Measuring Semantic Ambiguity

CoMPLEX MRes Case Presentation 3

Elizabeth Gallagher
Supervisors: Prof. Gabriella Vigliocco, Dr Stefan Frank and Dr Sebastian Riedel

June 11, 2013

Abstract

The reading time for a word varies with how ambiguous it is. Furthermore, the time it takes to read an ambiguous word is dependent on whether it is in context or not. Using results from latent semantic analysis and the lexical database WordNet, four ways of measuring the degree of semantic ambiguity are developed. In-context and out-of-context reading times for 190 words are used to test the relationship between ambiguity and reading time. Our results show that ambiguous words take less time to read both in-context and out-of-context.

Contents

1 Introduction

The time taken to interpret a word can vary depending on how ambiguous the word is. Visual lexical decisions are often found to be quicker for words which are ambiguous, even when controlling for word familiarity, this finding is called the *ambiguity advantage* (Rodd, Gaskell & Marslen-Wilson, 2002). Interpreting which meaning of an ambiguous word is being used can rely on the context the word is being used in (Leacock, Towell & Voorhees, 1996).

To look at the relationship between reading time and word ambiguity, for in-context and out-of-context words, it is first necessary to find a way to measure the degree of ambiguity that a word has. One measure could be the number of synonyms a word has, this can be found using WordNet[1](Miller & Fellbaum, 1991). Other measures can be found by looking at the neighbourhood of related words which often co-occur in texts with it.

In this report we will use results found from latent semantic analysis (LSA) to find three alternative measures of ambiguity along with the number of senses found using WordNet. The first will be to find the average similarity score between all pairs of close neighbours to a word. The other two measures will focus on finding a way to quantify how clustered a network of close neighbours of a word is; these measures will be the number of clusters in a neighbourhood network and its clustering coefficient.

Two datasets from Frank *et al* (2013) and Keuleers *et al* (2012) will provide us with a comparison of reading times for words in-context and out-of-context respectively. Using these we will investigate whether any of our ambiguity measures can predict the reading times of words. We will also consider this relationship when only looking high and low frequency words separately.

2 Semantic ambiguity and details of the data used

2.1 Homonyms and polysemy

Homonyms and polysemous words are two types of ambiguous words. Homonyms are different words which mean different things, but share the same orthographic and phonological form; for example "bark" can mean the outer layer of a tree, or the sound a dog makes. Polysemous words are considered as the same word with different, but related, senses; for example "wood" can refer to a piece of a tree or a collection of many trees. It may be expected that for the majority of cases in-context ambiguous words take less time to read than out-of-context ambiguous words due to the topic being introduced. For example in the sentence "the table is made of wood" it is obvious that the table is not made of a collection of trees. Although, it is not always the case that context reveals the meaning of an ambiguous word, for example in the sentence "I buried £100 in the bank" it is not very clear which meaning of "bank" is being used. The possibilities of using our ambiguity measures to distinguish between homonyms and polysemous words will be discussed as an extension.

2.2 Ambiguity advantage

During lexical decision tasks, ambiguous words are processed more quickly than unambiguous words with the same familiarity. This was first reported by Rubenstein, Garfield &

[1] see http://wordnet.princeton.edu/

Millikan (1970) and since then there have been several other observations of it. The popular theory for this effect is that ambiguous words have more lexical entries for comparison against, so they are recognised sooner than unambiguous words. Rodd, Gaskell & Marslen-Wilson (2002) investigated the ambiguity advantage in polysemous words and homonyms separately. Their findings suggested that the ambiguity advantage existed with polysemous words, but word recognition was delayed for homonyms.

2.3 In-context reading times

A collection by Frank *et al* (2013)[2]of both the word-by-word eye tracking data and self-paced reading times (RTs) from English sentences will be used in the analysis of in-context words.

The sentences in this dataset were obtained from independent English sentences, as opposed to sentences which only make sense within the context of surrounding sentences, taken from different narrative sources (i.e. not constructed for the purposes of experimental stimuli). These sentences were selected from three online unpublished novels. A list of high-frequency content words (used by Andrews, Vigliocco & Vinson, 2009) and the 200 most frequent English words were merged, and all sentences from the three novels which contained only words from this list of 7,754 words were selected. This list of sentences was then further restricted to only include sentences which were at least five words long and included at least two content words. 361 of these sentences, which could be interpreted out of context, were finally selected Frank *et al* (2013).

Two different paradigms for finding the reading times of participants were used and as a result the dataset contains the self-paced reading times of 117 participants and the eye tracking times of 43 participants reading these 361 sentences. Several reading times were found from the eye tracking data: the first-fixation time, the sum of all the fixations on a word before first fixation onto another word (first-pass time) or onto a word further to the right (right-bounded time) and also the sum of all the fixations from the first fixation up to the first fixation on a word further to the right (go-past time).

2.4 Out-of-context reading times

The British Lexicon Project (BLP)[3]is a database of lexical decision times for 28,730 mono- or disyllabic English words and non-words collected by Keuleers *et al* (2012). The reading times for these words were defined as the time taken for a participant to decide whether or not a stimulus is a real word or not. 2.3% of outliers were removed and then for each word the mean reading time and the mean standardised reading time of the correct word responses were calculated.

2.5 WordNet

WordNet is a network model which organises English nouns, adjectives, verbs and adverbs into synonym sets which distinguish different lexical concepts (Miller *et al*, 1990). This organisation is done by hand and is based on the semantic relations relevant for the word type being considered (e.g. hypernyms for nouns and troponyms for verbs).

WordNet 2.1 was used to find the number of noun, adjective, verb and adverb senses for each of our words. The total number of senses was used as an ambiguity measure. For

[2]data files can be found in the supplementary material of Frank *et al* (2013).
[3]data files available from http://crr.ugent.be/blp

example "breakfast" had three senses; one noun sense: "the first meal of the day (usually in the morning)", and two verb senses: "eat an early morning meal; "we breakfast at seven"" and "provide breakfast for".

2.6 LSA

Another way to look for the degree of ambiguity a word has is to consider the words which co-occur often with it in real texts.

LSA is a mathematical learning method which extracts and represents similarity in the meanings of words and passages by analysing large corpora of natural text (Dumais & Landauer, 1997). It assumes that words with a similar meaning will occur in similar pieces of text.

After inputting a large corpus of texts, first LSA creates a matrix $X_{i,j}$ of the occurence of a term i in a document j. Singular value decomposition (SVD) is then applied to this matrix to condense the information into a lower dimensional representation (Deerwester *et al*, 1990):

$$X = U\Sigma V_T \tag{1}$$

Where U and V are orthogonal matrices and Σ is a diagonal matrix. The similarity score between words p and q for k dimensions is found by the cosine of the angle between the vectors composed from the first k elements of the p^{th} and q^{th} rows of U.

The number of dimensions to use is important in maximising the accuracy with human judgements (Landauer, Foltz & Laham, 1998), see Figure 1.

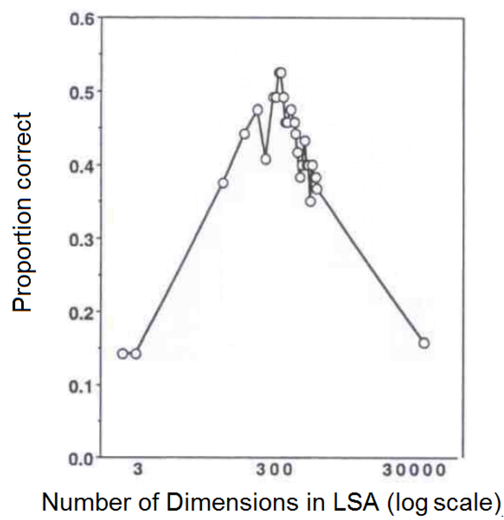

Figure 1: The effect of the number of dimensions used and performance on a synonym test. Adapted from Landauer, Foltz & Laham, 1998.

Programs on the CU Boulder LSA website[4] allow you to find the nearest neighbours in an LSA semantic space to a word along with their similarity scores. Furthermore you can submit a list of words and have returned a matrix of the similarity scores between every combination of two words. For our analysis, the topic space we selected was "General_Reading_up_to_1st_year_college (300 factors)", we used all factors and selected *terms* as input text.

[4] http://lsa.colorado.edu/

3 Selecting our data

For our analysis we selected 190 words which were present in both the in-context and out-of-context databases. These were selected in the following way.

A list of words was created from the sentences, full stops and commas were removed and all the characters were converted to lower case. This list was then compared to the BLP words and an intersection of 1094 words was found. For every word each of the part of speech (POS) tags for were found from the data in Frank *et al* (2013), for example “good” can be classed as an adjective, noun or an adverb depending on its context. We decided that we would only be interested in words which were either cardinal numbers, adjectives, nouns or verbs, and thus eliminated words which were never tagged in one of these categories. Plural nouns were deleted in cases where both their plural and singular form were present in the list. A further 95 words were deleted which had more than one time-form, e.g. “build” and “built”; “fall”,“falling” and “fallen”; “wound” and “wounded”, in these cases usually the present form was kept. And finally, we eliminated some words which did not have data for both the eye tracking and self-paced reading times.

For each of these 190 words we also have two other parameters; the log-transformed word frequency, found in the British National Corpus (BNC)[5], and the length of the word (number of letters).

3.1 Reading Times

The first pass reading times for the eye tracking data were used. For the self-paced reading times there can be a reaction delay and it becomes more accurate to take the reading time for the next word in the sentence rather than use the time for the current words’. Furthermore, due to additional cognitive processes, the reading time from the last word in a sentence can be inaccurate. Hence two sets of self-paced reading times were used; the original reading times and delay-corrected reading times which do not include last words. Since 23 of the words in our list are only used at the end of sentences, the delay-corrected self-paced reading times are missing data for some of the words.

Using these reading times, for each participant, the average reading time for each word on the list was found from all the sentences. Hence we created a 43 x 190 cell matrix for the eye tracking reading times, and two 117 x 190 cell matrices for the self paced reading times. Outliers were then removed from the reading times, the method used to do this is explained in Appendix A.

The spread of the reading times with outliers removed can be seen and compared in Figure 2.

3.1.1 Standardising reading times

To standardise the data to eliminate differences in the mean and variance of the reading times between subjects, the RTs were first log transformed, then z-transformed for each participant, and then averaged over all participants. To standardise the BLP out-of-context RTs we log transformed and then z-transformed the data.

The full list of words and data can be found in Appendix B.

[5]http://www.natcorp.ox.ac.uk/

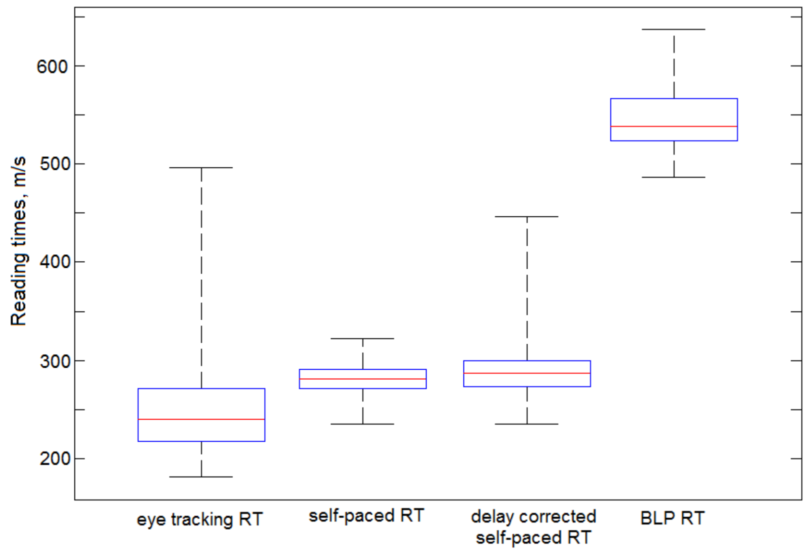

Figure 2: Box plots of the spread of the different reading time data. The extremes of the data are shown in black, the median in red and the 25^{th} and 75^{th} percentiles in blue.

