Generating Realistic Semantic Codes for Use in Neural Network Models

Ya-Ning Chang (ya-ning.chang@postgrad.manchester.ac.uk)
Neuroscience and Aphasia Research Unit (NARU), University of Manchester,
Brunswick Street, Manchester, M13 9PL,UK

Steve Furber (steve.furber@manchester.ac.uk)
Advanced Processor Technologies Group, University of Manchester, Oxford Road, Manchester, M13 9PL, UK

Stephen Welbourne (stephen.welbourne@manchester.ac.uk)
Neuroscience and Aphasia Research Unit (NARU), University of Manchester,
Brunswick Street, Manchester, M13 9PL,UK

Abstract

Many psychologically interesting tasks (e.g., reading, lexical decision, semantic categorisation and synonym judgement) require the manipulation of semantic representations. To produce a good computational model of these tasks, it is important to represent semantic information in a realistic manner. This paper aimed to find a method for generating artificial semantic codes, which would be suitable for modelling semantic knowledge. The desired computational criteria for semantic representations included: (1) binary coding; (2) sparse coding; (3) fixed number of active units in a semantic vector; (4) scalable semantic vectors and (5) preservation of realistic internal semantic structure. Several existing methods for generating semantic representations were evaluated against the criteria. The correlated occurrence analogue to the lexical semantics (COALS) system (Rohde, Gonnerman & Plaut, 2006) was selected as the most suitable candidate because it satisfied most of the desired criteria. Semantic vectors generated from the COALS system were converted into binary representations and assessed on their ability to reproduce human semantic category judgements using stimuli from a previous study (Garrard, Lambon Ralph, Hodges & Patterson, 2001). Intriguingly the best performing sets of semantic vectors included 5 positive features and 15 negative features. Positive features are elements that encode the likely presence of a particular attribute whereas negative features encode its absence. These results suggest that including both positive and negative attributes generates a better category structure than the more traditional method of selecting only positive attributes.

Keywords: semantics; semantic representations; neural networks; computational modelling; connectionist models.

Introduction

Computational models are frequently used to simulate human behavioural data and help understand the underlying cognitive processes. Any type of computational model requires decisions to be made about what representation scheme to use. Semantic representations are particularly important for models of many linguistic processes including spoken and written language.This paper aims to find a method of generating semantic representations, which can fulfil a set of requirements derived from the constraints imposed by incorporating semantic knowledge within a large-scale connectionist model. A list of criteria that we considered essential for sophisticated and efficient simulation using a connectionist model includes: (1) Binary coding: a binary coding scheme is essential for use in connectionist models because the models consist of many neuron-like units whose activation values vary between 0 and 1. The models are trained to match their activation values to predefined targets, which need to be at the extreme ends of the possible activations; (2) Sparse coding: a sparse coding scheme is one in which an item is represented by using a small number of active units in each vector. A sparse representation is attractive from a computational viewpoint because it allows efficient computation. By controlling sparseness, the redundancy of a code can be minimized and learning is generally fast and relatively easy. Importantly, it is likely to reflect the natural structure of the representation system in the brain; (3) Fixed number of active units in each vector: the idea of using a fixed number of active units in each semantic vector is not common in most existing coding schemes. However, it has an advantage that this coding is uniform and it makes sense to think about how similar items are by measuring the Euclidian distance between them – if items have different numbers of features then measuring Euclidian distance does not give a good indication of similarity (Furber, Bainbridge, Cumpstey & Temple, 2004). For connectionist models, there is a particular reason to want to adopt a fixed number of active units, which is that only the active units can contribute to activation in later layers. Units with a zero level of activation do not propagate information in the network and therefore do not generate any weight updates in response to the error signal; (4) Scalable semantic vectors: to keep the simulations computationally tractable, it is important to keep the size of semantic vectors manageable. Vector size is an important design consideration because it determines how many units in the model are needed for modelling semantic knowledge. Given a code length n, it has a maximum theoretical number of items that it can code for, which is 2^n. The capacity increases dramatically as the code length grows. Thus, the selection of the vector length also needs to consider the number of items to be represented; (5) Preservation of inherent semantic structure: the most important criterion is that the semantic vectors can support human-like semantic classifications. They need to preserve

the inherent semantic structure of the lexicon. Words which are semantically similar should be represented by vectors that are relatively close in the semantic space; by contrast, semantically unrelated words should tend to be far from each other. Preserving these semantic relationships will allow for the possibility of modelling tasks like categorization and synonym judgment, which are commonly used to probe semantic effects.

