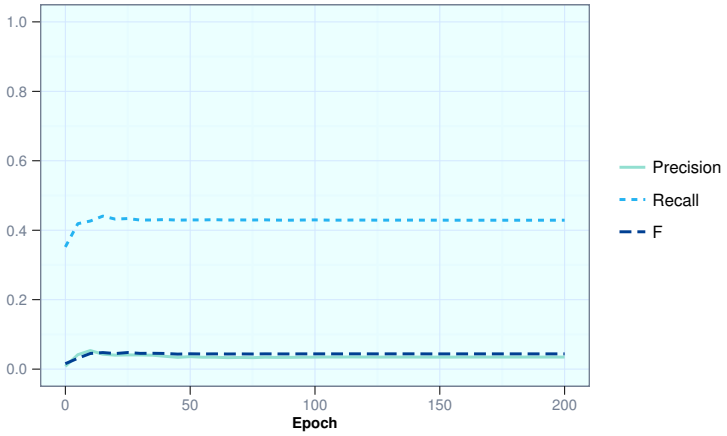

Figure 4.6: Precision, recall, and F for randomized test items. The pattern of performance throughout training is different from that observed for training and testing items.

than for randomized test items, $t(9) = 19.32$, $p < 0.001$. Precision for training items was higher at the completion of training than at the beginning, $t(9) = 44318, p < 0.001$, and precision for testing items was higher at the completion of training than at the onset, $t(9) = 21.24$, $p < 0.001$. No difference between precision at the start and end of training was found for the randomized test items, $p = 0.41$.

In the recall data, no difference was found between training, testing, and randomized test items at the start of training (all p's > 0.95). At the end of training, recall was higher for training items than testing items, $t(9) = 66.99, p < 0.001$, and randomized test items, $t(9) = 23.88$, $p < 0.001$. No difference was found between testing and ran-

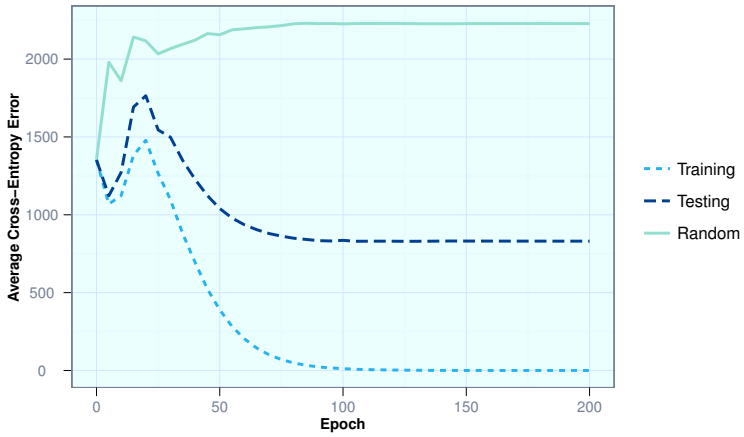

Figure 4.7: Cross-entropy error by item type. Training and testing items produce similar patterns of performance throughout training, with higher error on testing items than the training items at the end of training. Performance on randomized test items decreased throughout training.

domized testing patterns at the end of training, $p = 1.00$. Note, however, that the lack of difference between recall on testing and randomized items in the presence of higher precision for testing than randomized items suggests that the recall observed for randomized items is simply a result of activating a large number of output units. Indeed, the number of output units activated differed between randomized test items and test items, $t(9) = 14.44, p < 0.001$, with more output units activated in response to randomized test items ($M = 316.76, SD = 67.67$) than test items ($M = 15.30, SD = 4.39$). Further, testing and randomized test items showed qualitatively different patterns of recall during training, as shown in Figures 4.5 and 4.6. At the onset of training, recall for both item types is similar. Recall for randomized items varied little during training. Recall for testing items, however, drops to zero in the earliest epochs of training, then increases quickly before stabilizing. Recall was higher at the completion of training than at the onset of training items, $t(9) = 583.48, p < 0.001$, and testing items, $t(9) = 5.02, p = 0.037$. For randomized testing items, no difference was found between recall at the start of training and at the end of training, $t(9) = 3.31, p = 0.137$.

Post-hoc analysis of F-measure showed no difference between training, testing, and randomized test items at the start of training (all p's > 0.95). At the end of training, F-measure was higher for training items than testing items, $t(9) = 55.13, p < 0.001$, and randomized test items, $t(9) = 214.78, p < 0.001$, and F-measure for testing items was higher than for randomized test items, $t(9) = 27.29, p < 0.001$. Higher F-measures were observed at the end of training than the start for training, $t(9) = 22553, p < 0.001$, testing,

$t(9) = 31.91, p < 0.001$, and randomized test items, $t(9) = 6.78, p = 0.003$.

There was no difference between mean error for the training, testing, and randomized test items at the beginning of training (all p's > 0.95). At the completion of training, error was lower for training items than testing items, $t(9) = 15.94, p < 0.001$, and randomized test items, $t(9) = 26.44$, $p < 0.001$. Error for testing items was lower than for randomized items, $t(9) = 13.51$, $p < 0.001$. Error decreased from the start and end of training for training items, $t(9) = 1303.10$, $p < 0.001$, and testing items, $t(9) = 10.06$, $p < 0.001$. For randomized test items, error increased throughout training, $t(9) = 10.37$, $p < 0.001$. Note that, as shown in Figure 4.7, the randomized testing items showed a qualitatively different pattern of error from the training and testing items throughout training. Error on training and testing items decreased during the first few epochs of training then increased to level higher than the initial error produced on these items, followed by a rapid decrease in error. Error on the randomized test items, however, increased throughout training, particularly during the first few epochs of training.

Examination of the order that features were learned by the network provides insight into what information the network exploited to learn the complex interaction between the semantic vectors and the feature-based vectors. The optimal parameters used to train the networks were selected to produce the strongest performance on the testing items. As shown above, the training items were learned perfectly after only 500 epochs of training, with near perfect accuracy after only 200 epochs. This leaves a narrow window in which to examine the order in which features were learned. To emphasize the time-course of the network's learning 100 additional networks were trained using a lower learning rate of 0.00001; all other parameters were identical to those shown in Table 4.4. In this analysis, a feature was assumed to be familiar to the network when it's average F-measure across training sets was above 0.5. This threshold was selected to indicate that the network had gained familiarity with a particular feature rather than to suggest that the feature had been mastered by the network. Table 4.7 shows the first 30 features learned by the network, the average number of epochs taken for the F-measure to reach a level of 0.5, the standard deviation of the number of epochs, and the frequency of the feature in the list of 465 concepts (that is, the number of concepts that exhibited the feature).

The features listed in Table 4.7 fall into two general categories: those that divide the concepts into broad categories (e.g., ⟨an animal⟩, ⟨a vegetable⟩, and ⟨clothing⟩), and those that are strongly associated with these categories (e.g., ⟨has feathers⟩, ⟨beh - flies⟩, ⟨has wings⟩, and ⟨has a beak⟩ are all learned within 20 epochs of ⟨a bird⟩). After learning the feature ⟨an animal⟩ after 43 epochs of training, the network quickly learns subcategories

Feature	Epochs Until $F = 0.5$		Frequency
	Mean	St. Dev.	
an animal	43.13	5.65	91
a vegetable	61.97	8.65	28
clothing	64.80	8.60	28
a fruit	73.54	10.01	33
a bird	75.10	9.82	31
has feathers	76.87	9.30	30
a musical instrument	78.33	10.38	16
beh - flies	84.60	12.57	36
has wings	87.88	11.70	35
a mammal	92.88	17.34	41
has a beak	93.79	13.46	30
is edible	93.79	17.82	76
lives in water	99.80	14.36	29
tastes sweet	123.28	31.52	21
an insect	125.40	19.92	11
grows in gardens	126.62	28.92	18
made of metal	130.45	21.06	114
found in kitchens	130.56	21.44	29
a weapon	135.35	19.14	28
used for transportation	145.05	23.42	31
a fish	145.71	37.23	28
has 4 legs	146.87	22.34	46
a tree	148.23	28.12	5
has wheels	150.05	25.06	20
inbeh - produces music	151.46	35.89	12
beh - swims	152.93	25.32	30
used in bands	155.86	38.89	9
lives on farms	158.94	29.63	12
used for killing	170.05	27.65	19
grows in forests	170.10	34.08	5

Table 4.7: The first thirty features learned. These features either group concepts into broad categories (e.g., ⟨an animal⟩, ⟨a vegetable⟩), or are strongly associated with a small number of categories (e.g., ⟨has a beak⟩ is strongly associated with the category Birds).

of this general category: ⟨a bird⟩ is learned after 75 epochs, *a mammal* is learned after 93 epochs, ⟨an insect⟩ is learned after 125 epochs, and ⟨a fish⟩ is learned after 146 epochs. The network coarsely divides the category of plants via the features ⟨a vegetable⟩ learned at epoch 62, ⟨a fruit⟩ learned at epoch 74, and ⟨a tree⟩ learned after 148 epochs. Man-made objects are differentiated only into broad categories by the features ⟨a musical instrument⟩, ⟨found in kitchens⟩, ⟨a weapon⟩, and ⟨used for transportation⟩. Exemplars of

each of these categories may possess the feature ⟨made of metal⟩. Table 4.8 shows the 501st to 520th features that are learned by the network. Note that these features are very specific to a small number of concepts (e.g., ⟨used for chopping⟩) or are general features that do not tie to any specific category of concepts (e.g, ⟨is red⟩). Table 4.9 shows the first 40 concepts learned by the network. The earliest concepts learned are those that exhibit the earliest features learned. For example, eight of the first ten concepts learned are birds, a category that is well represented among the features shown in Table 4.7. The remaining concepts listed in the table fall into the categories identified in Table 4.7: fruits, vegetables, mammals, fish, musical instruments, and weapons.