4 Preliminary analysis

4.1 Word frequency and length with reading time

The relationship between word frequency and word length with reading time are shown in Figures 3a and b.

The eye tracking RT shows a significant negative correlation (the standardised RT has a correlation coefficient of -0.243) with word frequency, thus a more frequent word takes less time to read using this measurement. There is also a significant positive correlation between both the eye tracking and the BLP RTs and word length, hence longer words take longer to read. These results both make sense intuitively as less familiar words or longer words should take more processing. This result is reassuring but also means that we should account for word frequency and length in our analysis.

4.2 Number of senses with reading time

Using the in-context eye tracking and self-paced RTs and the out-of-context BLP RTs, we found the relationship between RT and the number of WordNet senses. Figure 17a in Appendix C shows the difference in reading times for in-context and out-of-context words, Figure 17b shows this for the standardised RTs.

There is a clear increase in the time taken to read a word for out-of-context words, however this is due to the different paradigm used to find these reading times. RTs are higher for the self-paced measurements than eye tracking measurements, again because of the difference in paradigm.

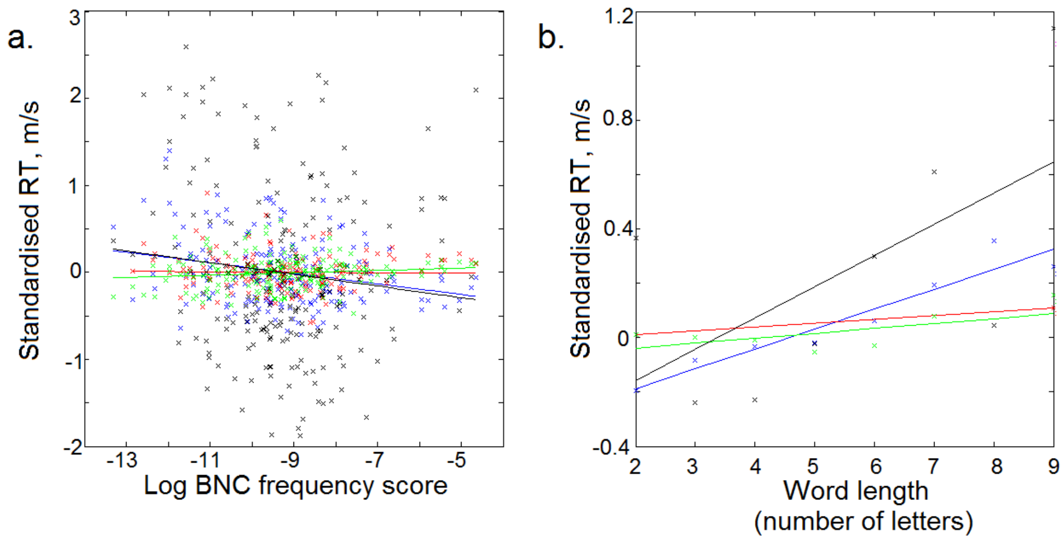

Figure 3: Standardised reading times for different word frequencies (log-transformed), as measured by the BNC frequency score (a) and word length (b). For in-context words with RT measured by eye tracking (blue) and delay corrected self-paced (red), non-corrected self-paced (green) and also out-of-context words from the BLP RT dataset (black). Data points (crosses) and line of best fit (smooth).

If WordNet was a reasonable measure for ambiguity it might be expected that we saw a trend in the RTs with the number of senses. However, there is no relation between the number of senses a word has and its reading time for any of the three reading time measurements (all the correlation coefficients are very low). Finding the Pearson linear partial correlation coefficients whilst controlling for word length and frequency also yielded insignificant results. Hence another measure for ambiguity is needed.

5 Analysis

Using the LSA website we found the list of neighbours and similarity scores for each word. Either the top 20 results or all the words with ≥ 0.5 LSA similarity score were saved. Certain words were discarded, such as names, numbers, abbreviations, and pluralisations and different tenses of the original word. For example, for the word "city" the six nearest neighbours were "cities", "streets", "suburbs", "boulevards", "urbs", "cbd", of these "cities", "urbs" and "cbd", see Figure 4. These deletions were done by hand, and not double checked, thus the editing may not be very consistent.

High frequency words tended to have a larger number of neighbours with a high similarity score (e.g. the number of neighbours of a word which have a similarity score of ≥ 0.75 has a significant positive correlation with word frequency). This makes sense intuitively, as more common words should have closer neighbours in the corpus.

In this analysis we have chosen to look at only the top 20 closest neighbours of a word. The other option of only looking at neighbours which are over a similarity score threshold, i.e. truely "close" neighbours, is discussed later as an extension.

a.

1	**city**
0.71	cities
0.67	streets
0.65	suburbs
0.64	boulevards
0.62	urbs
0.61	cbd
0.6	skyscrapers
0.6	metropolitan

b.

	city	streets	suburbs	boulevards	skyscrapers	metropolitan
city	1	0.67	0.65	0.64	0.6	0.6
streets	0.67	1	0.52	0.63	0.52	0.37
suburbs	0.65	0.52	1	0.56	0.52	0.8
boulevards	0.64	0.63	0.56	1	0.5	0.43
skyscrapers	0.6	0.52	0.52	0.5	1	0.39
metropolitan	0.6	0.37	0.8	0.43	0.39	1

Figure 4: a) LSA similarity scores for the first five nearest neighbours of "city", neighbours in red to be discarded. b) Similarity scores to be included in the analysis (highlighted squares) for each of the neighbour-to-neighbour pairs.

5.1 Average neighbour-to-neighbour similarity score

The similarity scores for all the top 20 neighbour-to-neighbour pairs were found for each word. We are interested in looking at the differences in the average top 20 neighbour-to-neighbour similarity score for each word, Figure 5 shows the distribution of these average scores. The mean similarity score between all the neighbours of all the words is 0.22. The fact that the mean of the average per word (0.6) is much greater than 0.22 shows that the average is reflecting something about the relationships the words have with their neighbours.

If a word has a high average score then this could be seen as a sign that it is an unambiguous word, the theory being that its neighbours are all tightly related and hence do not reflect multiple meanings. There are significant correlations between this average score and the number of WordNet senses and also the BLP log frequency score, Figure 6 shows these relations.

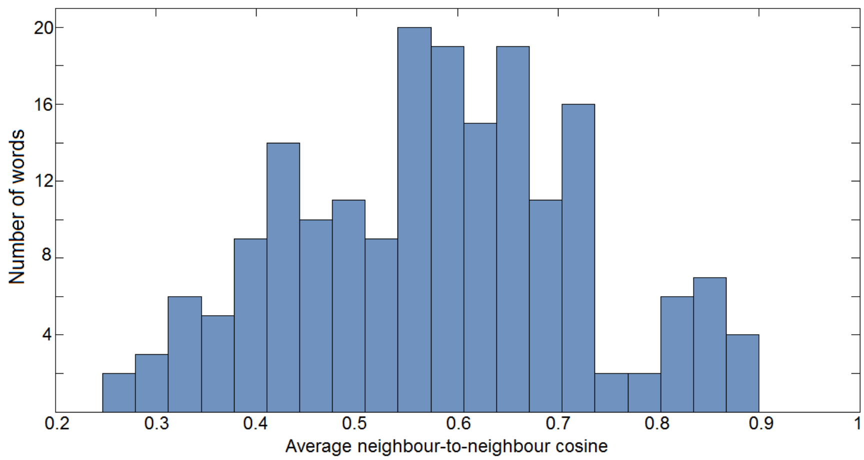

Figure 5: Average similarity scores between neighbours-to-neighbours of a word.

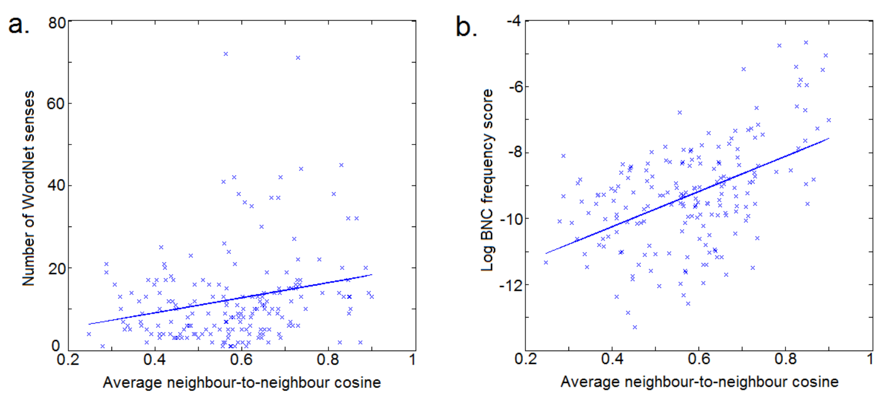

Figure 6: a) Average neighbour-to-neighbour cosine for top 20 neighbours and number of WordNet senses for each word. b) Average neighbour-to-neighbour cosine for top 20 neighbours and BLP log word frequency for each word. Line of best fit shown.

5.2 Network clusters

Using the mean neighbour-to-neighbour similarity score may not tell us anything about the degree of ambiguity a word has. This could be because the neighbours of an ambiguous word should in theory form clusters of closely related words, reflecting the different meanings of the word. Hence taking a mean of all the similarity scores could give the same mean as an unambiguous word. Figure 7 illustrates this point with the theoretical relationship spaces of an ambiguous and an unambiguous word, where the mean neighbour-to-neighbour score in both cases is 0.65.

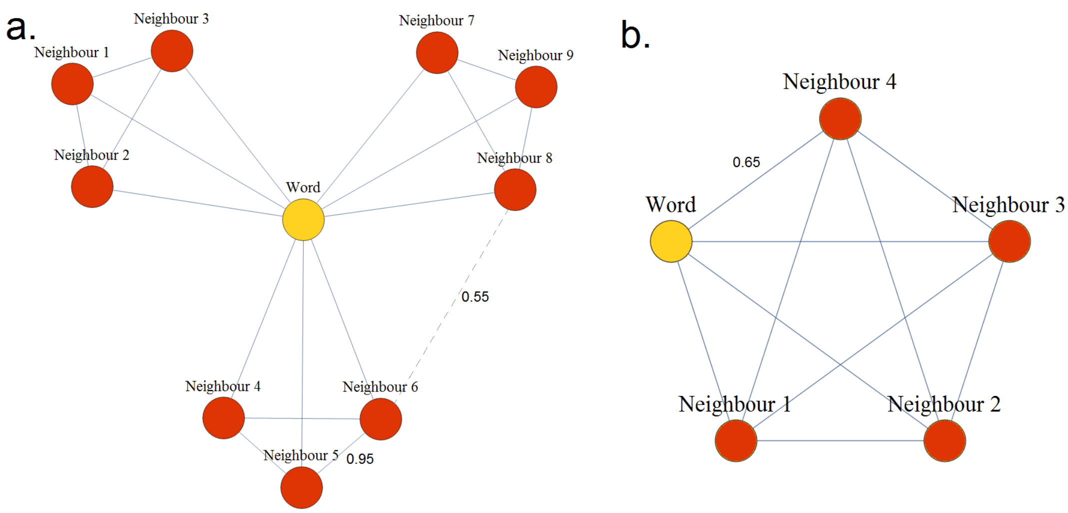

Figure 7: a) An ambiguous word, with three meanings illustrated by three clusters of neighbours, intercluster similarity scores are 0.95, and intracluster scores are 0.55 (not all lines and scores are shown in the diagram). b) An unambiguous word, all similarity scores are 0.65.

Clusters of closely related neighbours could reflect the different meanings of an ambiguous word. This can be seen in Figure 8, where the top 20 neighbours of "breakfast" are shown, and lines are drawn between all the neighbours which have a similarity score of ≥ 0.55 (i.e. closely related) to one another. The graph could be seen to reflect some of the different meanings for breakfast; a time of day and an eating activity. There are clusters of neighbours which relate to times of the day (e.g. "morning", "evening" and "afternoon"), kitchen related meanings (e.g. "kitchen", "cup" and "dishes") and, with less connections to these groups (and to each other), a group of breakfast foods (e.g. "pancakes", "jam" and "toast").