Review of existing semantic representation schemes

Several semantic representation schemes have been proposed either for behavioural studies or for use in computational modelling (Dilkina, McClelland & Plaut, 2008; Harm & Seidenberg, 2004; Plaut, 1997; Rogers, Lambon Ralph, Garrard, Bozeat, McClelland, Hodges & Patterson, 2004). These schemes are based on different techniques and there does not seem to be a consensus view as to how to produce a set of representations. It is therefore important to review these competing coding schemes from a modelling perspective using the criteria described above.

Feature Norms One traditional method is to obtain the feature norms through experiments (e.g., McRae, Cree, Seidenberg & McNorgan, 2005). In these experiments, subjects are given a list of words and asked to write down attributes about each word. To categorise the attributes and make proper constraints on subjects' responses, some lexical relations such as "is" and "has" are used to prompt subjects to list the features of the stimulus word. The most commonly listed features for a particular word are then considered as the core semantic attributes for that word. The collected attributes for an item can be easily converted into binary codes with the presence of an attribute coded as "1" and the absence as "0". Controlling for sparseness is not so easy, but it may be possible to rank the features by the number of subjects that identified them, and use this as a method for deciding which features to drop. Moreover, this method is not very flexible and practically can only be used for a small set of words.

Arbitrary Features Another way to generate semantic representations is to use random features. Features for a word are assigned randomly but the assignments may still respect broader aspects of semantic knowledge such as category knowledge. For example, the words within the same category can be designed to share more features than words belonging to different categories. This method has been applied to various computational studies designed to capture abstract semantic properties including simulations of lexical decision (e.g., Plaut, 1997) and semantic impairment (e.g., Rogers et al., 2004). The features for an item are assigned manually and are binary codes. The control of sparseness can be achieved by adjusting the fraction of the number of active units in a vector over the code length. The fixed number of active units in a vector is also controllable. In addition, the size of vector length is scalable and determined by the modellers. Although this coding scheme is good for producing coarsely structured semantic representations, it is not easily scalable and it would be very difficult to generate an artificial semantic structure that can capture the complexity found in human semantics.

Co-occurrence Statistics Semantic representations can also be derived from very large text corpora by evaluating which words appear in similar types of documents or co-occur within a fixed window. Several semantic representation schemes have been developed on the basis of this statistical co-occurrence including Latent Semantic Analysis (LSA) (e.g., Landauer, Foltz & Laham, 1998), Hyperspace Analogue to Language (HAL) (e.g., Lund & Burgess, 1996) and Correlated Occurrence Analogue to Lexical Semantics (COALS) (Rohde et al., 2006). These methods are all based on similar ideas but they are slightly different in the ways they collect data and deal with the high-dimensional co-occurrence matrices. LSA derives vectors based on a collection of segmented documents in which the number of occurrences of a word in various types of documents is computed as an element in the high-dimensional co-occurrence matrix. The dimensionality of the matrix is then reduced by using Singular Value Decomposition (SVD) while preserving the semantic relations between words as much as possible. Unlike LSA, the derivations in both HAL and COALS are based on words co-occurring within a fixed window in an un-segmented document. The key differences between these three systems are in their ways of expressing the tendency of two words to co-occur: LSA computes the cosines between the vectors of two words, HAL uses distance measure and COALS uses the correlation measure. In addition, HAL reduces the dimensionality of the matrices by eliminating all but the few thousand columns with the largest variant values, which is different from the SVD technique adopted by both LSA and COALS (see Rohde et al., 2006 for more detailed comparisons).