Order	Feature	Epochs	Order	Feature	Epochs
501	has pulp	372.98	511	beh - makes noises	376.87
502	found in cupboards	373.08	512	has teeth	376.92
503	used for serving food	373.08	513	used by pushing	377.17
504	lives in a nest	374.70	514	is red	377.73
505	beh - eats flies	374.75	515	is damp	377.83
506	used for washing	374.95	516	is white	378.08
507	is bright	375.15	517	used for chopping	378.23
508	is luxurious	375.20	518	is uncomfortable	379.65
509	is ugly	375.71	519	found on tables	379.70
510	is grey	376.67	520	lives in forests	380.05

Table 4.8: The 501^{st} to 520^{th} features learned. These are features that are often associated with a large number of diverse categories (e.g., ⟨is ugly⟩, ⟨is grey⟩), or are specific to only a few concepts (e.g., ⟨has pulp⟩).

4.8 Effect of Number of Input and Output Units

In this section, the effect of the number of input and output units on the network's ability to identify features of novel concepts is examined. Due to the high number of parameters calibrated by the backpropagation algorithm and the small number of training patterns available, it is possible that the network's parameters were overfit to the training data. Reducing the number of input units reduces the total number of parameters in the network and reduces the likelihood over overfitting the network's parameters to the training patters. This would result in an increase in the network's performance on the testing patterns.

This possibility was examined by reducing the number of inputs to the network to 50, 100, or 200 and training the network following the same procedure as used previously, but with lower-dimensional input vectors. That is, for each of the ten pairs of training

Order	Concept	Epochs	Order	Concept	Epochs
1	SPARROW	78.33	21	TROMBONE	153.52
2	PHEASANT	78.86	22	STORK	154.09
3	STARLING	99.20	23	WOODPECKER	161.02
4	PELICAN	101.63	24	SWORD	163.41
5	GOOSE	108.07	25	RAVEN	164.20
6	PARTRIDGE	110.23	26	RIFLE	165.00
7	FALCON	121.25	27	RAT	166.70
8	MACHETE	122.16	28	SAXOPHONE	169.69
9	SPINACH	125.80	29	CARIBOU	172.50
10	DOVE	126.63	30	NIGHTINGALE	172.86
11	PIGEON	136.84	31	CABBAGE	172.95
12	DAGGER	143.64	32	DOG	173.30
13	SALMON	146.59	33	NIGHTGOWN	173.98
14	PERCH	146.70	34	GUN	174.89
15	FINCH	146.93	35	HAWK	178.52
16	ELK	150.00	36	FREEZER	179.66
17	TANGERINE	151.48	37	BISON	181.25
18	HARPSICHORD	152.24	38	MACKEREL	181.48
19	VULTURE	152.76	39	SWAN	184.80
20	OWL	153.27	40	FOX	185.00

Table 4.9: The first forty concepts learned by the network. The concepts learned earliest by the network are those that exhibit a high number of the first features learned.

and testing data sets, ten networks were trained, resulting in 100 networks trained on each set of lower-dimensional input vectors. Since the SVD retains only the columns of the matrices U and V that correspond to the largest singular values, in decreasing order by singular value, creating input vectors with fewer components could be achieved by simply truncating the 300-dimensional input vectors to the required number of components.

Table 4.10 shows the network's performance on training, testing, and randomized testing patterns when using input patterns consisting of 50, 100, or 200 components. The network's performance when using the original 300-dimensional input vectors, from Table 4.6, is included as well. When using 100- and 200-dimensional input vectors, the network was able to perfectly learn all training items. When the dimension of the input vectors was reduced to only 50, the network continued to produce errors after 500 epochs of training. However, the average number of errors per concept was low for both false positives ($M = 0.023$, $SD = 0.382$) and false negatives ($M = 0.023$, $SD = 0.380$).

		Precision	Recall	F-Measure	Error
300 In	Training	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)	0.00 (0.00)
	Testing	0.43 (0.24)	0.40 (0.20)	0.38 (0.17)	830.62 (585.70)
	Random	0.04 (0.08)	0.43 (0.37)	0.04 (0.05)	2228.14 (1257.13)
50 In	Training	0.99 (0.03)	0.99 (0.03)	0.99 (0.03)	0.04 (0.60)
	Testing	0.36 (0.24)	0.27 (0.19)	0.27 (0.18)	986.19 (738.07)
	Random	0.05 (0.12)	0.36 (0.39)	0.04 (0.06)	2100.59 (1394.23)
100 In	Training	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)	0.00 (0.00)
	Testing	0.42 (0.25)	0.31 (0.20)	0.32 (0.18)	965.46 (726.27)
	Random	0.05 (0.11)	0.36 (0.39)	0.04 (0.06)	2214.68 (1268.72)
200 In	Training	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)	0.00 (0.00)
	Testing	0.44 (0.23)	0.37 (0.19)	0.37 (0.17)	858.97 (621.29)
	Random	0.05 (0.13)	0.35 (0.38)	0.05 (0.06)	2282.87 (1183.85)

Table 4.10: Precision, recall, and F for varying numbers of input units. Performance was similar when 200 or 300 input units were included in the network and the network had a reduced ability to generalize to novel concepts when only 100 or 50 input units were included.

An improvement in the network's ability to identify properties of novel concepts would manifest as in increase in the network's performance, as measured by precision, recall, and F, on the testing items. When only 200 input units were included in the network, there was no difference in precision, $t(199) = 1.38$, $p = 0.168$, the F-measure, $t(199) = 1.13$, $p = 0.260$, or error, $t(199) = 1.43$, $p = 0.137$. Recall was lower when only 200 input units were used than when 300 input units were used, $t(199) = 5.34$, $p < 0.001$.

When the network was trained using 100-dimensional input vectors, there was no

change in precision, $t(199) = .92$, $p = 0.359$. Recall was lower when using 100 input units than when using 300 input units, $t(199) = 8.47$, $p < 0.001$, as was the F-measure, $t(199) = 6.86$, $p < 0.001$. Total error at the end of training was higher for 100-dimensional inputs than for 300-dimensional inputs, $t(199) = 4.20$, $p < 0.001$.

When only 50 input units were included in the network, there was a decrease in precision, $t(199) = 4.76$, $p < 0.001$, recall, $t(199) = 10.14$, $p < 0.001$, and F-measure, $t(199) = 9.36$, $p < 0.001$. Error was higher when 50 inputs were used than when 300 inputs were used, $t(199) = 4.18$, $p < 0.001$.

These results support the choice of using 300-dimensional input vectors. Although the network's performance was similar when the number of input units was reduced to 200, using 100 or fewer inputs resulted in a degradation in performance.

When constructing the output representations, any feature that was associated with two or more concepts was included, resulting in a total of 824 features. However, the distribution of feature frequencies (that is, the number of concepts that possess a particular feature) is heavily skewed toward low frequencies. The average frequency of the features is 6.45 ($SD = 11.92$), and the most common feature frequency is only 2. It is likely that the network's performance on the testing items is impaired by these low frequency features. For example, if a feature with a frequency of two appears in the set of test items, there is only one instance of this feature in the training data. Thus, the network may overfit its parameters to this single example, reducing its ability to identify the feature for novel inputs. To investigate this possibility, the number of output units in the network was reduced and the network was trained using the same procedure as earlier in this chapter.

The number of outputs used in these experiments was determined by the number of features whose frequency is greater than or equal to some cutoff frequency. In the initial experiments performed in Section 4.7, this cutoff was set to 2, resulting in 824 output units in the network. Two additional training and testing data sets were created. The first used a cutoff frequency of five, resulting in a total of 270 output units included in the network, and the second set used a cutoff of ten, reducing the total number of output units to only 120. The results of training the network using these data sets are shown in Table 4.11. The results from the initial experiments are included for reference.