Hence finding a way to find the number of clusters of neighbours from their similarity scores with one another could yield a more accurate metric for predicting ambiguity.

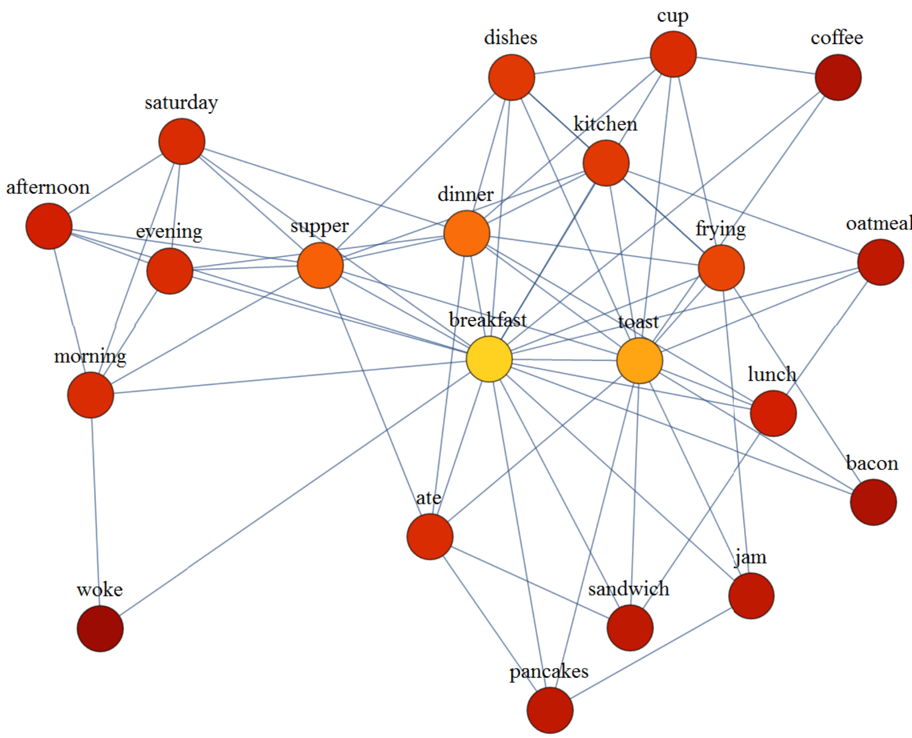

Figure 8: The relationships between the top 20 neighbours of “breakfast”. Lines are drawn between the neighbours which have a similarity score ≥ 0.55. Node colour reflects the number of connections this node has to others.

5.3 The number of clusters

Using the built-in Matlab “linkage” and “cluster” functions we can make a prediction of the number of clusters for each of the words from an input of the 20 x 20 matrix of neighbour-to-neighbour similarity scores.

This works by first returning the cluster indices and linkage distances for a tree of hierarchical clusters. In this analysis the weighted average distance was used to compute the distance between clusters, and the distance metric was set to “cosine”. Next clusters are found, these are defined as when a node and its subnodes had an inconsistency coefficient less than a cluster threshold. The inconsistency coefficient for a node at a certain linkage distance is calculated by:

$$\frac{\text{Linkage distance at this node} - \text{Mean of distances at this level and 2 levels below}}{\text{Standard deviation of distances at this level and 2 levels below}} \quad (2)$$

Using this method, we would predict that if a word has a lot of clusters then it has a high degree of ambiguity.

Figure 9 shows 4 nodes of the “breakfast” hierarchy tree, with the linkage distances. In this example the inconsistency coefficient for node 13 would be calculated using nodes 10 and 1, hence it would be 0.9087 and thus this whole section would be defined as a single cluster (using a cluster threshold of 1). Figure 10 shows the full dendogram for "breakfast" generated from this process, the 5 different clusters found are illustrated in different colours.

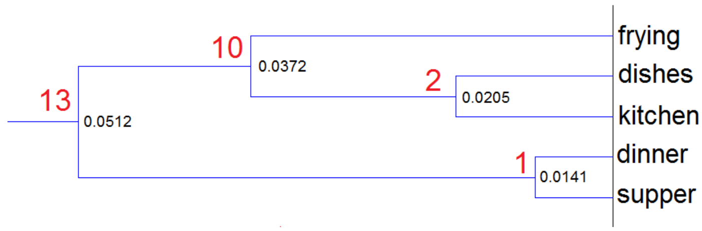

Figure 9: A section of the "breakfast" hierarchy. Numbers show the linkage distance (black) and the node number (red).

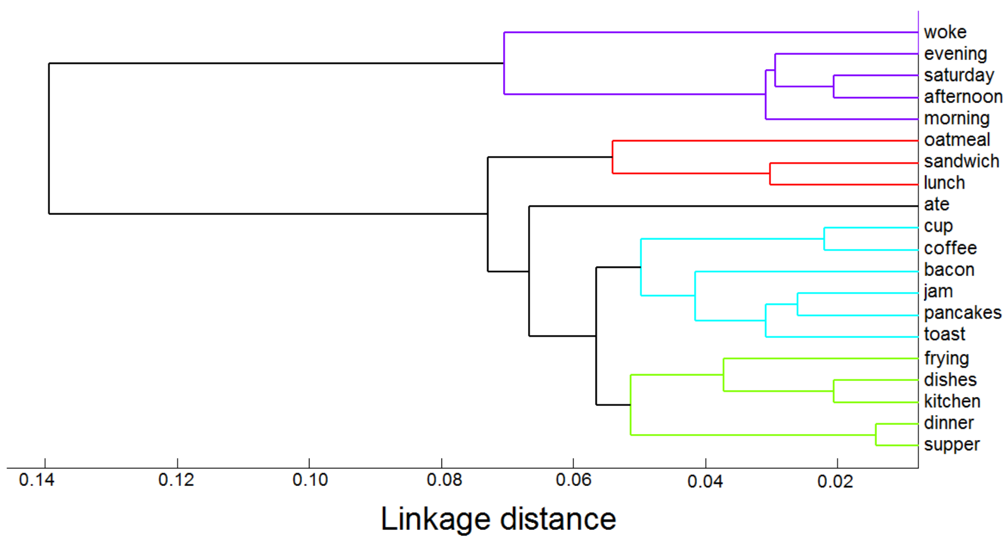

Figure 10: Dendogram showing the clusters of the top 20 neighbours of "breakfast". The five clusters found with a cluster threshold of 1 are shown in different colours.

5.3.1 Finding the clustering threshold

The clustering threshold we will use was found by considering the number of clusters generated for all the words for a range of clustering thresholds. Ideally we would want a threshold which gave a range of different numbers of clusters (i.e. it would be unuseful to pick one which just gave 1 cluster for each word), but at the same time we want one which does not give too high a number of clusters (which would be unrealistic). Figures in Appendix D show the distribution of cluster numbers for different thresholds, Figure 18 shows thresholds from 0.7 to 1.2, and for closer inspection Figure 19 and 20 show those between 0.9 and 1.1.

Anything higher than 1.2 gave a total number of clusters as 1 for all words, and below this range gave very large numbers of clusters. Using these graphs we have chosen to use a clustering threshold of 1 and 1.025.

5.4 Clustering coefficients

The "LocalClusteringCoefficients" function in Mathematica gives the local clustering coefficients of all vertices in a graph. Hence, another possibility to measure ambiguity would be to use the mean local clustering coefficient of a words neighbours. This is measured as the fraction of pairs of neighbours of a vertex that are connected over all of the pairs of neighbours of the vertex. This requires graphs to be produced of all the neighbour connections which have a similarity score over a certain threshold, hence the actual similarity scores are not taken into account.

The red node in Figure 11a has a local clustering coefficient of 0.5 since of its 6 neighbour-to-neighbour pairs, only 3 of these are also connected. The graph in Figure 11(a) has a mean clustering coefficient of 0.583, Figure 11(b) of 1 and Figure 11(c) of 0. Hence we will hypothesise that a high mean local clustering coefficient means the word has a low degree of ambiguity.

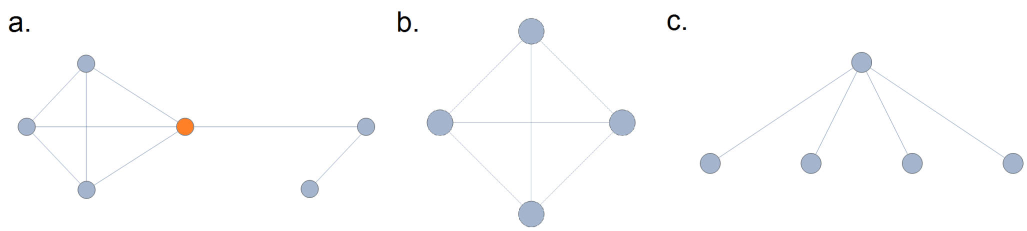

Figure 11: Three theoretical clusters. The red node has a local clustering coefficient of 0.5.

Using only connections between neighbours which have a similarity score of ≥ 0.55 to one another, we found the clustering coefficient for each word. "Breakfast" has a clustering coefficient of 0.63 and "chair" of 0.78, hence we would predict "breakfast" is more ambiguous than "chair".

6 Results

We looked for the correlations between our several different ambiguity measures and the standardised reading time data, as well as several other parameters. Seeing as word frequency and length are unrelated to ambiguity but effect the results significantly, we also found the Pearson linear partial correlation coefficients controlling for word length and frequency. Figure 12 shows these partial correlation coefficients for each of our ambiguity measures and the standardised reading times. The full list of these correlations can be found in Figures 21 and 22 in Appendix E.

	Standardised in-context eye tracking RT	Standardised in-context self paced (delay corrected) RT	Standardised in-context self paced RT	Standardised out-of-context BLP RT
Number of WordNet senses	0.006	-0.03	0.068	0
Average neighbour-to-neighbour similarity score for top 20 neighbours	0.079	0.123	0.059	0.193
Number of clusters (found using a clustering threshold of 1)	-0.144	-0.048	0.004	-0.116
Number of clusters (found using a clustering threshold of 1.025)	-0.166	-0.049	0.028	-0.13
Clustering coefficient (for neighbour-to-neighbour scores ≥ 0.55)	0.129	0.115	0.078	0.142

Figure 12: Pearson linear partial correlation coefficients, controlling for word length and frequency. Highlighted squares show significant correlations (with a p-value ≤ 0.05).

There is no significant correlation between the average neighbour-to-neighbour similarity score and any of the reading time measurements. However the partial correlation coefficients controlling for word length and frequency show that the mean neighbour-to-neighbour similarity score was significantly positively correlated with the out-of-context RT. Hence it takes longer to read out-of-context words when the words LSA measured neighbours are closely related to one another, i.e. when the word is fairly unambiguous.

We found no *significant* correlations between the clustering coefficient and the reading time measurements, however there is a weak positive trend in the eye tracking and the BLP RTs. From our hypothesis this weak result corresponds to words with a low degree of ambiguity taking longer to read for both in and out of context words.

6.1 Number of clusters

We found a significant negative correlation between the number of clusters and the standardised eye tracking RT. There was also a slight negative trend in the standardised BLP RTs. Hence, using the number of clusters as a measure of ambiguity, suggests it takes longer to read words with a lower degree of ambiguity. However, a hypothesis that ambiguous words in-context should take less time to read than ambiguous words out-of-context cannot be supported with this measurement. Figure 13 shows these relationships, the differences in means between the in-context and out-of-context reading times are insignificant (using a T-test and also a Wilcoxon rank sum test).