The semantic vectors generated by reducing a high-dimensional matrix are typically real-value vectors but COALS also provides binary-valued vectors. For the other two systems, however, it is relatively easy to convert the real values to binary values by thresholding. The vectors with values greater than a certain level are replaced with the value "1" and all others are replaced with the value "0". Sparse coding can be enforced by adjusting the threshold level used when converting real-value vectors into binary. Similarly, the fixed number of active units in a vector can be designed by modellers during the binarization process by restricting the number of 1's to the top n elements of the vector. By using the co-occurrence statistics, the sizes of semantic vectors for lexical items are scalable, which is particularly suitable for the computational modellers seeking a set of representations with low computational cost. The key advantage of this scheme is that it should be able to generate realistic semantic codes for any word lists of any length provided that the latent semantic information contained in the structure of large corpora is sufficiently

detailed and can be extracted efficiently.

WordNet WordNet is an online semantic database, which was developed by Miller in 1990. Information in WordNet is organised by many synonymous sets. These sets are linked by their lexical relations such as "is a"or "is part of" relations. A unique feature of WordNet is that it provides multiple word senses, which can be obtained from the database separately while other semantic systems do not distinguish between word senses. Similar to the feature norms, the attributes generated from WordNet have direct semantic interpretations. The semantic vectors generated by WordNet are binary to represent the presence and the absence of attributes, and generally rather sparse. But the number of semantic features for each word is not fixed and the range could be very wide. The size of the semantic vectors is less flexible because the size depends on how words relate to each other within the word list of interest. As a general rule the longer the list of lexical items to be coded the longer resultant semantic vector. Since the vectors are, directly derived from many synonymous sets in Word Net based on the researchers' semantic knowledge, the semantic structure is likely to be well preserved.
Table 1 summarises the results of these evaluations. Among these, COALS appears to be the best choice because it satisfied most of the criteria than other systems.

Table 1: Summary of the evaluations of different semantic representation schemes

	Feature Norms	Arbitrary Features	Co-occurrence Statistics			Word Net
			LSA	HAL	COALS	
Criterion 1 (Binary coding)	√	√	Δ	Δ	√	√
Criterion 2 (Sparse Coding)	Δ	Δ	Δ	Δ	Δ	√
Criterion 3 (Fixed Number of Active Units)	X	Δ	Δ	Δ	Δ	X
Criterion 4 (Scalability)	X	√	√	√	√	X
Criterion 5 (Semantic Structure)	√	X	√	√	√	√

Note: √: good fit; Δ: can be adapted to fit; X: poor fit or difficult to support

Method

The correlated occurrence analogue to the lexical semantics (COALS) system (Rohde et al., 2006) is designed to be very flexible. Although two of the criteria (i.e., sparse coding and fixed number of active units) are outside the scope of the original COALS system, they could be easily achieved by manipulating the semantic vectors generated from the system. However, it is crucial to examine whether the semantic codes generated from COALS preserve enough semantic structure that they can be used to predict the human semantic data. In addition it is important to investigate the best method of transforming the COALS vectors into binary codes. To generate binary vectors Rohde and colleagues simply set negative components to 0 and positive components to 1 based on the original real-valued vector from the SVD. This means that information contained in negative parts of the vector is lost. Thus the questions asked here are whether negative components also contribute to a good semantic similarity structure and, if so, what is the optimum number of positive and negative features required to produce a best fit to human data. The following sections describe how to generate semantic vectors based on COALS in a way that satisfies all the criteria discussed previously. We then go on to compare the performance of the vectors on a semantic categorisation task using human data taken from Garrard et al.'s (2001) categorisation study.