When the number of output units was reduced to 270, there was an increase in precision, $t(199) = 4.50$, $p < 0.001$, an increase in recall, $t(199) = 3.80$, $p < 0.001$, and no difference in F, $p = 0.48$, on the testing items. This result may seem counter-intuitive, as F is calculated from precision and recall and always falls between the two. However, F is low when either precision is low or recall is low (and, of course, when both are low).

		Precision	Recall	F-Measure	Error
824 Out	Training	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)	0.00 (0.00)
	Testing	0.43 (0.24)	0.40 (0.20)	0.38 (0.17)	830.62 (585.70)
	Random	0.04 (0.08)	0.43 (0.37)	0.04 (0.05)	2228.14 (1257.13)
270 Out	Training	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)	0.00 (0.00)
	Testing	0.48 (0.31)	0.45 (0.29)	0.37 (0.21)	771.06 (655.85)
	Random	0.05 (0.11)	0.39 (0.40)	0.05 (0.05)	1115.02 (1147.75)
120 Out	Training	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)	0.00 (0.00)
	Testing	0.59 (0.30)	0.43 (0.25)	0.45 (0.22)	595.24 (520.67)
	Random	0.07 (0.11)	0.40 (0.38)	0.09 (0.08)	1086.24 (952.73)

Table 4.11: Precision, recall, and F for different numbers of output units. As more low-frequency features are removed from the network, performance increases.

When averages are taken over several data points, it is possible that the average value of F falls below both the average precision and average recall.

When the network contains only 120 output units, there was an increase in precision, $t(199) = 13.24$, $p < 0.001$, an increase in recall, $t(199) = 2.49$, $p = 0.014$, and an increase in F, $t(199) = 9.61$, $p < 0.001$, when compared to the results of the original experiments. When compared to the results obtained using 270 output units, the network containing only 120 output units demonstrated higher precision, $t(199) = 7.71$, $p < 0.001$, and higher F, $t(199) = 8.41$, $p < 0.001$, but no difference in recall, $p = 0.164$.

These results are as expected and indicate that the network's performance is poorest on the lowest frequency features. A direct consequence of this observation is that the network's ability to generalize could be strengthened by increasing the number of items in the training set, providing the network with more data from which to generalize to novel concepts. However, this avenue provides only limited potential, as adding additional concepts requires a significant time commitment to collect the additional data.

5 Summary and Conclusions

5.1 Discussion

Chapter 2 described a method that can be used to create abstract representations of the meanings of words by analyzing a large body of written text (a resource that is seemingly inexhaustible in the on-line world). These representations take the form of high-dimensional vectors. The axes of the high-dimensional space in which these vectors exist are abstract. That is, the axes do not correspond to any particular features or properties of word meaning. The vectors can only be interpreted relative to one another and must be taken to be atomic; no individual component of any representational vector can be interpreted in isolation. Rather, word meaning is stored in a distributed fashion across all components of the vector. While no component of a vector acts to measure, for example, size, with larger values of that component corresponding to larger objects, such properties are represented in the vectors in a distributed manner, that is, via combinations of values across subsets of the components.

Chapter 3 provided several demonstrations that these vectors capture meaning that maps on to our intuitions and that the representations are able to reproduce a variety of less obvious results from the psycholinguistic literature. MDS applied to the co-occurrence representations produces sensible groupings of concepts into categories, both when the categories were vastly different from one another and when the categories were specializations of a common superordinate category. For example, MDS was used to categorize animals, body parts, cities, and geographical locations, demonstrating that the representations differ between categories that share little in common (animals, body parts, and places) and subcategories of a common category (cities and geographical locations, in the category places). MDS was also used to categorize common nouns and proper nouns that were either male, female, or surnames and to categorize words by part of speech. In all cases the representations contained sufficient information to categorize concepts into both coarse and more precise categories and also to identify an appropriate set of properties by which to classify concepts.

The results in Chapter 3 demonstrated that when MDS is applied to the semantic vectors, concepts are grouped into natural categories rather than classified according to some specific properties that do not produce intuitive groups, such as size, colour, or deformability. MDS was also used to show that the representations contain, to some extent, syntactic knowledge as well: when semantic categories cannot be identified, words can be grouped by part of speech. While the MDS results show less distinct groupings between part-of-speech categories than between semantic categories, words from each

part of speech category are centred around a distinct region of the plane. Each of the results in Section 3.2 show that classification was performed at a natural, intuitive level appropriate for the given set of words.

Further, when the cosine between two co-occurrence vectors is used as a surrogate measure of the priming effect in semantic-priming lexical decision experiments, the co-occurrence representations can be used to reproduce many behavioural results. The subtle differences in word similarity and association that subjects are sensitive to are captured in the co-occurrence vectors. When strength and type of semantic or associative relationship between words was varied, mean similarity between co-occurrence vectors exhibited the same pattern of results that was observed in subjects' performance on language-related tasks. The co-occurrence representations were able to reproduce behavioural results when the relationship between words was either direct or mediated through some other word.

Collectively, the MDS results and the successful behavioural simulations in Chapter 3 suggest that the method described in Chapter 2 was effective at capturing semantic knowledge about words solely from experience with how words are used together in language. Using simple scaling techniques, concepts could be accurately classified into both concrete and abstract categories, and category membership could be imputed both when all categories fell under a common superordinate category and when there was no shared superordinate category. The model was able to simulate tasks in a way that both mimics our intuition and agrees with less intuitive observations from psycholinguistic studies. This suggests that the high-dimensional space occupied by the co-occurrence vectors shares, in some sense, the same organizational structure as semantic memory in the brain, at least to the extent that similarity between vectors in the high-dimensional space is correlated with the increase in speed with which subjects can recognize a word when it is preceded by a related word over when it is preceded by an unrelated word. This structure is obviously greatly simplified in the high-dimensional space.

That the above results were produced by a model that was trained with only linguistic input supports the hypothesis that language acquisition can be achieved from symbolic input alone. The representations produced by the model are symbolic, as their components do not correspond to any particular property of the words or their referents and are not grounded in any way to the physical world. The criticism of symbolic models of cognition made by Harnad (1990) and Searle (1980), and directed specifically toward co-occurrence models by Glenberg and Robertson (2000), is predicated on this trait: the symbolic representations produced by co-occurrence models are not suitable for use

in tasks, such as sensicality judgments, that require grounded knowledge of concepts. The defence that co-occurrence based representations are grounded in the linguistic environment (Burgess, 1998, 2000; Burgess & Lund, 2000) is not applicable to the model described in Chapter 2. The components of the co-occurrence vectors produced by this model do not measure association to other words, as in HAL. Rather, the components of the vectors position a concept in a high-dimensional space in which the axes have no interpretable meaning. However, this does not preclude the possibility that the representations contain grounded knowledge.

Chapter 4 described a neural network model that can be used to produce a list of physical and behavioural properties of a concept from its semantic vector. The network was able to generalize this ability to novel concepts to identify properties of concepts with accuracy higher than chance when the network was not explicitly trained on the concept. Qualitatively, the performance on training and testing items was similar throughout training and differed from the pattern of performance shown by randomized test items used as a baseline for performance. That the network's performance on the testing data was qualitatively similar to that observed on the training data and that the network's performance on the testing items was higher than would be observed by chance suggests that there is some regularity to the vectors constructed from co-occurrence data that can be accessed using available techniques.

The order that features are learned by the model is qualitatively similar to the way in which semantic knowledge is acquired by children. The earliest features learned by the network are those that best separate concepts into broad categories and those that are strongly associated with these categories. The network learned, for example, the property ⟨a bird⟩ early in training and learned the associated features ⟨has feathers⟩, ⟨beh - flies⟩, ⟨has wings⟩, and ⟨has a beak⟩ shortly after. This cluster of four features is strongly associated with the property of being a bird. With only a few exceptions, these properties are true of all birds and, again with only few exceptions, these properties are not possessed by other objects. Features learned later in training are those that are more broadly applicable across a range of unrelated categories. For example, the features ⟨is ugly⟩, ⟨is grey⟩, and ⟨beh - makes noises⟩ were learned after approximately 375 epochs of training. These features do not act to separate the concepts into coherent, natural categories. Features learned during the earliest stages of training, such as ⟨an animal⟩ and ⟨a vegetable⟩ are associated with clearly defined categories with little or no shared membership. These results agree with those of Rogers and McClelland (2004), who used a network trained using backpropagation to explore the time-course of knowledge acquisition in a parallel distributed processing model of semantic memory. In their ex-

periments, Rogers and McClelland used a small, artificial corpus of concepts whose features were hand-selected by the researchers and used localist representations for input. That is, each concept was represented by a single input unit in the network. In the current work, rich distributed representations derived from natural written language were used as input and subject-produced feature-production norms were used as output. Although significant work remains to match the myriad results shown by Rogers and McClelland, the initial results shown in Chapter 4 are promising and demonstrate that at least some of their findings hold in a more complex environment.