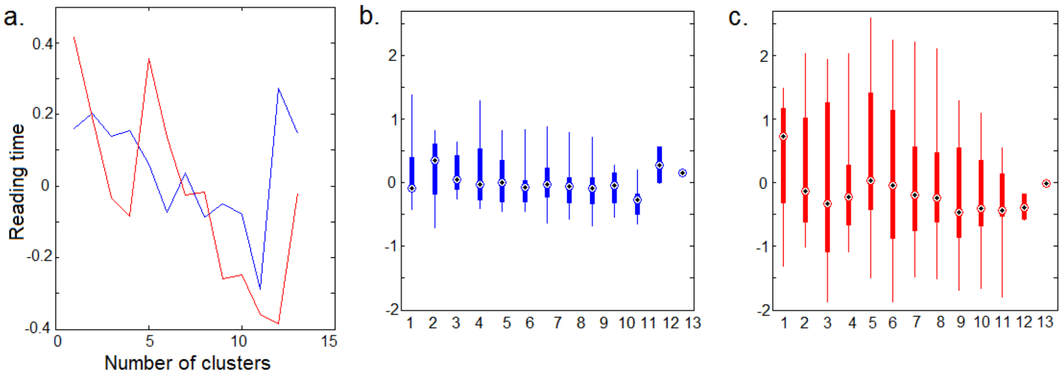

Figure 13: Standardised reading times against number of clusters. a) The average reading times for each of the number of clusters (using a threshold of 1.025). b) and c) show box and whiskers plots of all the data. The in-context eye tracking RT is shown in blue and the out-of-context BLP RT is shown in red.

6.2 Comparing high and low frequency words

As there were significant correlations with word frequency, we decided to look at words with different frequencies separately. We split the data into terciles which corresponded to low frequency words (63 words, with frequencies $<$-9.9), medium frequency words (64 words, with frequencies between -9.9 and -8.8) and high frequency words (63 words, with frequencies $\geq$-8.8). The correlations for the ambiguity measures and reading times for each of these sets of data were found (see Figure 14). These results generally further support the patterns found using all the word frequencies, and reinforce the slight positive correlation we found between the clustering coefficient and RTs for high frequency words. Figure 15 shows in-context (delay corrected self-paced) and out-of-context reading times against the clustering coefficient for high frequency words.

Interestingly, there are some sign changes in certain correlations between low and high frequency words. Although not always significant, some of theses correlation coefficient changes are reasonable high and worth noting. Firstly, for the average neighbour-to-neighbour score with self-paced delay-corrected RTs, we found a negative correlation for low frequency words and a positive correlation for high frequency words. In other words, highly ambiguous words take longer to read than unambiguous words in-context if they are of a low frequency, and take *less* time to read in-context if they have a high frequency. The same conclusion was found when using the number of clusters as an ambiguity measure and using the self-paced (but not delay corrected) RTs.

	Low frequency words				Medium frequency words				High frequency words			
	Standardised in-context eye tracking RT	Standardised in-context self-paced (delay corrected) RT	Standardised in-context self-paced RT	Standardised out-of-context BLP RT	Standardised in-context eye tracking RT	Standardised in-context self-paced (delay corrected) RT	Standardised in-context self-paced RT	Standardised out-of-context BLP RT	Standardised in-context eye tracking RT	Standardised in-context self-paced (delay corrected) RT	Standardised in-context self-paced RT	Standardised out-of-context BLP RT
Number of WordNet senses	-0.18	-0.21	-0.045	-0.165	-0.046	0.158	0.145	0.08	0.1	-0.03	0.118	0.025
Average neighbour-to-neighbour similarity score for top 20 neighbours	-0.02	-0.15	-0.095	0.282	0.048	-0.03	0.226	-0.049	0.051	0.505	0.206	0.23
Number of clusters (found using a clustering threshold of 1)	-0.144	-0.19	0.138	-0.318	-0.149	-0.04	0.016	-0.02	-0.253	0.101	-0.216	-0.099
Number of clusters (found using a clustering threshold of 1.025)	-0.206	-0.16	0.166	-0.338	-0.124	-0.08	0.007	-0.047	-0.307	0.094	-0.161	-0.097
Clustering coefficient (for neighbour-to-neighbour scores ≥ 0.55)	0.113	-0.05	-0.082	0.196	0.059	0.03	0.255	-0.079	0.123	0.471	0.187	0.266

Figure 14: Correlation coefficients for ambiguity measures and reading times for high, medium and low word frequencies. Highlighted squares indicate significant (p-value ≥ 0.05) correlations.

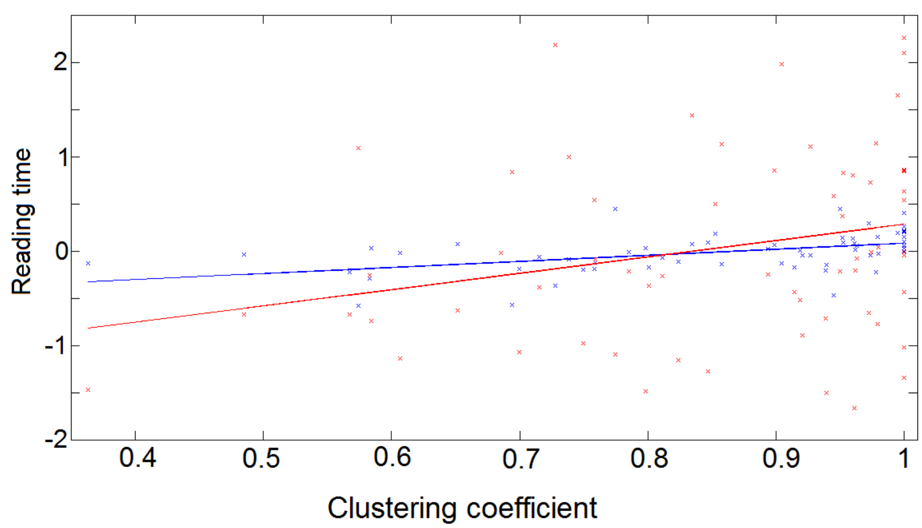

Figure 15: Standardised reading times against the clustering coefficient for high frequency words. The in-context self-paced (delay corrected) RT is shown in blue and the out-of-context BLP RT in red, lines of best fit are plotted too.

7 Discussion

7.1 Extensions

In this analysis we have used *all* of the top 20 neighbours when finding the average neighbour-to-neighbour similarity score and in finding the number of clusters. However, in the top 20 neighbours list generated by the LSA website, sometimes the average similarity score is very low and this is not always a reflection that the word is an uncommon one. For example "aid" has only 5 neighbours which are ≥ 0.5 to it, but has a word frequency score which is very close to the mean frequency for all words. Hence it would be of interest to only include in the analysis the words which had an average top-20-cosine-to-word-frequency ratio over a certain threshold, or similarly, only analyse the neighbours which have a score over a certain threshold.

For some words the linkage distances can all be quite low, for example the word "chair", see Figure 23 in Appendix E. A further extension would be to consider using a cut-off point for the linkage distance where any nodes below this distance are all classed as one cluster.

Other interesting extensions would be to use a different corpus on the LSA website or to investigate words which were greater than two syllables. Some smaller additions would be to look at the effect of word position in the in-context data and the word type (e.g. just look at the nouns) as accounting for these may change reading times significantly.

Homonym vs Polysemous

The methods used here only measure for a degree of ambiguity and do not distinguish between polysmous words and homonyms. Hence a further extension would be to find a way to classify these using our ambiguity measures.

It may be predicted that for an ambiguous word the neighbour-to-neighbour pairs which reflect a word from one meaning and a word from another would have a lower similarity score for homonyms than for polysemous words. For example, if we took the homonym "bark" and found the distance between its two neighbours "dog" and "tree" we would have a lower similarity score than for the neighbours "tree" and "plank" from the polysemous word "wood".

Using this argument, we might expect a homonym to have a smaller average neighbour-to-neighbour similarity score than a polysemous word. Furthermore, it might be expected that the neighbours of a polysemous word would form clusters with more interrelations than the neighbours of a homonym. Hence the clustering coefficient would be higher for polysemous words than for homonyms. Finding the thresholds for when a word has a score high enough to be catagorised as a polysemous word would be an interesting extension. It could also mean we could test the result of Rodd, Gaskell & Marlsen-Wilson that the ambiguity advantage only exists for polysemous words.

7.2 Conclusions

In this report we have discussed the use and significance of four different measures of word ambiguity, one using the number of synonyms measured by WordNet and three using data from LSA. Using these we have investigated the relationship between ambiguity and reading time for both in-context and out-of-context words. We have also discussed the effect word frequency has.

Using all three of our LSA based measures of word ambiguity, we found that words with a high degree of ambiguity took less time to read than those with a low degree. This finding supports the idea of an “ambiguity advantage”. We found no pattern with the number of WordNet senses and reading time, suggesting that this measure is independent of the degree of ambiguity a word has. We also were unable to find any significant results as to whether or not in-context or out-of-context ambiguous words had any distinct patterns in the time taken to read. From our results we can also (although with some uncertainty) infer that highly ambiguous words take longer to read than unambiguous words in-context if they have a low frequency, and conversely take less time to read if they have a high frequency.

The analysis used here gives possible ways of finding a metric for ambiguity using the LSA similarity scores between neighbours of a word. As well as developing these ideas it would be of interest to use these approaches with a list of neighbours generated from a different LSA source.

Appendix

A. Removing outliers

Outliers were found by looking at both per word and per participant ranges. First, for each word separately the RT scores which were greater than the upper quartile plus four times the interquartile distance for each word were found. Next, from this list, if this RT score was also greater than the upper quartile plus four times the interquartile distance for this participant, then it was classed as an outlier. In this way, for the eye tracking data, 0.21% RT scores were classed as outliers, for the delay-corrected self-paced RT data 0.37% were classed as outliers and for the original self-paced RTs 0.54% were classed as outliers. The outliers to be removed from the eye tracking RT data can be seen in Figure 16.

For example, for the word "big", the delay-corrected self-paced RTs are greater than the upper quartile plus four times the interquartile distance for participants 27, 95 and 112. Of these, only participants 95 and 112 have RTs greater than the upper quartile plus four times the interquartile distance for this word, these will be classed as outliers.

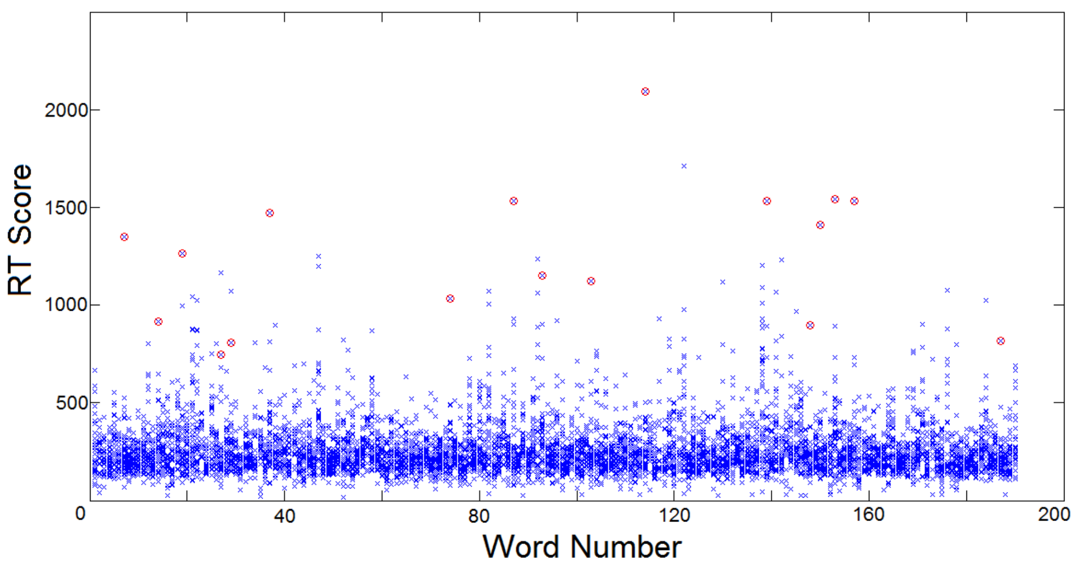

Figure 16: Eye tracking reading time data for each word. Outliers to be removed highlighted in red.