Generating Semantic Vectors based on COALS

To explore whether negative components were as important as positive components, a binarization process of coding both positive and negative components were used. A 100-dimensional semantic vector was generated for all items in the Garrard et al.'s (2001) word list except one item "watering can", which was discarded because the system did not support the compound words. The 100-dimensional vector was duplicated to create two 100-dimensional vectors with the first 100 dimensions coding the positive elements and the second 100 dimensions coding the negative elements. The two 100-dimensional vectors were combined into a 200-dimensional semantic vector. The first half of the 200-dimensional vector contained only the positive components and the second half of the 200-dimensional vector contained only the absolute values of negative components. Assuming the best number of positive and negative components is n and m respectively then the top n values of the first half of the combined vector and the top m values of the second half of the vector were selected as the key features. All the selected features were set to 1 and others are 0. This resulted in a 200-dimensional binary vector with the property that matching elements from the two halves of this vector (e.g., the 1st and 101st elements) code for the same feature and have a special relationship whereby if one is on then the other must be off. (Note: both paired vectors can be off indicating that neither the presence nor the absence of this feature is particularly important to the meaning of the item.)

Testing Procedure

To determine the usefulness of the binary semantic codes we examined the relationship between the category structures derived from the artificially generated sets of semantic vectors and human data taken from Garrard et al.'s (2001) study. In their study, Garrard and colleagues asked subjects to categorise items into a living thing group and nonliving thing group. On a finer scale, the living thing group can be divided into animals, birds and fruit and the nonliving thing group also can be subdivided into household objects, tools and vehicles. There were in total 6 subgroups. Semantic vectors for 61 items in the Garrard et al.'s (2001) list were generated using the method described above. Each vector had a length of 200. The n features of the first half of

the semantic vector represent the important positive features and the *m* features of the second half of the semantic vector indicate the important negative features. The numbers of positive features (n) and negative features (m) were varied to determine the optimum values of n and m.

Evaluation of semantic vectors

Two parameters based on semantic distances between words can be used to evaluate the match with the semantic structure in the human data: distance validity index (*DVI*) and distance ratio (*DR*). *DVI* counts the number of groups where the within group distance (i.e., the averaged Euclidean distance between items in the same group) is smaller than all the between group distances (i.e., the averaged Euclidean distance between items in the different groups). The larger the value of *DVI* the better the semantic categories have been partitioned. This is rather coarse measure of semantic structure and for this data the value of *DVI* ranges from 1 to 6 (i.e., the number of subgroups). The expected best value is *DVI*=6 indicating that all the within group distances are smaller than between group distances. *DR* computes the average of all the distance ratios. The distance ratio is the sum of between group distances to the sum of within group distances. Ideally there will be a larger between group distance and a smaller within group distance so that *DR* should be as large as possible. It should be noted that the value of *DR* is also positively correlated with the total number of features within a vector because it is computed on the basis of the Euclidean distance. The distance for the vectors having more features is generally larger than that for the vectors having less features. This indicates that *DR* is only a useful comparator for code sets with the same number of features.

Thus far we have tacitly assumed that the subgroups will be exactly the same as those from human data. However, even if a set of semantic codes can be shown to have a *DVI* of 6 and a high *DR*, it cannot be guaranteed that all its items would actually be categorized into the correct groups based solely on their intrinsic correlations. To evaluate this we needed to test whether the clustering results based on the intrinsic correlations among semantic vectors were similar to semantic categories from human data. We tested this by using the adjusted rand index (*ARI*) (Hubert & Arabie, 1985). *ARI* is commonly used to measure the similarity between two different ways of partitioning a set of items. To compare the partitions of human data and the artificial semantic codes, *ARI* counts the number of agreements and disagreements between them. It ranges from 0 to 1, with 0 indicating the two partitions are completely different and with 1 indicating the two partitions are exactly the same.

All three indices (*DVI, DR* and *ARI)* were used to evaluate the semantic vectors. The maximum number of active features including positive and negative in a semantic vector was set to 40 and the minimum was 10. Thus, the population sparseness ranged from 0.05 to 0.2. The numbers of positive (*n*) and negative (*m*) features varied in a complementary manner which was dependent on the total active features (*t*). To find out what were the optimum numbers of *n* and *m*, 24 different combinations of positive and negative features were assessed by using the three indices: *DVI, DR* and *ARI*. The evaluations were performed in two steps. The first step was to compare different sets of semantic vectors based on the predefined categories by choosing the combinations with a *DVI* of 6 and using the *DR* score to select the top candidates within groups with the same number of features. The second step was to use the *ARI* score as an independent additional test to confirm that the candidate with the highest *ARI* was also one of the possible candidates from the first step.