It should be noted that, while performance was higher on the training items than on the testing items, this result is common to all machine learning methods and is unsurprising. The items reserved for testing in each training set were carefully selected to produce the best match between the distribution of features across the training items and testing items. The results in Chapter 3 show that the semantic vectors of words with similar meaning are similar. Just as neural networks are able to recognize a novel instance of a handwritten letter "A" by generalizing knowledge acquired through previous experience with the same letter, the network in the current research was able to generalize to novel semantic vectors and, based on regularities in the input vectors, produce a similar pattern of activation across the output units. To see how these regularities in the input vectors arise, consider, as an example, the words CAT and DOG. Both words refer to relatively small domesticated mammals that are often kept as pets. Their interactions with our environment are similar: both sit on people's laps, eat food out of bowls, get pet, and may or may not be allowed on the furniture. These similarities are reflected in the linguistic habitats in which the two words are found. The contexts in which both CAT and DOG occur in written language describe the physical and behavioural properties and the ways in which each animal interacts with its environment, other objects, and humans. This is done both directly, through sentences such as *she pet the cat/dog*, *the cat/dog bit the man*, and *the cat/dog chased its tail*), and indirectly. For example, the size of an object can be inferred from the way in which that object interacts with other objects of known size. The sentence *the cat/dog slept on the couch* suggests an upper bound on the size of a cat or dog. The model described in Chapter 2 attempts to capture these regularities in language through analysis of word co-occurrence. By using SVD to reduce the dimension of the co-occurrence matrix, the model attempts to capture the higher-order co-occurrence information that supports language acquisition in humans, that is, the indirect relationships between words that contribute to their meanings. The results in Chapter 3 showed that the semantic vectors produced by the method rate object similarity in a way that agrees with our intuitive judgment and that the structure of

the high-dimensional space in which these vector exist mirrors, in a very coarse approximation, the structure of human semantic memory. More succinctly, words that refer to similar objects are used in similar linguistic contexts and this is reflected in the semantic vectors in a way that agrees with our intuition about similarity.

In addition, the referents of the words CAT and DOG share many common traits: both are animals, are mammals, have four legs, have tails, and have fur. These similarities suggest that the two words would share many features in the feature-production norms of McRae et al. (2005). This was observed in the data: of the 824 features included in the network, CAT possess 15 and DOG possess 13, with seven features shared between the two (CAT shares 46.7% of its features with DOG and DOG shares 53.8% of its features with CAT). Thus, similar concepts have similar feature-based output vectors as well. The network is faced with the task of translating similar vectors in the space of input vectors to similar vectors in the space of output vectors. If certain conditions on the number of hidden units in the network are met, a two-layer neural network with non-linear activation functions trained using backpropagation is capable of learning, to within a specified accuracy, any arbitrary mapping between a set of input and a set of output vectors (Cybenko, 1989). Thus, while it is no great surprise that the network was able to learn the correct mapping for the training items, it is interesting (but not wholly unexpected) that the network was able to generalize this result to novel items that were not observed during training. However, this work, reported earlier in Durda et al. (2009), is the first attempt to do so.

Louwerse (2008) argues that both semantic features and word co-occurrence are necessary sources of information to enable language acquisition. His symbol interdependency hypothesis proposes that many concepts are grounded in the perceptual, motor, and introspective systems of the brain. Other concepts are not directly grounded, but are partially grounded indirectly through their association with other words. While performing a language-related task, the grounded representations are partially, but not fully, activated. Further, it is argued that grounded knowledge is so important to language acquisition and processing that the physical properties of concepts have become encoded within the statistical structure of language. That is, the physical properties of, say, cats and dogs inform the way in which we speak about them. Riordan and Jones (2011) showed that there is significant redundancy between the symbolic and featural inputs available during language acquisition, but that the two sources are also complementary to one another, concluding that symbolic and grounded theories should not stand in conflict with one another and that research should focus on the mechanisms through which the two sources of information are integrated during language acquisi-

tion. The results of Chapter 4 complement the work of Riordan and Jones by demonstrating the there is sufficient knowledge of perceptual properties of words stored in the symbolic relationships between words to extract the former through analysis of the latter. Further, these results provide evidence supporting the symbol interdependency hypothesis of Louwerse. The input representations of concepts were derived entirely from word usage, but the network was nonetheless able to identify reliable cues about the features of the concepts, even when it had not been explicitly trained on a concept.

5.2 Shortcomings and Future Work

One shortcoming of the work presented in Chapter 4 is the simple network architecture used. In the two-layer feedforward network, activation flows in only one direction: from the input units to the output units. Ideally, the network should be able to operate in the reverse direction. Given a set of features that a concept possesses, the network should be able to access the correct abstract semantic representation of that concept, roughly simulating the act of recognizing an object. This is not possible with a feedforward network. A recurrent network trained using backpropagation through time (BPTT; Werbos, 1990) would allow the network to associate co-occurrence representations with feature-based representations in both directions. Using a recurrent network has the potential to improve the network's performance when generalizing to novel concepts as well. Recurrent networks operate by incrementally updating the activations of units until the units "settle" into a stable pattern of activation. During this process, the activations of all units in the network are able to influence the input to other units. Further, the BPTT training algorithm can produce networks with an attractor basin structure. That is, once the network arrives at a pattern that is sufficiently similar to a pattern on which the network was trained, the weights between the units push the activations closer toward the learned pattern. Thus, the learned patterns act as "basins" where activation in the network collects. This principle would apply equally well to subsets of the feature units. Thus, through recurrent connections, the network could identify the pattern that, for example, ⟨a bird⟩, ⟨has feathers⟩, ⟨beh - flies⟩, ⟨has wings⟩, and ⟨has a beak⟩ nearly always occur together and this could be encoded in the network's recurrent connections during learning. When the network encounters a novel concept which it identifies as having wings and having a beak, it can generalized based on knowledge about the relationships between features to impute the concept with the features ⟨a bird⟩, ⟨has feathers⟩, and ⟨beh - flies⟩. This could potentially improve the network's performance on novel concepts as features could be activated based on information in both the input vector and

information about feature correlation encoded in the network's connections. If a feature is only weakly represented in the input pattern, feature correlation could provide the additional information required to correctly activate a common cluster of features. Indeed, it has been shown that feature correlation plays an important role in semantic acquisition and later recall of acquired knowledge (McRae et al., 1999, 1997; Rogers & McClelland, 2004).

A further difficulty with the network described in Chapter 4 concerns how features will be generalized to novel concepts. The network is able to identify features of novel concepts based on similarities between the novel input and learned representations. As shown by the simulation of the results of Chiarello et al. (1990) and Ferrand and New (2003), the co-occurrence representations used as input to the network capture associative relationships between concepts: cosine similarity between words that are associatively related is higher than between words that are unrelated. It is reasonable to expect that, in the same way the network may correctly generalize the feature ⟨has 4 legs⟩ from the representation for CAT to the semantically similar representation for DOG, the network may also incorrectly generalize this feature to MEOW, a word that is only related to CAT through association. This is similar to the criticism that co-occurrence models are unable to distinguish between sensical and non-sensical sentences (Glenberg & Robertson, 2000). In this case, the co-occurrence representations do not contain sufficient information to distinguish between the sensical generalization of the property ⟨has 4 legs⟩ from CAT to DOG and the non-sensical generalization from CAT to the associated word MEOW.

Additional future research directions concern the representations used for featural information. The feature units in the network used in Chapter 4 were simple localist representations: each output unit corresponded to a single feature and each unit's binary activation indicated the presence or absence of that particular feature. Unlike the corpus used to construct the co-occurrence representations, which occurs naturally as a by-product of written communication, the data upon which the output representations are based were collected by asking subjects to perform an artificial task, namely, to exhaustively list properties that describe different living and non-living things. In light of this, a tremendous amount of effort would be required to expand the number of items for which output representations are available, whereas producing additional input representations requires only a marginal increase in computational effort for each additional item. An ideal solution to this issue is to eliminate the human effort all together and design a network so that it can identify featural primitives from the input corpus. However, it is not clear how this task could be accomplished.