B. List of the data used

	WordNet senses	Word frequency	Word length	Eye RT	Standardised eye RT	Delay-corrected self RT	Standardised delay-corrected	Self RT	Standardised self RT	BLP RT	Standardised BLP RT
aid	6	-9.28585622	3	266.2857	0.072732031	282.484375	0.099443564	278.785	0.116095512	537.359	-0.237645931
am	16	-8.23683219	2	232.32	-0.222915252	260.843137	-0.370787536	274.954	-0.131746406	621.5	2.181164983
anger	5	-10.1376158	5	209.0588	-0.371910438	278.753247	-0.077468965	282.662	-0.014928084	532.3684	-0.392791196
are	14	-5.39996777	3	205.6423	-0.398453059	291.260914	0.1452662	282.736	0.01656677	573.4848	0.844232172
axe	3	-11.7470537	3	281.0588	0.321821957	286.696078	0.144718622	288.119	0.118818422	519.0513	-0.814019787
band	14	-9.63223435	4	222.1622	-0.189479699	259.013699	-0.225281462	267.773	-0.116977541	523.2	-0.681645523
battle	4	-9.58938233	6	248.6341	-0.004898158	308.992963	0.059503663	290.135	0.105964353	544.0278	-0.032561747
be	14	-5.0419861	2	209.5043	-0.326224359	296.222415	0.215039949	289.831	0.11866145	530.8378	-0.440665174
been	13	-5.9482843	4	209.7778	-0.318760402	285.282001	0.025542634	277.062	-0.044297069	573.7027	0.850547433
began	10	-8.32163678	5	219.0476	-0.207694068	272.30303	-0.138273643	275.458	-0.063822725	583.4857	1.131697785
belt	9	-10.7916488	4	241	-0.032452443	283.62963	-0.005467784	279.01	-0.1399635	511.1053	-1.070535973
big	17	-8.47132116	3	290.3	0.210254863	292.482558	0.03194529	283.261	0.024198266	521.4211	-0.738277867
bit	14	-8.77189169	3	208.7287	-0.342560768	296.609375	0.00204131	293.372	0.181347386	492.5946	-1.683912683
black	23	-8.32782959	5	289.3158	0.374270979	288.380952	-0.289664697	278.668	-0.152258421	536.5789	-0.261800002
bored	4	-11.1666814	5	271.4031	0.258531188	348.683333	0.390095222	302.85	0.106046545	619.6111	2.130552706
bottom	12	-9.6679585	6	233.7895	-0.326550508	322.906593	0.662352119	313.571	0.520026994	537.875	-0.221686106
boxes	13	-10.6077303	5	217.5135	-0.394966121	264.589041	-0.142688394	269.342	-0.013987534	524	-0.656240462
breakfast	3	-10.0800266	9	297.2016	0.469192705	403	0.417993081	322.611	0.450007134	536	-0.279750216
breath	5	-9.89309573	6	266	0.079268496	299.354839	-0.049431044	275.187	-0.120495884	597	1.512422551
broken	72	-9.55972716	6	254.619	0.112344092	288.306854	0.076333185	291.809	0.116727619	521.2308	-0.744346926
brother	5	-9.49343194	7	393.9	0.798027846			264.652	0.012807872	602.15	1.655244903
build	12	-9.55677845	5	376.1026	0.831084156	273.451456	-0.196437681	274.784	-0.197614353	507.4359	-1.190340737
burn	20	-11.0092139	4	315.36	0.531376796	288.283333	-0.083944641	279.197	-0.091796057	552.2632	0.21725744
buy	6	-9.16394509	3	181.697	-0.670698761	267.328571	-0.238394474	271.971	-0.18303734	520.525	-0.766876692
campaign	7	-9.2275494	8	333.0909	0.688691433	266.337838	-0.108682928	257.48	-0.29281637	561.3684	0.48916447
carriage	5	-10.8402259	8	260.4211	0.129329333	305.355556	0.300091438	287.183	0.024236246	578.7105	0.995059235
ceased	2	-11.058766	6	264.8421	-0.090677781			260.615	-0.188836689	589.2368	1.294785413
chair	6	-9.57334133	5	256.3333	0.089784492	286.994186	-0.229115549	279.583	-0.053932706	520.6	-0.76448107
chance	9	-8.96507365	6	279.4872	0.253736665	265.04878	-0.215045662	305.524	0.17961972	531.3	-0.426195015
choked	15	-12.3153637	6	226.8889	-0.189154997	309	0.213124638	250.59	-0.277184483	551.025	0.179936974
city	3	-8.42189814	4	185.037	-0.632429463	266.081967	-0.037331842	258.016	-0.120145179	523.3243	-0.677694891
claims	11	-9.26122414	6	245.0256	-0.006612599	256.753623	-0.10398606	267.897	0.051818115	559.4872	0.433348934
closed	26	-9.24736764	6	230.6179	-0.036513744	278.724359	-0.040313713	287.345	0.061993817	542.2432	-0.087193728
cold	16	-9.09942618	4	247	-0.093578409	317.45873	0.479843834	274.96	-0.062749039	503.9459	-1.305094189
come	22	-7.53870879	4	268.8889	0.069399323	268.434066	-0.222439926	281.022	0.066690466	583.9189	1.144038249
control	19	-8.10666497	7	218.4444	-0.33695243	296.12	-0.13486981	289.962	0.061431125	498.8421	-1.474353084
counter	13	-10.6220162	7	314.2439	0.58313605			300.844	0.228466839	547.6944	0.079129903
cut	71	-8.71467437	3	271.25	0.168952147	267.760417	-0.169553135	275.167	-0.030551506	531.0513	-0.433980735
dad	1	-10.2201547	3	245.2973	0.025742884	314.485417	-0.041901581	282.139	-0.054964768	495.6111	-1.58240019
dared	3	-11.5756409	5	244.3243	-0.043091677	266.878378	-0.169706241	273.632	-0.013833613	616.25	2.040109892
dark	16	-8.93275698	4	216.7895	-0.274228473	294.005051	0.027440506	278.235	-0.084425567	551.2051	0.185371595
dead	25	-9.02288301	4	301.4815	0.316153348	291.580858	0.05990301	266.049	-0.263521318	559.5385	0.434872934
demand	11	-9.0698531	6	237.6216	-0.090678545	272.716418	-0.116109237	274.646	-0.03001807	522.925	-0.690387485
desk	1	-10.0947792	4	226.7879	-0.309178283	303.857143	0.088769028	287.703	0.078425703	523.3784	-0.675977517
did	13	-7.15726438	3	245.4595	-0.016532231	287.252507	0.009551189	277.335	-0.097826379	538.325	-0.207780839
die	14	-9.84028515	3	209.641	-0.421562105	269.144928	0.008686786	271.514	-0.057101289	545.6316	0.016384555
dismay	4	-12.048683	6	440.3902	1.308427857			266.197	-0.087310242	586.1579	1.207673147
do	16	-6.63871039	2	237.227	-0.049322775	311.685897	0.289508595	279.565	0.012328458	524.025	-0.655447179
door	5	-8.40265634	4	198.7995	-0.451225745	305.284262	0.03320015	273.224	-0.114355283	498.4474	-1.487515822
dress	21	-9.93360143	5	212.3429	-0.280822616	276.467742	0.076511978	284.823	0.277446287	533.1667	-0.367878045
drink	10	-9.56237287	5	268.6341	0.228970301	311.890244	0.354931185	298.949	0.199581218	532.9167	-0.375676502
eat	6	-9.717667	3	252.7667	-0.087149592	283.119599	0.007667823	266.825	-0.175659023	524	-0.656240462
escape	15	-9.92192378	6	261.9048	0.081768141	281.911765	-0.020716831	291.862	0.069550373	527.05	-0.559738204
feel	17	-8.35257322	4	257.7436	-0.012541329	265.661765	-0.227231747	265.377	-0.165894898	523.4324	-0.674260321
felt	1	-8.20601411	4	218.8864	-0.159930976	277.483068	-0.085045546	286.514	0.022815993	578.8158	0.998083399
first	17	-6.71694175	5	261.0078	0.26218877	305.538696	0.20169066	278.731	-0.050082508	562.9189	0.535026568
five	3	-8.14686169	4	310.1818	0.629579668	312.074627	0.44909973	310.299	0.390189466	537.9211	-0.220262518