Results

Searching the best semantic vectors

Table 2 shows the results of the six candidates from 24 sets of semantic codes with different combinations of positive and negative features, in which they had the maximum possible number of *DVI* (6). The set with 5 positive and 15 negative features (ID 2) had the largest value of *ARI*. This set was one of those with the maximum value of *DVI* and the value of *DR* for this set was also larger than that for other candidate sets with the same total number of features. These results suggest that for this application a set of semantic vectors with 5 positive features and 15 negative features best captures the semantic categories generated from human data. It is also interesting to note that among the top candidates ID 1 was the only one with only positive features and its *ARI* was much lower than that for all the other possible candidates. The differences between *ARI* for the top candidate and for the two other candidates (ID 3 and ID 5) which included both positive and negative features were relatively small, suggesting that the exact number of positive and negative features may not be critical. However the majority of candidate codes (4/6) did have more negative than positive features.

Table 2: Results of searching the best semantic vectors

ID	Total	Positive	Negative	*DVI*	*DR*	*ARI*
1	10	10	0	6	5.62	0.38
2	20	5	15	6	5.74	**0.57**
3	20	10	10	6	5.70	0.54
4	30	5	25	6	5.83	0.45
5	30	10	20	6	5.77	0.52
6	40	5	35	6	5.84	0.49

Note: ID: identification; *DVI*: distance validity index; *DR*: distance ratio; *ARI*: adjusted rand index

To further test the significance of including negative codes, 20 sets of semantic codes for each of three groups (positive, neutral and negative-biased) were generated. Each set had the same number of features, ranging from 10 to 48. In the positive group, the vector included only positive

features whereas the vector in the neutral group had an equal number of positive and negative features. In the negative-biased group, the vector had more negative than positive features with a ratio of 3:1. One-tailed paired t-tests were conducted to compare both the neutral and negative-biased groups to the positive group, where the three indices were used as dependent variables. As predicted, the *DVI*s and *ARI*s of the neutral and negative-biased groups were significantly higher than that for the positive group (Table 3). For the *DRs*, the difference between the negative-biased and the positive groups was not significant while there was a significantly lower mean *DR* for the neutral group than for the positive group. The comparison between the negative-bias and neutral groups showed that both *DVI* and *DR* were higher for the negative-biased group than for the neutral group and there was no difference in their *ARIs*. Overall the results demonstrated the negative-biased group was superior to both the positive and neutral groups, confirming that the inclusion of negative codes is important to capturing the way semantic knowledge is represented in humans.

Table 3: Results of Significance Tests

Group	Index	MD	MSE	t Value	df	*P* Value
Neutral -Positive	*DVI*	0.05	0.01	3.63	19	.001*
	DR	-0.10	0.03	-3.08	19	.003*
	ARI	0.45	0.14	3.33	19	.002*
Negative-Biased -Positive	*DVI*	0.04	0.02	1.78	19	.046*
	DR	-0.04	0.03	-1.25	19	.114
	ARI	0.75	0.12	6.10	19	<.001*
Negative-Biased -Neutral	*DVI*	0.30	0.11	2.85	19	.005*
	DR	0.06	0.01	7.58	19	<.001*
	ARI	-0.02	0.02	-0.87	19	.197

Note: *DVI*: distance validity index; *DR*: distance ratio; *ARI*: adjusted rand index
MD: mean difference; MSE: mean standard error; *: *P* value is significant at the .05 level.

Hierarchical Clustering Analysis

Figure 1 shows the hierarchical clustering based on the optimum vectors indicated in Table 2.