More practical considerations are related to the implementation of the algorithm described in Chapter 2. During the first pass through the corpus to collect the individual word frequency counts, a trie was used to store the dictionary of all tokens so far encountered as well as their frequency. This data structure is inefficient in terms of memory usage. The trie is a tree-based data structure in which each node has an outgoing branch for each character in the alphabet over which the words are defined. In this work, all words were converted to uppercase, and the apostrophe and dash were considered to be part of a word. Thus, each node in the structure contained an array of 28 pointers to the next character. While the trie allows for simple implementation and fast look-up of words, it requires a large amount of memory and is particularly poor when many of the words in the trie contain long, unique suffixes. Consider, for example, a trie containing the words MISCELLANEOUS and MISSISSIPPI. When retrieving one of these strings from the trie, the correct string can be determined when the fourth character is examined. Due to the structure of a trie, however, the entire word must be examined. Further, each of the unique suffixed, CELLANEOUS and SISSIPPI, are stored separately and require one node in the trie per character. This results in a structure that more closely resembles a linked list than a tree, and requires a large amount of memory to store and can be searched in below optimal time. Further, the trie requires $O(n^2)$ time to construct, where n is the total length of all strings in the trie.

The suffix tree is an alternative to the trie that can be used to provide fast look-up of strings without the high memory requirements of a trie (Bieganski, Riedl, Cartis, & Retzel, 1994; Weiner, 1973). A suffix tree can be viewed as an optimal trie in which every node has at least two children. This eliminates the linked list structure that emerged in the trie in the example above. In a suffix tree, the correct word would be identified at the fourth character and the remainder of the word could be skipped. In addition to lower memory requirements and more efficient string look-up, a suffix tree can be constructed in time that is linear in the total length of all strings in the tree (Farach, 1997). This provides a significant advantage when constructing a large dictionary, as is required during the initial frequency counts.

The backpropagation algorithm used to train the network in Chapter 4 often requires a high number of iterations to converge on a set of weights that minimizes the error over the training items. This slow convergence often leads to high training times and limits the size of the networks that can be used in practice. An alternative algorithm that can be used to train feed-forward neural networks with one layer of hidden units is the Extreme Learning Machine (Huang, Zhu, & Siew, 2004). This algorithm randomly assigns weights to the connections from the input units to the hidden units. The weights

on the connections between the hidden and output units are then determined analytically, eliminating the need for a time-consuming iterative training process. Networks trained using this algorithm demonstrate similar performance to those trained using backpropagation with a significant savings in training time. Using this algorithm would allow for larger networks to be used in the current work and would reduce the time required to perform additional experiments analyzing the network's performance. The Extreme Learning Machine algorithm would be particularly useful when integrating the semantic representations produced in Chapter 2 into more comprehensive neural network models of language processing, such as the "triangle" model used by Seidenberg and McClelland (1989) and Harm and Seidenberg (2004), which integrates semantic, orthographic, and phonological processing.

5.3 Summary of Contributions

The results shown in this dissertation demonstrated that it is possible to identify perceptual information from (symbolic) co-occurrence-based representations. This is a step toward addressing a common and difficult problem for co-occurrence models of semantic memory: representations produced from word co-occurrence are not grounded. This criticism arose from work in the field of grounded (or embodied) cognition, in which the symbol grounding problem was introduced as a problem for most models of memory. In the theory of grounded cognition, knowledge is acquired by integrating multimodal representations of experience with the world, body, and mind into a common representational memory system. Later recall of this knowledge occurs via a simulation process in which the brain reproduces the multimodal representations that were captured during knowledge acquisition. Simulation is assumed to be a core method of computation in the brain, operating on a representational system that is shared between processing systems. In this theory, cognition is closely tied to the perceptual and motor systems, as well as to introspective and emotional states. Taken as a definition of semantic memory, the common memory system of grounded cognition is more intricate than the view of semantic memory as a general storehouse of knowledge – essentially a dictionary, thesaurus, and encyclopedia rolled into one (McNamara & Holbrook, 2003).

A problem closely related to the symbol grounding problem is that the richness of human experience cannot be used to inform a computational model of semantic knowledge. Our experience with the world occurs in a variety of media, such as visual and other forms of perceptual input, but computational models of semantic memory are limited to purely linguistic input. Due to this inability to integrate non-linguistic input,

representations constructed solely from co-occurrence data are necessarily impoverished and, thus, are unable to represent the full complexity of human semantics.

These problems have been used to suggest that co-occurrence models exist in opposition to grounded cognition. However, the results in this dissertation suggest quite the opposite. Consistent with the theory of grounded cognition, linguistic input forms only a single component of the input to the semantic system. Burgess (1998, 2000) argued that the co-occurrence representations created by the HAL model are grounded because each component of a word's semantic vector measures the strength of association between that word and some other word from the corpus. Here, it was shown that information about feature characteristics can be represented in a latent manner throughout a vector representations and can be used to identify features of a concept from co-occurrence representations in which the axes are arbitrary and have no clear interpretation. This offers a more direct form of symbol grounding. The representations produced by the method described in Chapter 2 contain sufficient information to identify grounded properties of objects, suggesting that the representations themselves are, to some extent, grounded. Unfortunately, such simple computational models lack the large array of processing mechanisms that exist in the brain and that are posited to participate in the computational mechanism of simulation that is central to grounded cognition. In this model, each localist feature node can be interpreted as a great simplification of the multimodal representations on which the simulation process operates. Under this interpretation, co-occurrence is consistent with the principles of grounded cognition and the co-occurrence representations serve as impoverished substitutes for the rich multimodal representations that exist in the brain.

This is not to say that the model presented in Chapter 2 is fully grounded. A truly grounded model must be able to represent the features themselves in a distributed and multimodal manner across various ersatz processing systems analogous to those found in the brain. In this dissertation, it was shown that the representations derived from language usage contain information related to the features of concepts and that this information can be exploited in a simple neural network architecture. These results support the work of Louwerse (2008) by showing that language partially encodes information about embodied properties of objects. While co-occurrence-based models do not produce representations that are truly grounded to the physical world, the representations produced by such methods are at least partially grounded and provide a practical and psychologically valid alternative to representations derived from feature norms collected from human subjects.

Bibliography

Andrews, M., Vigliocco, G., & Vinson, D. (2005). Integrating attributional and distributional information in a probabilistic model of meaning representation. In T. Honkela, V. Könönen, M. Pöllä, & O. Simula (Eds.), *Proceedings of AKRR'05, international and interdisciplinary conference on adaptive knowledge representation and reasoning*. Espoo, Finland: Helsinki University of Technology.

Andrews, M., Vigliocco, G., & Vinson, D. (2007). Evaluating the contribution of intralinguistic and extra-linguistic data to the structure of human semantic representations. In *Proceedings of the 29th annual conference of the Cognitive Science Society*.

Andrews, M., Vigliocco, G., & Vinson, D. (2009). Integrating experiential and distributional data to learn semantic representations. *Psychological Review*, *116*(3), 463–498.

Andrews, S. (1982). Phonological recoding: is the regularity effect consistent? *Memory & Cognition*, *10*, 565–575.

Andrews, S. (1992). Frequency and neighborhood effects on lexical access: Lexical similarity or orthographic redundancy. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *18*, 234–254.

Balota, D. A., & Lorch, R. F. (1986). Depth of automatic spreading activation: Mediated priming effects in pronunciation but not in lexical decision. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *12*(3), 336–345.

Barsalou, L. W. (2008). Grounded cognition. *Annual Review of Psychology*, *59*, 617–645.

Berry, M., Do, T., O'Brien, G., Krishna, V., & Varadhan, S. (1993). *SVDPACKC (Version 1.0) User's Guide* (Tech. Rep.). Knoxville, TN, USA: University of Tennesse.

Bieganski, P., Riedl, J., Cartis, J., & Retzel, E. (1994). Generalized suffix trees for biological sequence data: applications and implementation. In *Proceedings of the twenty-seventh Hawaii international conference on system sciences, 1994.* (Vol. 5, p. 35-44).

Borg, I., & Groenen, P. (2005). *Modern multidimensional scaling* (2nd ed.). New York: Springer.

Burgess, C. (1998). From simple associations to the building blocks of language: Modeling meaning in memory with the HAL model. *Behaviour Research Methods, Instruments,& Computers*, *30*, 188-198.

Burgess, C. (2000). Theory and operational definitions in computational memory models: A response to glenberg and robertson. *Journal of Memory and Language*, *43*(3), 402-408.

Burgess, C., & Conley, P. (1998a). Developing semantic representations for proper

names. In *Proceedings of the twentieth annual conference of the Cognitive Science Society* (pp. 185–190).

Burgess, C., & Conley, P. (1998b). Representing proper names and objects in a common semantic space: A computational model. *Brain and Cognition*, *40*, 67-70.

Burgess, C., & Lund, K. (1997a). Modelling parsing constraints with high-dimensional context space. *Language and Cognitive Processes*, *12*(2/3), 177-210.

Burgess, C., & Lund, K. (1997b). Representing abstract words and emotional connotation in high-dimensional memory space. In *Proceedings of the ninteenth annual conference of the Cognitive Science Society* (pp. 61–66). Hillsdale, NJ: Lawrence Erlbaum Associates.