	WordNet senses	Word frequency	Word length	Eye RT	Standardised eye RT	Delay-corrected self RT	Standardised delay-corrected	Self RT	Standardised self RT	BLP RT	Standardised BLP RT
flashing	11	-11.8398739	8	332.9756	0.672511534			307.877	0.408384117	554.5897	0.287159436
follow	24	-9.21348883	6	255.0873	0.14638037	291.018551	0.088924636	306.551	0.225834898	544.4474	-0.019742365
forgot	4	-11.3917648	6	214.3333	-0.300553212	257.736842	-0.131398158	270.474	-0.107618785	553.5385	0.255610177
fork	9	-11.7076732	4	188.5161	-0.640924219	252	-0.287881188	268.468	-0.024211142	530.4103	-0.454063845
foul	16	-11.4853289	4	259.6471	0.165438053	279.492537	0.049356847	273.773	0.016374712	579.8684	1.028294825
gave	44	-8.40672049	4	218.373	-0.273010405	295.74881	0.14849444	285.899	0.024228172	624.4706	2.260450759
get	37	-7.32639586	3	207.4146	-0.346822403	286.526377	0.000243214	289.612	0.051701463	528.3684	-0.518196005
getting	37	-8.78140948	7	232.3876	-0.118540823	290.960652	0.129279939	291.558	0.130622961	587.6857	1.250956596
give	45	-7.86298002	4	228.1603	-0.121773407	284.342012	-0.007526915	290.163	0.103022345	502.7143	-1.345782345
glory	4	-11.0212061	5	222.1176	-0.297617347	268.013889	-0.061549645	261.403	-0.295816151	514.0541	-0.974879667
go	35	-7.39937052	2	209.6684	-0.299399132	282.305281	-0.046090357	272.451	-0.116854133	516.4865	-0.896385696
good	27	-7.25966311	4	246.55	0.097858504	315.083532	0.079178473	284.801	0.029744783	493.25	-1.661804074
got	36	-7.78040089	3	211.4298	-0.387087602	291.612503	0.071100756	294.91	0.077098311	594.1714	1.433454123
guard	14	-10.4028373	5	205.5122	-0.47750322	277.278912	-0.017346789	281.344	0.029744701	506.3333	-1.226508755
had	19	-5.47069666	3	225.0991	-0.095905956	290.229406	0.14323374	284.071	0.071667852	557.2727	0.367406213
hair	6	-8.91115538	4	216.7	-0.242991542	293.192672	0.125275677	288.558	0.080872153	510.7949	-1.080636894
hall	13	-9.0466039	4	243.5429	-0.100203865			293.625	0.295153153	510.1622	-1.10124589
hand	16	-7.98901521	4	235.7381	0.009911399	290.744444	0.076054369	299.079	0.125323664	524.7895	-0.631207675
happened	5	-9.01419831	8	197.6434	-0.532530338	284.984375	-0.280583641	289.842	-0.004306512	536.7105	-0.257723108
has	20	-5.95083601	3	202.5405	-0.431785825	278.23	-0.04379056	281.424	0.012114771	569.3714	0.724537974
hate	2	-10.6103934	4	336.7	0.725718602	281.25	0.066140911	259.406	-0.084725138	527.775	-0.536881297
have	20	-5.48661724	4	230.8837	-0.037722703	285.903242	0.051956822	292.725	0.135490418	574.0303	0.860039554
head	42	-7.90311384	4	313.4472	0.516256904	326.808333	-0.200914445	288.332	0.07837847	513.9211	-0.979182292
hear	5	-9.07109184	4	226.3	-0.195489666	273.542056	-0.126816492	271.656	-0.183396838	527.8684	-0.533938317
heaved	8	-12.5828757	6	397.8889	0.831202615	321.75	0.327528372	267.644	-0.259076927	616.4545	2.045628005
help	12	-7.90742271	4	211.0968	-0.429751841	265.376812	-0.176926349	255.435	-0.335803743	533.2105	-0.366510272
holding	38	-9.39383345	7	235.7714	-0.090026692	273.268817	-0.132312171	291.306	0.12630663	575.8	0.911222502
horse	6	-9.56518186	5	342.6667	0.863586325	283.971014	-0.068040115	285.543	0.035863948	510.55	-1.088609975
is	13	-4.64688144	2	232.0337	-0.057581854	289.78568	0.100570673	293.672	0.113404154	618.4865	2.100345356
joint	13	-9.53880975	5	306.8889	0.270657588			252.712	-0.159962674	544.6053	-0.014920904
kill	17	-10.0127786	4	202.6543	-0.339714514	308.068403	-0.082173312	289.806	0.059881139	505.9474	-1.239188388
knew	11	-8.32145655	4	279.5135	0.266716488	278.712367	-0.127709509	284.944	-0.042084797	614.1389	1.983050267
knife	4	-10.5517053	5	192.7597	-0.513206998	301.265152	0.179691295	283.665	0.0036329	544.95	-0.004398906
know	12	-7.30747056	4	298.5238	0.444360731	324.388462	0.1305278	280.042	-0.010223046	572.0833	0.803546933
lamb	6	-11.0770249	4	379.5862	0.695195278			267.307	-0.224608901	519.95	-0.785254598
laughed	1	-9.95321885	7	317.2308	0.552278094			259.618	-0.103188508	544.825	-0.008213365
lay	4	-9.20299005	3	233.8333	-0.052601861	287.931061	0.0570996	279.533	-0.045438238	560.9429	0.47655458
lead	32	-8.81081271	4	242.75	-0.013051148	310.909091	0.148875959	276.475	-0.060582652	526.2368	-0.585411883
leader	2	-9.2671225	6	290.7368	0.267696676			286.297	-0.037768394	519.8649	-0.787977378
leaning	10	-11.1744635	7	233.6216	-0.248166524	246.933333	-0.397429384	244.6	-0.346380963	613.0811	1.954385774
leave	17	-8.60885331	5	202.9286	-0.345047988	267.720238	-0.208412342	276.549	-0.086543475	522.1282	-0.715742741
let	8	-8.36977825	3	252.2386	0.113450921	281.97422	-0.032154299	291.768	0.067316186	546.3	0.036741591
lip	3	-11.0756085	3	246.9683	-0.042068239	446.714286	0.909016552	314.372	0.380408112	502.7949	-1.343117121
listening	4	-10.1591762	9	239.0244	-0.133971752	291.515464	0.166882366	309.959	0.397218088	611.6579	1.915742117
little	10	-7.44420596	6	249.8264	0.12224319	285.912041	0.054110708	284.403	0.001391088	542.45	-0.080854847
live	19	-8.71888912	4	236	-0.137987632	292.869565	-0.114575242	258.919	-0.206187084	508.4872	-1.155928067
living	18	-8.76225357	6	314.6667	0.384469653			276.514	0.013340831	544.2895	-0.024565223
lost	22	-8.59017119	4	208.5854	-0.362731613	285.590152	-0.04350067	277.377	-0.097803319	582.6667	1.108340965
lunch	3	-9.99508762	5	251.3984	-0.037584204	345.5	0.138558201	285.341	0.016385822	529.625	-0.47869876
me	1	-6.78046777	2	214.2133	-0.217392339	297.451852	-0.124246948	292.944	0.067234879	542.0541	-0.092996129
meet	15	-8.8827182	4	234.4865	-0.081726534	248.1	-0.265691327	256.014	-0.220280967	531.9737	-0.405124693
miss	12	-9.04893262	4	226.0541	-0.247590215	250.885714	-0.30835602	255.557	-0.273309069	553.475	0.253703762
morning	4	-8.71527539	7	249.481	0.062309198	330.883333	-0.191320393	286.051	0.025357013	563.1538	0.541964447
mouth	11	-9.307843	5	215.5	-0.367545251	285.252924	-0.023937542	281.032	-0.055541438	499.8947	-1.439303306
move	21	-8.59780414	4	231	-0.156362422	273.571429	-0.58275952	273.223	-0.056785786	581.9189	1.086988719
narrow	10	-9.88574752	6	219.5556	-0.320903072	257.393617	-0.292437488	267.826	-0.222610541	606.5	1.774932837
nodded	5	-9.90378169	6	249.7073	0.01803397	292.642157	0.002457778	298.005	0.022416211	594.5278	1.443423399
none	4	-9.31438848	4	283.1304	0.204790387	268.393443	0.025786436	308.262	0.596380416	526.725	-0.569994612
office	7	-8.2918037	6	191.3659	-0.576996342	279.700144	-0.012801507	279.053	-0.067114194	537.9487	-0.219407381
own	2	-7.26126931	3	310.5263	0.31736645	329.747917	0.261584827	286.967	0.039670453	544.75	-0.01050246
pay	12	-8.53932008	3	200.2424	-0.51159322	270.6	-0.023665808	268.507	-0.049473539	508.8947	-1.142606217
peace	5	-9.3007401	5	333.1765	0.620458224			253.2	-0.315042663	530.7895	-0.44218017
place	32	-7.63936371	5	236.35	-0.232458369			293.196	0.2130182	544.7778	-0.009654611
plate	16	-10.1260486	5	308.8	0.432495324			286.692	0.004176638	526.775	-0.568416291
pointed	15	-9.65112829	7	397.9	0.851582788			313.176	0.443587864	524.7632	-0.632041494

	WordNet senses	Word frequency	Word length	Eye RT	Standardised eye RT	Delay-corrected self RT	Standardised delay-corrected	Self RT	Standardised self RT	BLP RT	Standardised BLP RT
poor	7	-8.82331149	4	181.2222	-0.714754528	258.685714	-0.279681129	268.394	-0.160589653	513.0526	-1.007303303
prison	2	9.65027716	6	264.2927	0.196338154	320.75	0.291034545	290.442	0.02561609	510.3	1.096753977
proud	2	-10.3775195	5	242.3784	-0.065459275	275.428571	-0.127399894	303.371	0.138102239	555.9474	0.327813757
put	10	-7.71920713	3	248.1707	0.080891294	289.877239	0.064592963	292.856	0.067558447	573.8378	0.85446359
ran	41	-9.3714396	3	228.1481	-0.200329013	280.450617	0.008148577	287.427	0.069301746	549.3611	0.129651871
refused	6	-9.62391345	7	216.9412	-0.303576623	277.352113	0.088113772	282.563	-0.008207782	566.8649	0.651176071
riding	17	-10.5858092	6	214.3333	-0.369834324	277.902778	0.081366201	257.246	-0.268331136	539.2647	-0.178780811
ring	15	-9.77903134	4	294.7692	0.203504444			293.579	0.109647424	523.8108	-0.662244912
room	5	-8.15198281	4	218.0556	-0.230591693	372.874242	0.18034685	301.944	0.243425867	561.5263	0.49384062
row	8	-9.87631753	3	285.6	0.212829553			261.197	-0.303438394	530.0769	-0.464516657
sat	9	-9.09375324	3	249.9785	0.180647374	288.779093	0.09116682	293.88	0.136508358	612.6154	1.941750641
saw	3	-8.28948823	3	240.5789	-0.091199352	270.721649	-0.147598481	263.206	-0.294960683	497.8205	-1.508440137
sent	10	-8.92894492	4	255.7	0.130161686	287.696637	0.098170695	291.943	0.108326742	522.25	-0.711864534
shave	7	-12.8526153	5	217.2571	-0.314128941	266.741379	0.017234319	260.328	-0.169578796	551.6667	0.199288501
ship	6	-10.0703974	4	250.1429	0.04169599	284.225806	-0.239048604	276.2	-0.152689648	517.0789	-0.877323121
shivered	2	-11.9729712	8	496.5714	1.394470186			306.015	0.071599056	596.6757	1.503387034
shoe	5	-11.4611045	4	292.2353	0.219391119			304.42	0.187786322	545.1351	0.00124901
shook	10	-9.87057467	5	229.2093	-0.103451221	277.001502	-0.093104285	278.02	-0.117167905	594.9375	1.454878464
shop	7	-9.30398432	4	356.6486	0.72666262			300.359	0.046355169	557.1081	0.362493668
shoulder	7	-9.97017825	8	265.1579	-0.054376812	314.125	0.208591011	288.046	0.071175312	518.4474	-0.833377192
shut	5	-10.1331935	4	274.56	0.179269455	292.919355	0.033393524	304.635	0.248439496	528.9444	-0.500078582
sighed	2	-10.8018229	6	319.3333	0.654003697	296.75303	0.136292315	293.875	0.038732897	608.2222	1.822081816
sit	8	-9.60438382	3	309.1429	0.557015864	276.892473	-0.113281094	273.677	-0.114842868	539.0769	-0.184571887
slammed	4	-11.6308108	7	319.3684	0.51764758	284.615782	0.031038678	289.695	0.084376048	607.1026	1.791444339
slap	4	-12.3603713	4	242.5528	-0.012801198	318.714286	0.143597325	289.516	0.073792165	567.7895	0.678275138
smart	9	-11.0286061	5	221.4194	-0.305775866	309.306931	0.249884517	274.88	-0.083016377	515.6053	-0.924779761
soldier	3	-10.9934449	7	241.4884	0.054084827	329.039007	0.256043085	315.236	0.277370562	564.641	0.585816842
somewhere	2	-9.84090262	9	235.8919	-0.090335152	263.194444	-0.132812711	255.944	-0.241231204	572.0263	0.801889635
spare	13	-10.4970647	5	226.7317	-0.154053072	263.542105	-0.238197257	254	-0.379399987	564	0.566929128
spoke	7	-9.61796498	5	214.7059	-0.452633367	283.074074	-0.051731016	291.783	0.015275134	580.6389	1.050373168
staggered	5	-11.9626084	9	355.1579	0.797693654	287.114286	-0.012327955	291.754	0.018829665	618.9459	2.112693041
steady	10	-10.4486525	6	225.3659	-0.266120093	292.34375	0.04548931	267.727	-0.078897303	526.6154	-0.573455301
step	21	-9.3329223	4	230.6047	-0.146907013	311.368217	-0.153041616	281.622	-0.05019665	525.625	-0.604755625
stood	12	-8.95364787	5	237.7984	0.008274719	300.837274	0.210343678	291.034	0.109314208	530.4324	-0.453368673
strength	9	-9.51772463	8	270.9744	0.203837927			296.086	0.02490058	487.2564	-1.86508741
sure	10	-8.50333666	4	226.2667	-0.324582477	296.884615	0.093152158	298.25	0.26681177	504.8158	-1.276418594
sword	1	-11.1612697	5	239.6129	-0.033351523	289.693146	0.057312645	283.519	-0.037750834	537.25	-0.241018289
swung	13	-10.9309048	5	219.3	-0.240027457	289.465753	0.158096027	283.137	0.076586363	622.9118	2.218892497
table	6	-8.54770245	5	242.8403	0.071414021	280.033333	-0.56827144	279.58	-0.022084051	573.2941	0.838701205
talk	11	-8.85468335	4	207.4124	-0.317986735	316.052967	0.032088636	281.064	-0.070190701	486.9211	-1.876535417
tell	9	-8.33296655	4	215.4286	-0.238568089	275.598778	-0.0687801	278.483	-0.057945377	536.4737	-0.265062236
think	14	-7.5625326	5	284.3529	0.290083248	235.230769	-0.469650204	235.106	-0.475959691	564.4474	0.580113026
toilet	4	-11.3214783	6	269.1905	0.17446827	346.666667	0.480914417	259.758	-0.267603031	520.7297	-0.760338107
told	8	-7.96740726	4	204.8333	-0.34023992	277.597713	-0.064605554	283.254	0.012784866	532.5946	-0.385728556
took	42	-7.89012186	4	214.7302	-0.231712786	289.318966	0.089095938	307.159	0.270694272	572.7895	0.824058254
train	17	-9.46815414	5	251.25	-0.059860681	278.974026	0.023950867	258.896	-0.300370197	546.1026	0.030731193
trick	8	-11.0970685	5	287.1111	0.144828195			293.784	0.317201306	545.2821	0.005729615
truck	3	-11.3917648	5	261.1008	0.257435572	326.405527	0.025992306	285.072	-0.009886665	580.875	1.057133248
tube	9	-10.8148242	4	365.9394	0.893000624	258.896104	-0.341153372	264.089	-0.217414652	527.6579	-0.540571121
tucked	4	-11.5803193	6	213.5556	-0.361538641	285.53	-0.058229024	270.773	-0.241843394	637.2222	2.596564906
turn	38	-8.54612187	4	307.619	0.409993511	343.65	0.402204561	280.522	0.040911212	566.0769	0.628047628
two	2	-6.5968733	3	233.2381	-0.100170165	298.972029	0.151139708	280.726	-0.018172572	520.35	-0.772467817
van	4	-10.101169	3	201.0351	-0.57272475	275.280374	-0.103513448	280.897	-0.01303583	548.4615	0.102402032
voice	13	-8.29447533	5	379.5714	0.822501385	308.259259	0.445144034	288.963	0.138445018	510.3158	-1.096239501
wait	6	-9.54170588	4	207	-0.356448493	256.097826	-0.34672137	262.027	-0.255538852	526.3889	-0.580608325
walk	17	-9.29118893	4	225.7647	-0.396164427	270.559748	-0.189465237	272.286	-0.106958564	537.2703	-0.240390947
wall	9	-9.07807651	4	201.2381	-0.409772937	315.788069	0.046193622	294.693	0.070072634	532.6053	-0.385395487
want	9	-7.81816543	4	206.2585	-0.357591593	271.119322	-0.191658707	277.656	-0.050820767	511.0256	-1.073126494
was	14	-4.75819217	3	216.3066	-0.178767708	292.654441	0.144532926	287.757	0.100013803	543.4848	-0.049164067
watch	13	-9.4401842	5	197.4444	-0.541449883	255.72973	-0.353530736	254.747	-0.303256898	525.2162	-0.617692113
way	13	-7.01070949	3	260.8434	0.21779417	372.729167	0.211836728	282.009	-0.024634088	512.725	-1.017924961
went	30	-7.83143608	4	324	0.545957131	285.917105	0.017281718	292.208	0.091745956	536.8462	-0.253521818
were	13	-5.78163529	4	212.1383	-0.325178982	299.994286	0.1883242	293.749	0.109265493	601.9706	1.650289936
wiped	1	-11.3976071	5	225.2632	-0.146324081	272.477528	-0.120378882	255.944	-0.302811009	585.7027	1.194755646
wood	8	-9.54813775	4	210.1463	-0.337520934	313.059783	0.329524082	260.87	-0.181616093	510.7027	-1.083637492
wrong	12	-8.88074574	5	200	-0.581945833	292	-0.067143855	286.876	0.107371744	489.4324	-1.790996043
wrote	10	-9.21469888	5	237.6	-0.100136547	274.441558	-0.084337046	266.081	-0.302788378	539.8718	-0.160072447
yawn	3	-13.2986409	4	319.4286	0.525182774			273.667	-0.279077215	557.0769	0.361562884