The y axis shows the Jaccard's distance, a measure of similarity between words. The lower the value the more similar the clusters are. The semantic vectors can accurately represent the semantic categories at a coarse scale, which means that the living things and nonliving things are well separated. To compare the clustering results with human data collected by Garrard et al. (2001), the items in Figure 1 were coloured according to its category in the human semantic data. This clearly shows whether the clustering results were consistent with human categories. Ideally, items with the same colour would be clustered together, indicating the items are clustered into the same group as in the human category. Most of the items are correctly clustered. However there are a few interesting exceptions, for example, the word "chicken" was clustered into the fruit category (items coloured in purple) based on the artificial semantic codes, while it should have been clustered into the bird category (Orange). Presumably this is because in many texts the word "chicken" might more frequently co-occur with other food (including fruit) in the kitchen context so this category might be more accurately described as food. Within the nonliving things, it appears that the broader category of tool is well distinguished from the vehicle group but the boundaries between tools and household items is less clear. It is likely that most of the tools and household objects tend to occur in a similar context in the text so it would be difficult to differentiate them in a fine scale by using the co-occurrence statistic approach.

General Discussion

Several schemes for generating semantic codes have been reviewed in this paper with a focus on the requirements of computational modelling. The primary aim was to determine an appropriate system for representing semantic knowledge, which could be used for a large-scale computational modelling of semantic related tasks.

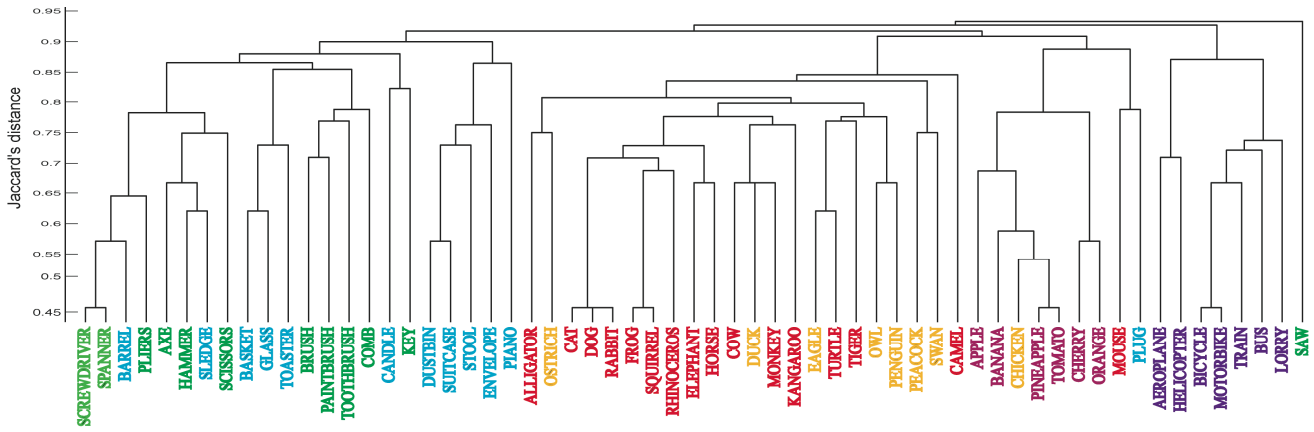

Figure 1: Hierarchical cluster analysis of 61 words based on the 200-dimensional semantic vectors with 5 positive and 15 negative features. (Items coloured base on human categories: Animal (Red), Birds (Orange), Fruit (Purple), Tool (Green), H'hold (Dark Blue),Vehicles (Purple)).