Burgess, C., & Lund, K. (2000). The dynamics of meaning in memory. In E. Dietrich & A. Markman (Eds.), *Cognitive dynamics* (pp. 117–156). Mahwah, NJ: Erlbaum.

Chiarello, C., Burgess, C., Richards, L., & Pollock, A. (1990). Semantic and associative priming in the cerebral hemispheres: Some words do, some words don't...sometimes, some places. *Brain & Language*, *38*, 75-104.

Cohen, G., & Faulkner, D. (1986). Memory for proper names: Age differences in retrieval. *British Journal of Psychology*, *4*, 187–197.

Collins, A. M., & Loftus, E. F. (1975). A spreading-activation theory of semantic processing. *Psychological Review*, *82*(6), 407–428.

Collins, A. M., & Quillian, M. R. (1969). Retrieval time from semantic memory. *Journal of Verbal Learning and Verbal Behavior*, *8*, 240–247.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems*, *2*(4), 303–314.

Damerau, F. J. (1993). Generating and evaluating domain-oriented multi-word terms from texts. *Information Processing and Management*, *29*(4), 433–447.

de Groot, A. M. B. (1980). *Mondelinge woordassociatie normen: 100 woordassociaties op 460 Nederlandse zelfstandige naamwoorden (Oral word association norms: 100 word associations to 460 Dutch nouns)*. Lisse, The Netherlands: Swets & Zeitlinger.

de Groot, A. M. B. (1983). The range of automatic spreading activation in word priming. *Journal of Verbal Learning and Verbal Behavior*, *22*(4), 417–436.

Dice, L. R. (1945). Measures of the amount of ecologic association between species. *Ecology*, *26*(3), 297–302.

Durda, K. (2006). *Investigating the structure of semantic memory.* Unpublished master's thesis, University Of Windsor.

Durda, K., & Buchanan, L. (2008). WINDSORS: Windsor improved norms of distance and similarity of semantics. *Behavior Research Methods*, *40*(3), 705–712.

Durda, K., Buchanan, L., & Caron, R. (2009). Grounding co-occurrence: Identifying features in a lexical co-occurrence model of semantic memory. *Behaviour Research Methods*.

Durda, K., Caron, R., & Buchanan, L. (2010). An application of operational research to computational linguistics: word ambiguity. *INFOR*, *48*, 1–11.

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, *14*, 179-211.

Farach, M. (1997). Optimal suffix tree construction with large alphabets. In *Proceedings of the 38th annual symposium on foundations of computer science* (pp. 137–148). Washington, DC, USA: IEEE Computer Society.

Fellbaum, C. (Ed.). (1998). *WordNet: An electronic lexical database*. Cambridge, MA: The MIT Press.

Ferrand, L., & Alario, F.-X. (1998). Normes d'associations verbales pour 366 noms d'objets contcret [word association norms for 366 names of objects]. *L'Annee Psychologique*, *98*, 659-709.

Ferrand, L., & New, B. (2003). Mental lexicon: Some words to talk about words. In P. Bonin (Ed.), (p. 25-43). Hauppage, NY: Nova Science Publishers.

Forster, K. I., & Chambers, S. M. (1973). Lexical access and naming time. *Journal of Ver*, *12*, 627–635.

Fredkin, E. (1960). Trie memory. *Communications of The ACM*, *3*(9), 490–499.

Fredrickson, J. R., & Kroll, J. F. (1976). Spelling and sound: Approaches to the internal lexicon. *Journal of Experimental Psychology: Human Perception and Performance*, *2*, 361–379.

French, R. M., & Labiouse, C. (2002). Four problems with extracting human semantics from large text corpora. In *Proceedings of the 24th annual conference of the Cognitive Science Society* (pp. 316–322). Hillsdale, NJ: Lawrence Erlbaum Associates.

Gianotti, G., Silveri, M. C., Daniele, A., & Guistolisi, L. (1995). Neuroanatomical correlated of category-specific semantic disorders: A critical survey. *Memory*, *2*, 247–264.

Glenberg, A. M., & Kaschak, M. P. (2003). The body's contribution to language. In B. Ross (Ed.), *The psychology of learning and motivation* (Vol. 43). New York: Academic Press.

Glenberg, A. M., & Robertson, D. A. (2000). Symbol grounding and meaning: A comparison of high-dimensional and embodied theories of meaning. *Journal of Memory and Language*, *43*(3), 379-401.

Glushko, R. J. (1979). The organization and activation of orthographic knowledge in reading aloud. *Journal of Experimental Psychology: Human Perception and Perfor-*

mance, *5*, 647–691.

Griffiths, T. L., Steyvers, M., & Tenenbaum, J. B. (2007). Topics in semantic representation. *Psychological Review*, *114*(2), 211-244.

Harm, M. W., & Seidenberg, M. S. (1999). Phonology, reading acquisition, and dyslexia: Insights from connectionist models. *Psychological Review*, *106*(3), 491–528.

Harm, M. W., & Seidenberg, M. S. (2001). Are there orthographic impairments in phonological dyslexia? *Cognitive Neurpsychology*, *18*(1), 71–92.

Harm, M. W., & Seidenberg, M. S. (2004). Computing the meaning of words in reading: Cooperative division of labor between visual and phonological processes. *Psychological Review*, *111*(3), 662–720.

Harnad, S. (1990). The symbol grounding problem. *Physica D: Nonlinear Phenomena*, *42*(1-3), 335–346.

Hebb, D. O. (1949). *The organization of behavior*. New York: John Wiley & Sons.

Hinton, G. E., McClelland, J. L., & Rumelhart, D. E. (1986). Distributed representations. In D. E. Rumelhart & J. L. McClelland (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition Vol. 1: Foundations* (pp. 77–109). Cambridge, MA: MIT Press.

Hinton, G. E., & Shallice, T. (1991). Lesioning an attractor network: Investigations of acquired dyslexia. *Psychological Review*, *98*(1), 74–95.

Holland, J. H. (1992). *Adaptation in natural and artificial systems*. MIT Press.

Howell, S. R., Jankowicz, D., & Becker, S. (2005). A model of grounding language acquisition: Sensorimotor features improve lexical grammatical learning. *Journal of Memory and Language*, *53*(2), 358–276.

Huang, G.-B., Zhu, Q.-Y., & Siew, C.-K. (2004). Extreme learning machine: a new learning scheme of feedforward neural networks. In *Proceedings of the IEEE international joint conference on neural networks* (Vol. 2, p. 985-990 vol.2).

Jared, D., & Seidenberg, M. (1990). Naming multi-syllabic words. *Journal of Experimental Psychology: Human Perception and Performance*, *16*, 92–105.

Jared, F., McRae, K., & Seidenberg, M. S. (1990). The basis of consistency effects in word naming. *Journal of Memory and Language*, *29*, 687–715.

Jones, M. N., Kintsch, W., & Mewhort, D. J. K. (2006). High-dimensional semantic space accounts of priming. *Journal of Memory and Language*, *55*(4), 534-552. (Special issue on memory models)

Jones, M. N., & Mewhort, D. J. K. (2007). Representing word meaning and order information in a composite holographic lexicon. *Psychological Review*, *114*(1), 1-37.

Klepousniotou, E. (2002). The processing of lexical ambiguity: Homonymy and poly-

semy in the mental lexicon. *Brain and Language, 81*, 205-223.

Krauss, R. M. (1998). Why do we gesture when we speak? *Current Directions in Psychological Science, 7*, 54–59.

Kuhl, P. K. (2004). Early language acquisition: Cracking the speech code. *Nature Reviews: Neuroscience, 5*(11), 831–841.

Kullback, S., & Leibler, R. A. (1951). On information and sufficiency. *The Annals of Mathematical Statistics, 22*(1), 79–86.

Landauer, T. K., & Dumais, S. T. (1997). A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review, 104*(2), 211-240.

Landauer, T. K., Foltz, P. W., & Laham, D. (1998). An introduction to latent semantic analysis. *Discourse Processes, 25*, 259-284.

Lemaire, B., & Denhière, G. (2004). Incremental construction of an associative network from a corpus. In K. Forbus, D. Gentner, & T. Regier (Eds.), *Proceedings twenty-sixth annual conference of the Congnitive Science Society* (p. 825-830). Chicago.

Li, P., Burgess, C., & Lund, K. (2000). The acquisition of word meaning through global lexical co-occurrence. In *Proceedings of the 30th annual child language research forum* (p. 167-178). Stanford, CA: Center for the Study of Language and Information.

Lin, D. (1998). Automatic retrieval and clustering of similar words. In *COLING-ACL98*. Montreal, Quebec.

Livesay, K., & Burgess, C. (1997). Mediated priming in high-dimensional meaning space: What is "mediated" in mediated priming? In *Proceedings of the nineteenth annual conference of the Cognitive Science Society* (pp. 436–441). Lawrence Erlbaum Associates.