C. WordNet senses and reading time.

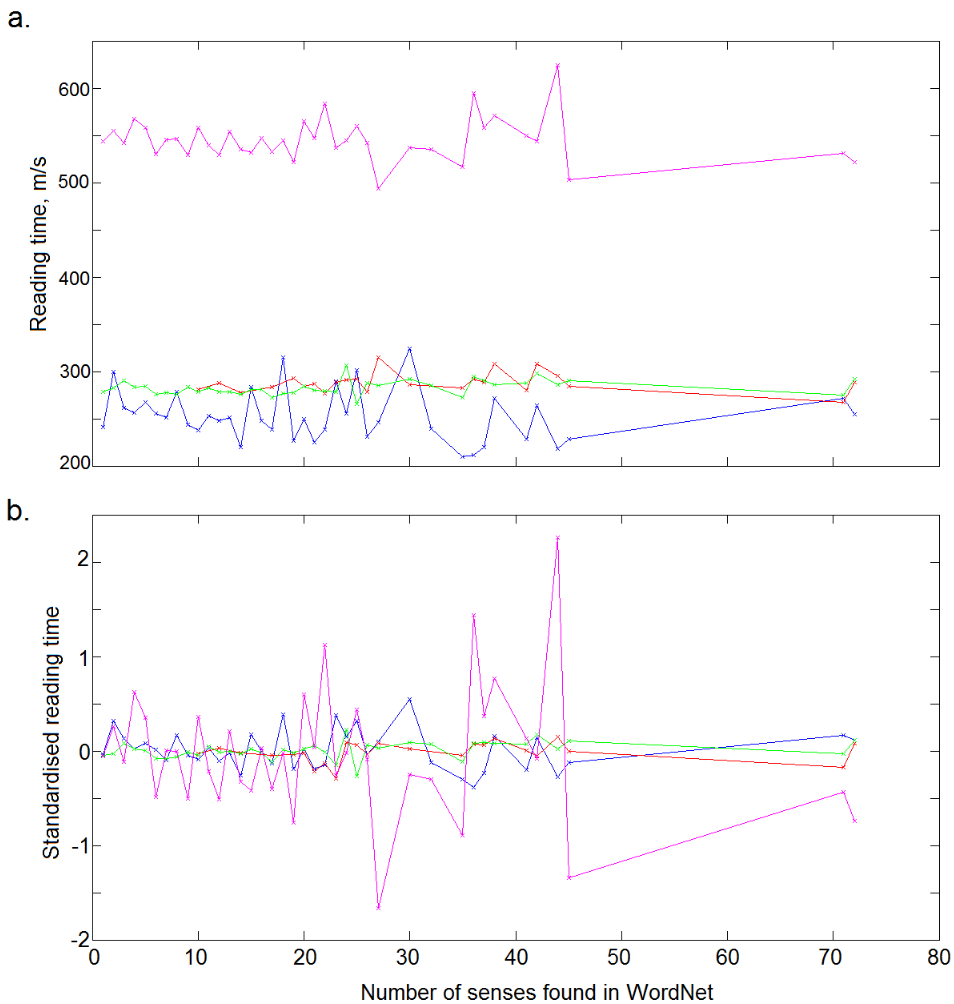

Figure 17: Average reading times (a) and average standardised reading times (b) for different number of senses measured by WordNet. For in-context words with RT measured by eye tracking (blue) and self-paced (with delay correction) (red) and self-paced (no correction) (green) and also out of context words from the BLP RT dataset (magenta).

D. Finding the clustering threshold

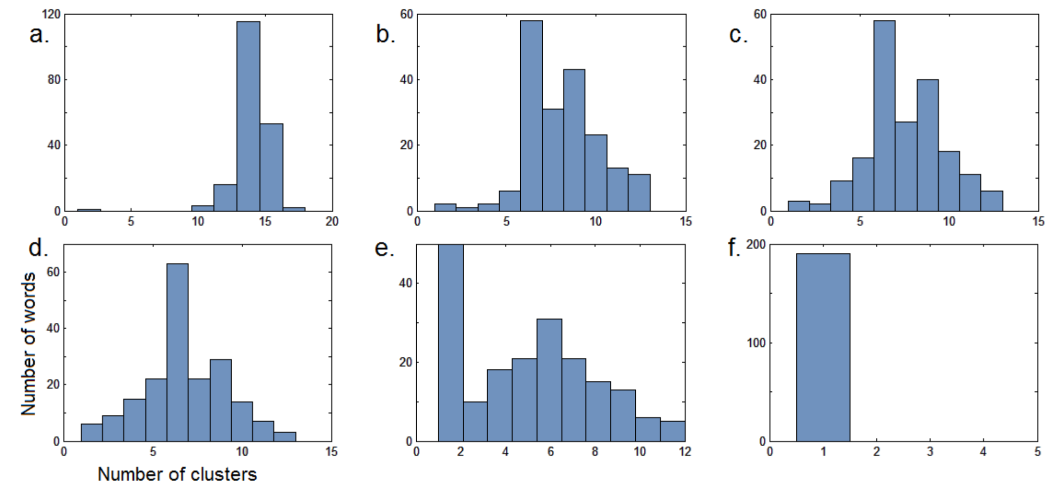

Figure 18: Histograms of the number of clusters found for all the words using different clustering thresholds; 0.7 (a), 0.8 (b), 0.9(c), 1(d), 1.1(e), 1.2(f). (c) to (e) show a good distribution of values.

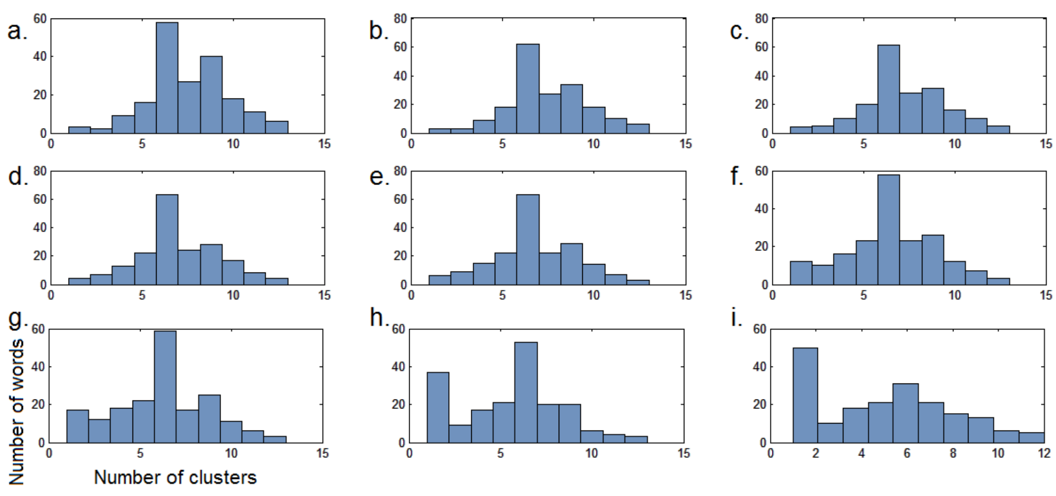

Figure 19: Looking closer. Histograms of the number of clusters found for all the words using different clustering thresholds; 0.9 (a), 0.925 (b), 0.95(c), 0.975(d), 1(e), 1.025(f), 1.05 (g), 1.075 (h), 1.1 (i).

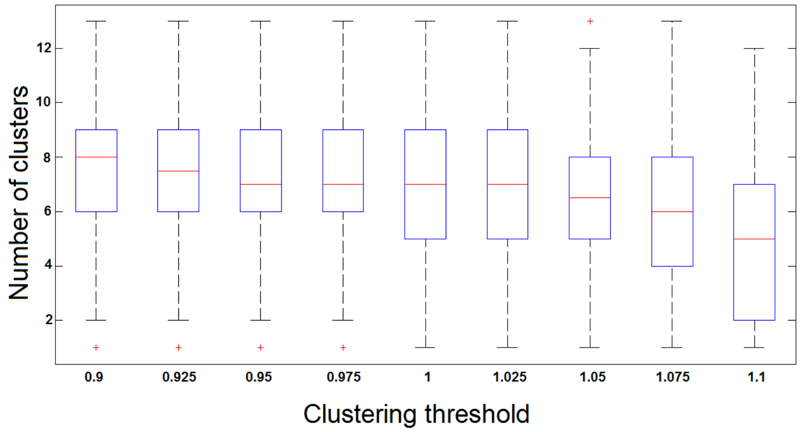

Figure 20: Box and whisker diagrams of the number of clusters found for all the words using different clustering thresholds.