The desired computational criteria were as follows: (1) binary coding; (2) sparse coding; (3) fixed number of active units in a vector; (4) scalable vectors; (5) preservation of inherent semantic structure. The COALS system (Rohde et al., 2006) provided the best fit to the criteria. The original COALS system discretizes the real-valued vectors based only on the positive components. However, we evaluated codes with varying numbers of positive and negative features by comparing the semantic categories generated from the artificial semantic codes with human category data from Garrard et al.'s (2001) study. The results showed that a set of semantic vectors having 5 positive and 15 negative features could best account for the human semantic categories. It was perhaps surprising that the best binary vectors found here had more negative features than positive features (i.e., 15 negative features v.s. 5 positive features). This is at variance with the prevalent assumption that only positive components should be used to construct semantic codes. Positive features reveal what attributes are likely to be present; in contrast, the negative features provide information about what attributes are likely to be absent. So this result implies that knowledge of the absence of features (e.g., knowing that washing machines cannot walk) is as important as knowledge of positive features. Given both of these types of information, it is possible to separate categories on the basis of their distance in the semantic space, as the optimisation results shown in Table 2. Further significance tests reveal that there was a clear trend for semantic vectors containing both positive and negative features to show a more human-like semantic structure, suggesting that this may be a generally applicable principle. Overall the best performing set of semantic vectors matched the human categorisation data quite well. Further work may be conducted to investigate the performance of the present semantic codes on other types of semantic tasks (e.g., synonym judgement) for a more complete evaluation.

In addition, there are some inherent limitations of using these semantic vectors. The first is that the semantic features are not interpretable because they only encode the semantic regularities among word meanings. What exactly the feature represents is difficult to interpret. But this is only a problem in applications where a direct interpretation of features is required. Another limitation is that it can generate good semantic vectors for most of the uninflected words but it could be difficult to properly account for the deeper meaning of words like morphological regularities (e.g., bake/baker) (Harm, 2002), which would require some additional coding.

To summarise, a novel semantic representation scheme was produced based on modifications to the COALS system. This was evaluated against human categorisation data, and the resultant coding scheme was able to reproduce the human data quite closely. The key finding was that the negative features, which indicate what attributes definitely do not belong to a lexical item, were at least as important as the positive features. The semantic system developed here can be applied to generate semantic codes for a larger word list used to train more sophisticated computational models.

Acknowledgments

This research was supported by grants under the Cognitive Foresight Initiative (jointly funded by EPSRC, MRC and BBSRC - EP/F03430X/1) and the Neuroscience Research Institute at the University of Manchester.

References

Dilkina, K., McClelland, J. L. &Plaut, D. C. (2008).A single-system account of semantic and lexical deficits in five semantic dementia patients.*Cognitive Neuropsychology*, 25(2), 136-164.

Furber, S. B., Bainbridge, W. J., Cumpstey, J. M. & Temple, S. (2004). A sparse distributed memory based upon n-of-m codes. *Neural Networks*, 17.

Garrard, P., Lambon Ralph, M. A., Hodges, J. R. & Patterson, K. (2001).Prototypicality, distinctiveness, and intercorrelation: Analyses of the semantic attributes of living and nonliving concepts. *Cognitive Neuropsychology*, 18(2), 125-174.

Harm, M. W. (2002). Building large scale distributed semantic feature sets with wordnet. Pittsburgh, Carnegie Mellon University.

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

Hubert, L. &Arabie, P. (1985).Comparing partitions.*Journal of Classification*, 2(1), 193-218.

Landauer, T. K., Foltz, P. W. &Laham, D. (1998).An introduction to latent semantic analysis.*Discourse Processes*, 25(2-3), 259-284.

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

McRae, K., Cree, G. S., Seidenberg, M. S. &McNorgan, C. (2005).Semantic feature production norms for a large set of living and nonliving things.*Behavior Research Methods*, 37(4), 547-559.

Miller, G. A. (1990). Introduction to wordnet: An on-line lexical database*. *International Journal of Lexicography*, 3(4), 235.

Plaut, D. C. (1997). Structure and function in the lexical system: Insights from distributed models of word reading and lexical decision. *Language and Cognitive Processes*, 12(5-6), 765-805.

Rogers, T. T., Lambon Ralph, M. A., Garrard, P., Bozeat, S., McClelland, J. L., Hodges, J. R. & Patterson, K. (2004). Structure and deterioration of semantic memory: A neuropsychological and computational investigation. *Psychological Review*, 111(1), 205-235.

Rohde, D. L. T., Gonnerman, L. &Plaut, D. C. (2006). An improved model of semantic similarity based on lexical co-occurrence. *COMMUNICATIONS OF THE ACM*, 8, 627-633.