Louwerse, M. M. (2007). Symbolic or embodied representations: A case for symbol interdependency. In D. McNamara, S. Dennis, & W. Kintsch (Eds.), *Handbook of latent semantic analysis* (pp. 107–120). Mahwah, NJ: Erlbaum.

Louwerse, M. M. (2008). Embodied representations are encoded in language. *Psychonomic Bulletin and Review, 15*, 838–844.

Louwerse, M. M., Cai, Z., Hu, X., Ventura, M., & Jeuniaux, P. (2006). Cognitively inspired natural-language based knowledge representations: Further explorations of latent semantic analysis. *International Journal of Artificial Intelligence Tools, 15*, 1021–1039.

Lowe, W. (2000). Towards a theory of semantic space. In *Proceedings of COGSCI 2000* (p. 576-581). Lawrence Erlbaum Associates.

Lund, K., & Burgess, C. (1996). Producing high-dimensional semantic spaces from lexical co-occurrence. *Behaviour Research Methods, Instruments, & Computers*, *28*(2), 203-208.

Martin, A., & Chao, L. L. (2001). Semantic memory and the brain: Structure and processes. *Current Opinion in Neurobiology*, *11*, 194–201.

McDonald, S. A., & Shillcock, R. C. (2001). Rethinking the word frequency effect: The neglected role of distributional information in lexical processing. *Language and Speech*, *44*(3), 295–323.

McKoon, G., & Ratcliff, R. (1992). Spreading activation versus compound cue accounts of priming: Mediated priming revisited. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *18*(6), 1155–1172.

McNamara, T. P. (1992). Theories of priming: I. associative distance and lag. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *18*(6), 1172–1190.

McNamara, T. P., & Altarriba, J. (1988). Depth of spreading activation revisited: Semantic mediated priming occurs in lexical decisions. *Journal of Memory and Language*, *27*, 545-559.

McNamara, T. P., & Holbrook, J. B. (2003). Semantic memory and priming. In A. F. Healy & R. W. Proctor (Eds.), *Handbook of psychology: Experimental psychology, vol. 4* (pp. 447–474). Hoboken, NJ: John Wiley & Sons.

McRae, K., Cree, G. S., Seidenberg, M. S., & McNorgan, C. (2005). Semantic feature production norms for a large set of living and nonliving things. *Behaviour Research Methods, Instruments, & Computers*, *37*, 547-559.

McRae, K., Cree, G. S., Westmacott, R., & de Sa, V. R. (1999). Further evidence for feature correlations in semantic memory. *Canadian Journal of Experimental Psychology*, *53*(4), 360–373.

McRae, K., de Sa, V. R., & Seidenberg, M. S. (1997). On the nature and scope of featural representations of word meaning. *Journal of Experimental Psychology: General*, *126*(2), 99–130.

Meyer, D. E., & Schvaneveldt, R. W. (1971). Facilitation in recognizing pairs of words: Evidence of a dependence between retrieval operations. *Journal of Experimental Psychology*, *90*, 227-334–.

Monsell, S., Doyle, M. C., & Haggard, P. N. (1989). Effects of frequency on word recognition tasks: Where are they. *Journal of Experimental Psychology: General*, *118*, 43–71.

Moss, H. E., Ostrin, R. K., Tyler, L. K., & Marslen-Wilson, W. D. (1995). Accessing different types of lexical semantic information: Evidence from priming. *Journal of*

Experimental Psychology: Learning, Memory, and Cognition, *21*(4), 863-883.

Nelson, D. L., McEvoy, C. L., & Schreiber, T. A. (1999). *The University of South Florida word association norms.* (Retrieved from `http://w3.usf.edu/FreeAssociation`)

Pantel, P., & Lin, D. (2002). Discovering word senses from text. In *KDD'02: Proceedings of the eighth ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 613–619). New York, NY, USA: ACM.

Peereman, R., & Content, A. (1995). Neighbourhood size effect in naming: Lexical activation or sublexical correspondences? *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *21*(2), 409–421.

Pereira, F., Tishby, N., & Lee, L. (1993). Distributional clustering of English words. In *Proceedings of the thirty-first annual meeting of the ACL* (p. 183-190).

Pexman, P. M., Holyk, G. G., & Monfils, M.-H. (2003). Number-of-feature effects and semantic processing. *Memory & Cognition*, *31*(6), 842–855.

Plaut, D. C. (1995). Semantic and associative priming in a distributed attractor network. In *Proceedings of the seventeenth annual conference of the Cognitive Science Society* (pp. 37–42). Hillsdale, NJ: Lawrence Erlbaum Associates.

Plaut, D. C., & Booth, J. R. (2000). Individual and developmental differences in semantic priming: Empirical and computational support for a single-mechanism account of lexical processing. *Psychological Review*, *107*, 786 – 823.

Plaut, D. C., McClelland, J. L., Seidenberg, M. S., & Patterson, K. (1996). Understanding normal and impaired word reading: Computational principles in quasi-regular domains. *Psychological Review*, *103*(1), 56-115.

Plaut, D. C., & Shallice, T. (1993). Deep dyslexia: A case study of connectionist neuropsychology. *Cognitive Neuropsychology*, *10*(5), 377 – 500.

Pulvermüller, F. (2005). Brain mechanisms linking language and action. *Nature Reviews Neuroscience*, *6*, 576–582.

Quillian, M. R. (1968). Semantic information processing. In M. Minksy (Ed.), (pp. 227–270). Cambridge, MA: MIT Press.

Razavi, A. H., Matwin, S., Inkpen, D., & Kouznetsov, A. (2009). Parameterized contrast in second order soft co-occurrences: A novel text representation technique in text mining and knowledge extraction. In Y. Saygin et al. (Eds.), *ICDM workshops* (p. 471-476). IEEE Computer Society.

Riordan, B., & Jones, M. N. (2011). Redundancy in preceptual linguistic experience: Exploring the cohesion of semantic categories in feature-based and distributional models of semantic representation. *Topics in Cognitive Science*, *3*(2), 303–345.

Rips, L. J., Shoben, E. J., & Smith, E. E. (1973). Semantic distance and the verification of semantic relations1. *Journal of Verbal Learning and Verbal Behavior*, *12*(1), 1–20.

Rodd, J. M. (2004). The effect of semantic ambiguity on reading aloud: A twist in the tale. *Psychonomic Bulletin & Review*, *11*(3), 440-445.

Rodd, J. M., Gaskell, M. G., & Marslen-Wilson, W. D. (2002). Making sense of semantic ambiguity: Semantic competition in lexical access. *Journal of Memory and Language*, *46*, 245-266.

Rodd, J. M., Gaskell, M. G., & Marslen-Wilson, W. D. (2004). Modelling the effects of semantic ambiguity in word recognition. *Cognitive Science*, *28*, 89-104.

Rogers, T. T., & McClelland, J. L. (2004). *Semantic cognition: A parallel distributed processing approach*. Cambridge, MA: MIT Press.

Roget, P. M. (1911). *Roget's thesaurus of English words and phrases (1911 ed.)*. (Retrieved from `http://www.gutenberg.org/ebooks/10681`)

Rosch, E. H. (1975). Cognitive representation of semantic categories. *Journal of Experimental Psychology*, *104*(3), 192-233.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning internal representations by error propagation. In D. E. Rumelhart & J. L. McClelland (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition Vol. 1: Foundations* (p. 318-362). Cambridge, MA, USA: MIT Press.

Saffran, E. M. (2000). The organization of semantic memory: In support of a distributed model. *Brain & Language*, *71*(1), 204–212.

Saffran, E. M., & Schwartz, M. F. (1994). Of cabbages and things: Semantic memory from a neuropsychological perspective. In M. Umilta & M. Moscovitch (Eds.), *Attention and performance XV: Conscious and nonconscious information processing* (pp. 507–536). Cambridge, MA: MIT Press.

Saffran, E. M., & Sholl, A. (1999). Clues to the functional and neural architecture of word meaning. In C. Brown & P. Hagoort (Eds.), *The neurocognition of language* (pp. 241–272). Oxford: Oxford University Press.

Schütze, H. (1992). Dimensions of meaning. In *Supercomputing '92* (p. 787-796). Minneapolis, MN.

Schütze, H. (1998). Automatic word sense discrimination. *Computational Linguistics*, *24*(1), 97-123.

Searle, J. (1980). Minds, brains, and programs. *Behavioral and Brain Sciences*, *3*(03), 417–424.

Sears, C. R., Hino, Y., & Lupker, S. J. (1995). Neighborhood size and neighborhood frequency effects in word recognition. *Journal of Experimental Psychology: Human*

Perception and Performance, *21*, 876–900.

Sears, C. R., Siakaluk, P. D., Chow, V. C., & Buchanan, L. (2008). Is there an effect of print exposure on the word frequency effect and the neighborhood size effect? *Journal of Psychological Research*, *37*, 269–291.