E. Correlation coefficients

	Number of WordNet senses	Word Frequency	Word Length	Average top 20 similarity score	Number of top 20 neighbours with a similarity score ≥ 0.55	Number of top 20 neighbours with a similarity score ≥ 0.75	**Average neighbour-to-neighbour similarity score for top 20 neighbours**	Average neighbour-to-neighbour similarity score for top 20 neighbours with a ≥ 0.55 similarity score	Average neighbour-to-neighbour similarity score for top 20 neighbours with a ≥ 0.75 similarity score	In-context eye tracking RT	In-context self paced (delay corrected) RT	In-context self paced RT	Out-of-context BLP RT	Standardised in-context eye tracking RT	Standardised in-context self paced (delay corrected) RT	Standardised in-context self paced RT	Standardised out-of-context BLP RT	**Number of clusters (found using a clustering threshold of 1)**	**Number of clusters (found using a clustering threshold of 1.025)**	**Clustering coefficient (for neighbour-to-neighbour similarity scores ≥ 0.55)**	Number of top 20 neighbour-to-neighbour pairs with a similarity score ≥ 0.55
Number of WordNet senses		0.321	-0.171	0.043	0.037	-0.03	0.236	0.126	0.181	-0.099	-0.078	0.062	-0.043	-0.078	-0.034	0.097	-0.042	0.043	0.067	0.208	0.208
Word Frequency			-0.444	0.55	0.393	0.463	0.492	0.419	0.499	-0.295	-0.012	0.108	-0.112	-0.243	-0.016	0.121	-0.109	0.1	0.105	0.341	0.425
Word Length				-0.165	-0.039	-0.226	-0.155	-0.18	-0.446	0.287	0.03	0.091	0.226	0.263	0.005	0.085	0.226	-0.111	-0.137	-0.09	-0.114
Average top 20 similarity score					0.811	0.724	0.674	0.546	0.309	-0.145	0.043	0.124	0.132	-0.068	0.095	0.139	0.131	-0.017	-0.032	0.585	0.707
Number of top 20 neighbours with a similarity score ≥ 0.55						0.306	0.565	0.36	-0.051	-0.134	-0.054	0.07	0.131	-0.083	-0.052	0.099	0.129	-0.036	-0.066	0.516	0.608
Number of top 20 neighbours with a similarity score ≥ 0.75							0.469	0.461	0.278	-0.08	0.135	0.13	0.006	-0.01	0.193	0.133	0.006	0.04	0.03	0.364	0.449
Average neighbour-to-neighbour similarity score for top 20 neighbours								0.871	0.566	-0.114	0.019	0.097	0.123	-0.042	0.1	0.121	0.124	0.177	0.139	0.832	0.944
Average neighbour-to-neighbour similarity score for top 20 neighbours									0.571	-0.04	0.091	0.171	0.091	0.012	0.169	0.193	0.093	0.146	0.118	0.744	0.83
Average neighbour-to-neighbour similarity score for top 20 neighbours										-0.021	-0.187	-0.197	0.046	-0.034	-0.039	-0.133	0.052	0.284	0.274	0.41	0.401
In-context eye tracking RT											0.192	0.1	0.134	0.966	0.164	0.073	0.136	-0.19	-0.235	-0.04	-0.092
In-context self paced (delay corrected) RT												0.517	-0.106	0.228	0.723	0.445	-0.111	-0.087	-0.078	0.063	0.008
In-context self paced RT													0.017	0.148	0.467	0.903	0.016	-0.02	-0.022	0.14	0.105
Out-of-context BLP RT														0.144	-0.034	-0.024	0.999	-0.143	-0.161	0.106	0.143
Standardised in-context eye tracking RT															0.172	0.107	0.146	-0.173	-0.2	0.044	-0.017
Standardised in-context self paced (delay corrected) RT																0.497	-0.036	-0.049	-0.05	0.102	0.087
Standardised in-context self paced RT																	-0.024	0.004	0.024	0.123	0.12
Standardised out-of-context BLP RT																		-0.138	-0.157	0.106	0.142
Number of clusters (found using a clustering threshold of 1)																			0.938	0.163	0.122
Number of clusters (found using a clustering threshold of 1.025)																				0.134	0.088
Clustering coefficient (for neighbour-to-neighbour similarity scores ≥ 0.55)																					0.875
Number of top 20 neighbour-to-neighbour pairs with a similarity score ≥ 0.55																					

Figure 21: Correlation coefficients between different parameters. Highlighted squares show significant correlations (with a p-value ≤ 0.05).

	Number of WordNet senses	Average top 20 similarity score	Number of top 20 neighbours with a similarity score ≥ 0.55	Number of top 20 neighbours with a similarity score ≥ 0.75	Average neighbour-to-neighbour similarity score for top 20 neighbours	Average neighbour-to-neighbour similarity score for top 20 neighbours with a ≥ 0.55 similarity score	Average neighbour-to-neighbour similarity score for top 20 neighbours with a ≥ 0.75 similarity score	In-context eye tracking RT	In-context self paced (delay corrected) RT	In-context self paced RT	Out-of-context BLP RT	Standardised in-context eye tracking RT	Standardised in-context self paced (delay corrected) RT	Standardised in-context self paced RT	Standardised out-of-context BLP RT	Number of clusters (found using a clustering threshold of 1)	Number of clusters (found using a clustering threshold of 1.025)	Clustering coefficient (for neighbour-to-neighbour similarity scores ≥ 0.55)	Number of top 20 neighbour-to-neighbour pairs with a similarity score ≥ 0.55
Number of WordNet senses		-0.166	-0.098	-0.214	0.099	0.016	0.028	0.001	-0.077	0.035	-0.001	0.006	-0.03	0.068	0	0.009	0.032	0.114	0.087
Average top 20 similarity score			0.772	0.64	0.551	0.433	0.079	0.002	0.055	0.062	0.217	0.064	0.123	0.072	0.215	-0.079	-0.099	0.502	0.622
Number of top 20 neighbours with a similarity score ≥ 0.55				0.158	0.458	0.295	-0.066	-0.052	-0.058	0.005	0.164	-0.015	-0.05	0.032	0.16	-0.071	-0.102	0.437	0.525
Number of top 20 neighbours with a similarity score ≥ 0.75					0.316	0.333	0.064	0.074	0.159	0.096	0.072	0.125	0.227	0.093	0.071	-0.01	-0.024	0.25	0.319
Average neighbour-to-neighbour similarity score for top 20 neighbours						0.841	0.397	0.023	0.026	0.038	0.194	0.079	0.123	0.059	0.193	0.155	0.11	0.811	0.933
Average neighbour-to-neighbour similarity score for top 20 neighbours with a ≥ 0.55 similarity score							0.402	0.073	0.053	0.12	0.149	0.101	0.138	0.145	0.15	0.099	0.068	0.706	0.801
Average neighbour-to-neighbour similarity score for top 20 neighbours with a ≥ 0.75 similarity score								0.219	-0.155	-0.222	-0.081	0.169	-0.101	-0.163	-0.074	0.19	0.155	0.248	0.241
In-context eye tracking RT									0.191	0.113	0.073	0.963	0.165	0.089	0.075	-0.158	-0.2	0.055	0.022
In-context self paced (delay corrected) RT										0.526	-0.117	0.227	0.723	0.452	-0.121	-0.084	-0.075	0.069	0.011
In-context self paced RT											-0.001	0.157	0.48	0.899	-0.003	-0.02	-0.018	0.101	0.053
Out-of-context BLP RT												0.089	-0.036	-0.043	0.999	-0.121	-0.134	0.143	0.199
Standardised in-context eye tracking RT													0.172	0.117	0.092	-0.144	-0.166	0.129	0.084
Standardised in-context self paced (delay corrected) RT														0.508	-0.038	-0.048	-0.049	0.115	0.103
Standardised in-context self paced RT															0.043	0.004	0.028	0.078	0.064
Standardised out-of-context BLP RT																-0.116	-0.13	0.142	0.196
Number of clusters (found using a clustering threshold of 1)																	0.937	0.144	0.096
Number of clusters (found using a clustering threshold of 1.025)																		0.113	0.058
Clustering coefficient (for neighbour-to-neighbour similarity scores ≥ 0.55)																			0.857
Number of top 20 neighbour-to-neighbour pairs with a similarity score ≥ 0.55																			

Figure 22: Pearson linear partial correlation coefficients whilst controlling for word length and frequency. Highlighted squares show significant correlations (with a p-value ≤ 0.05).

F. Number of clusters for "chair"

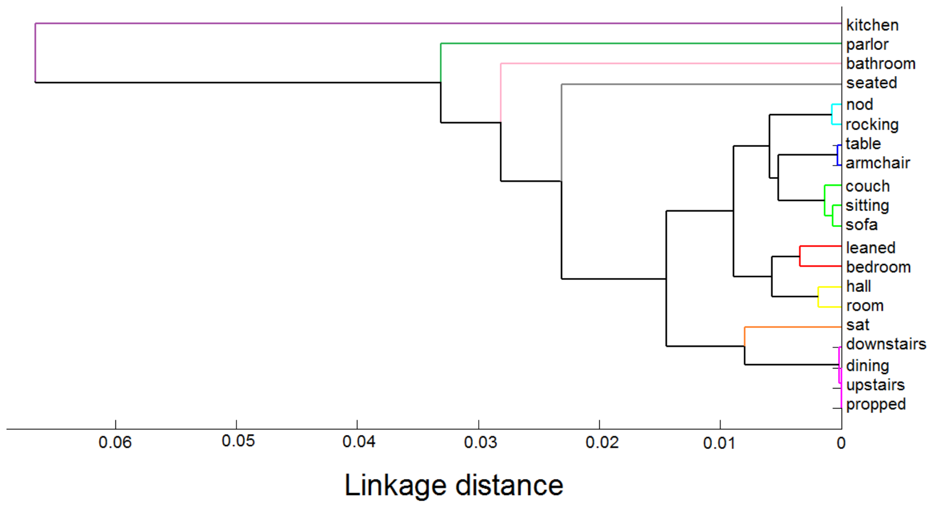

Figure 23: Dendogram showing the clusters of the top 20 neighbours of "chair". The ten clusters found with a cluster threshold of 1 are shown in different colours.

References

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

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. 1990. Indexing by latent semantic analysis. *Journal of the American society for information science* 41(6): 391-407.

Dumais, S. T., & Landauer, T. K. 1997. A solution to Platos problem: The latent semantic analysis theory of acquisition, induction and representation of knowledge. *Psychological review* 104(2): 211-240.

Frank, S. L., Monsalve, I. F., Thompson, R. L., & Vigliocco, G. 2013. Reading time data for evaluating broad-coverage models of English sentence processing. *Behavior Research Methods* 1-9.

Keuleers, E., Lacey, P., Rastle, K., & Brysbaert, M. 2012. The British Lexicon Project: Lexical decision data for 28,730 monosyllabic and disyllabic English words. *Behavior Research Methods* 44(1): 287-304.

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

Leacock, C., Towell, G., & Voorhees, E. M. 1996. Towards building contextual representations of word senses using statistical models. *Corpus Processing for Lexical Acquisition, MIT Press, Cambridge, Massachusetts.* 97-113.

Miller, G. A., Beckwith, R., Fellbaum, C., Gross, D., & Miller, K. J. 1990. Introduction to wordnet: An on-line lexical database. *International journal of lexicography* 3(4): 235-244.

Miller, G. A., & Fellbaum, C. 1991. Semantic networks of English. *Cognition* 41(1): 197-229.

Rodd, J., Gaskell, G., & Marslen-Wilson, W. 2002. Making sense of semantic ambiguity: Semantic competition in lexical access. *Journal of Memory and Language* 46(2): 245-266.

Rubenstein, H., Garfield, L., & Millikan, J. A. 1970. Homographic entries in the internal lexicon. *Journal of Verbal Learning and Verbal Behavior* 9(5): 487-494.