Seidenberg, M. S., & McClelland, J. L. (1989). A distributed, developmental model of word recognition and naming. *Psychological Review*, *96*(4), 523-568.

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, *27*, 379-423, 623-656.

Shaoul, C., & Westbury, C. (2006). Word frequency effects in high-dimensional co-occurrence models: A new approach. *Behaviour Research Methods*, *38*(2), 190-195.

Shaoul, C., & Westbury, C. (2008). Performance of hal-like word space models on semantic clustering. In M. Baroni, S. Evert, & A. Lenci (Eds.), *ESSLLI workshop on distributional lexical semantics* (pp. 42–46). Hamburg, Germany.

Shaoul, C., & Westbury, C. (2009). *The Westbury Lab Wikipedia corpus (2010)*. Edmonton, AB: University of Alberta. (downloaded from `http://www.psych.ualberta.ca/~westburylab/downloads/westburylab.wikicorp.download.html`)

Shaoul, C., & Westbury, C. (2010). Exploring lexical co-occurrence space using HiDEx. *Behavior Research Methods*, *42*(2), 393-413.

Sibley, D. E., Kello, C. T., Plaut, D. C., & Elman, J. L. (2008). Large-scale modeling of wordform learning and representation. *Cognitive Science*, *32*, 741–754.

Sitnikova, T., West, W. C., Kuperberg, G. R., & Holcomb, P. J. (2006). The neural organization of semantic memory: Electrophysiological activity suggests feature-based segregation. *Biological Psychology*, *71*(3), 326 – 340.

Sloman, S. A. (1998). Categorical inference is not a tree: The myth of inheritance hierarchies. *Cognitive Psychology*, *35*(1), 1–33.

Steyvers, M., & Tenenbaum, J. B. (2005). The large-scale structure of semantic networks: Statistical analyses and a model of semantic growth. *Cognitive Science*, *29*, 41–78.

Taft, M., & Russell, B. (1992). Pseudohomophone naming and the word frequency effect. *The Quarterly Journal of Experimental Psychology*, *45A*(1), 51–71.

Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, *59*(236), 433–460.

Turney, P. D. (2001a). Answering subcognitive Turing Test questions: A reply to French. *Journal of Experimental & Theoretical Artificial Intelligence*, *13*(4), 409–419.

Turney, P. D. (2001b). Mining the Web for Synonyms: PMI-IR versus LSA on TOEFL. In *Proceedings of the 12th European conference on machine learning* (pp. 491–502). Springer-Verlag.

Turney, P. D. (2012). Domain and function: A dual-space model of semantic relations

and compositions. *Journal of Artificial Intelligence Research*, *44*, 533-585.

Twilley, L. C., Dixon, P., Taylor, D., & Clark, K. (1994). University of Alberta norms of relative meaning frequency for 566 homographs. *Memory & Cognition*, *22*(1), 111-126.

Van de Cruys, T., Rimell, L., Poibeau, T., & Korhonen, A. (2012). Multi-way tensor factorization for unsupervised lexical acquisition. In M. Kay & C. Boitet (Eds.), *COLING 2012: 24th international conference on computational linguistics* (p. 2703-2720). Indian Institute of Technology Bombay.

Vigliocco, G., Warren, J., Siri, S., Arciuli, J., Scott, S., & Wise, R. (2006). The role of semantics and grammatical class in the neural representation of words. *Cerebral Cortex*, *16*, 1790–1796.

Vinson, D. P., & Vigliocco, G. (2008). Semantic feature production norms for a large set of objects and events. *Behavior Research Methods*, *40*(1), 183–190.

Weiner, P. (1973). Linear pattern matching algorithms. In *SWAT '08: IEEE conference record of 14th annual symposium on switching and automata theory* (pp. 1–11).

Werbos, P. J. (1990). Backpropagation through time: What it does and how to do it. *Proceedings of the IEEE*, *78*(10), 1550-1560.

Westbury, C., & Buchanan, L. (2002). The probability of the least likely non length-controlled bigram affects lexical decision RTs. *Brain and Language*, *81*(1–3), 66–78.

Whaley, C. P. (1978). Word-nonword classification time. *Journal of Verbal Learning and Verbal Behavior*, *17*, 143–154.

Williams, J. N. (1996). Is automatic priming semantic? *European Journal of Cognitive Psychology*, *8*(2), 113–161.

Appendices

Appendix A Subset of Rosch (1975) Norms

Furniture	Fruit	Vehicle	Weapon	Vegetable
chair	apple	automobile	gun	pea
sofa	banana	truck	pistol	carrot
couch	peach	car	revolver	spinach
table	pear	bus	machine-gun	broccoli
dresser	apricot	taxi	rifle	asparagus
coffee-table	tangerine	jeep	switchblade	corn
desk	plum	ambulance	knife	cauliflower
bed	grapes	motorcycle	dagger	lettuce
davenport	strawberry	streetcar	shotgun	celery
bookcase	grapefruit	van	sword	cucumber
Tools	**Bird**	**Sport**	**Toy**	**Clothing**
hammer	robin	football	doll	pants
screwdriver	sparrow	baseball	yo-yo	shirt
drill	bluebird	basketball	marbles	dress
sandpaper	canary	tennis	rattle	skirt
sander	blackbird	softball	teddy-bear	blouse
toolbox	dove	canoeing	dollhouse	suit
T-square	lark	handball	ball	slacks
chisel	parakeet	rugby	jacks	jacket
rasp	mockingbird	hockey	wagon	coat
hacksaw	wren	swimming	kite	sweater

Table A.1: Ten highly ranked exemplars from each category of the Rosch (1975) category norms. These stimuli are the highest ranked exemplars that did not appear in any other category and that appeared in the vocabulary of the co-occurrence model.

Appendix B Part-of-speech Stimuli

Adjectives				
yummy	delicious	tasty	sweet	bitter
sour	salty	slippery	slimy	spiky
prickly	smooth	rough	sticky	soft
hard	wet	dry	furry	sad
happy	funny	boring	nasty	naughty
angry	mean	nice	beautiful	pretty
lovely	friendly	grumpy	scary	lonely
loud	noisy	quiet	slow	fast
poor	rich	strong	weak	old
new	young	lazy	sleepy	tired
furry	tall	short	round	fat
long	skinny	thin	thick	smelly
big	little	tiny	small	huge
enormous	gigantic	large	yellow	red
orange	blue	purple	brown	black
white	green	pink	one	two
three	four	five	six	seven
eight	nine	ten		

Table B.1: Adjectives used in part-of-speech MDS.

Adverbs				
quickly	slowly	quietly	loudly	gently
softly	gracefully	carefully	neatly	easily
truthfully	kindly	bravely	scarily	sleepily
excitedly	energetically	safely	loosely	cheerfully
happily	angrily	lightly	silently	sweetly
brightly	rudely	nervously	anxiously	cleverly
healthily	naturally	deeply	heavily	correctly
colourfully	colorfully	playfully	fiercely	lazily

Table B.2: Adverbs used in part-of-speech MDS.

Nouns				
apples	babies	balls	beds	bears
boys	bells	birds	brothers	boats
giants	dinosaurs	cakes	cars	cats
children	corn	chairs	chickens	cows
dogs	wind	dolls	frogs	ducks
eggs	eyes	snails	waves	lizards
feet	clouds	fish	trains	flowers
pets	books	girls	snakes	grass
pies	hands	pizzas	oranges	bikes
horses	houses	kittens	legs	letters
ants	men	tomatoes	money	teeth
mice	friends	spiders	pigs	rabbits
rain	rings	clocks	fairies	planes
songs	sheep	shoes	sisters	trees
plants	trucks	sticks	sun	toys
things				

Table B.3: Nouns used in part-of-speech MDS.

Verbs				
creep	crawl	walk	run	jump
skip	hop	slither	climb	dig
squirm	fly	sit	stalk	stomp
tiptoe	gallop	blow	dance	glide
swim	wash	play	throw	drink
eat	chew	sing	shout	growl
bark	buzz	laugh	smile	cry
go	moo	quack	talk	yell
scream	screech	squawk	squeal	glow
listen	paint	look	read	knit
sleep	draw	shine	watch	kick
dive	find	build	work	explore
shop	clean	catch	shake	

Table B.4: Verbs used in part-of-speech MDS.

Vita Auctoris

Name:	Kevin Durda
Place of Birth:	Windsor, Ontario
Year Of Birth:	1979
Education:	University Of Windsor, Windsor, Ontario 1998 - 2002 B.C.S.
	University Of Windsor, Windsor, Ontario 2002 - 2003 B.M.H.
	University Of Windsor, Windsor, Ontario 2003 - 2006 M.Sc
	University Of Windsor, Windsor, Ontario 2006 - 2013 Ph.D.