Using Curvature and Markov Clustering in Graphs for Lexical Acquisition and Word Sense Discrimination

Appeared in MEANING-2005, 2nd Workshop organized by the MEANING Project, February 3rd-4th 2005, Trento, Italy

Beate Dorow
Institute for NLP
University of Stuttgart, Germany
beate.dorow@ims.uni-stuttgart.de

Dominic Widdows, Katarina Ling
CSLI, Stanford University, California
dwiddows@csli.stanford.edu
katarinaling@stanford.edu

Jean-Pierre Eckmann, Danilo Sergi
Département de Physique Théorique
Université de Genève, Switzerland
Jean-Pierre.Eckmann@physics.unige.ch
Danilo.Sergi@physics.unige.ch

Elisha Moses
Department of Physics of Complex Systems
Weizmann Institute of Science
Rehovot, Israel
fnmoses@wicc.weizmann.ac.il

Abstract

In order to use linguistic knowledge to build intelligent applications in large-scale environments such as the World Wide Web (WWW), it is vital that methods of determining meaning and recognizing ambiguity should be automatic and empirical. Methods for learning meaning must be simple, adaptive and scalable.

We introduce two complementary approaches for categorizing words which exhibit these desireable properties, and can recognize meaning and ambiguity with great accuracy. Both methods use a graph-theoretic representation of words and their paradigmatic relationships. Ambiguity is specifically addressed and accommodated by allowing a word to belong to several clusters.

The first approach is based on the concept of *curvature* and divides the word graph into classes of similar words by removing words of low curvature which connect several dispersed clusters. The second method clusters the links in our graph instead of clustering the nodes. Links contain more specific contextual information than nodes representing words. We thus naturally accommodate ambiguity by allowing multiple class membership.

1 Introduction

Graphs have been widely used to model many practical situations (Chartrand, 1985), including semantic issues: The link structure of the WWW has been investigated and manipulated to detect shared interest communities (Eckmann and Moses, 2002), and modeling WordNet as a graph has yielded insight about semantic relatedness and ambiguity (Sigman and Cecchi, 2002).

In this paper, we present a graph model for nouns and their conceptual similarity collected from the British National Corpus (BNC)[1]. Simple regular expressions were used to search the text for coordinations (lists) of noun phrases whose constituents are often related by shared characteristics.

The resulting semantic structure can be used for classification of words (by gathering nodes into clusters and labeling the clusters) and ambiguity recognition (by determining when a node in the graph connects several dense subgraphs representing different senses) which are important tasks in large-scale web-type applications.

We introduce two tools to approach these tasks: the curvature measure of Eckmann and Moses (2002) and the Markov Clustering (MCL) of van Dongen (2000). The first algorithm removes the nodes of low curvature (the hubs of the graph), upon which the word graph breaks up into disconnected coherent semantic clusters. MCL decomposes the word graph into small coherent pieces via simulation of random walks in the graph which eventually get trapped in dense regions, the resulting clusters.

Both methods effectively place each node into exactly one cluster, breaking the graph into equivalence classes. The shortcomings of any such approach become apparent once we consider ambiguity—when each word is treated as an indivisible unit in the graph, we need to split these semantic atoms to account for different senses. We then investigate an alternative approach which treats each individual coordination pattern as a semantic node, and agglomerates these more contextual units into usage clusters corresponding closely to word senses.

2 The graph model

To build a graph representing the relationships between nouns, we used simple regular expressions to

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

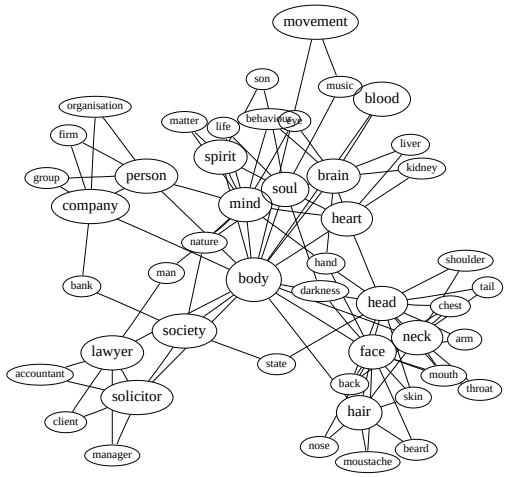

Figure 1: Local graph around *body*. Three areas of meaning are visible, namely *body* "torso", *body* "social group" and *body* "matter".

search the BNC, which is tagged for parts of speech, for examples of lexicosyntactic patterns which are often indicative of a semantic relationship (Hearst, 1992). The hypothesis is that nouns in coordinations are semantically similar (cf. Riloff and Shepherd (1997), Roark and Charniak (1998), Widdows and Dorow (2002)). We collected coordinations of noun phrases using simple patterns, dropped prenominal modifiers, and built a word graph by

1. Introducing a node for each of the nouns;
2. Connecting two nouns by an edge if they co-occurred in a coordination, more precisely, if they are separated by "and", "or", and commas.

For example, the coordination

> a recognised body or an individual solicitor or registered foreign lawyer

gives rise to edges *body*↔*solicitor*, *body*↔*lawyer* and *solicitor*↔*lawyer* in the word graph.

Figure 1 displays a particular example of the subgraph centered around *body* and consisting of the top 17 neighbors of *body* and the top 8 neighbors of these neighbors (where the neighbors are ranked according to their frequency of co-occurrence with *body* in lists). The word graph has this very simple interpretation: Words which are directly linked are semantically close. The aim of our procedures is to disentangle the several meanings of body visible in the graph. The graph thus obtained consists of $88,900$ nodes (word types) and $551,745$ edges. We ignore the order in which two words co-occur in a coordination, the edges in our graph are not given any direction. In the next step we keep only those

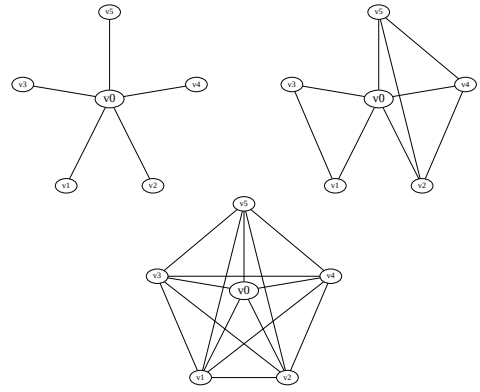

Figure 2: Neighborhood of a node of minimum, medium and maximum curvature, Eq. (1). Left: $\text{curv}(v_0) = 0$, Right: $\text{curv}(v_0) = 0.4$, Bottom: $\text{curv}(v_0) = 1$.

links in the graph which appear in a triangle. This filters out the less important links, since by transitivity a triangle's links confirm each other's significance. This decreases the noise significantly, and results in a reduced word graph consisting of $48,727$ nodes and $505,412$ edges.

3 Graph curvature and quantifying semantic ambiguity

Words which link several unrelated areas in the graph are likely to be ambiguous. On the other hand, words in tightly-knit node groups tend to be quite definite in their meaning. Words in such strong communities can be recognized because their neighbors are often closely linked to one another.

We measure the semantic cohesiveness of a word's neighborhood (and as a result ambiguity) as the *curvature* (also referred to as *clustering coefficient* (Watts and Strogatz, 1998)) of the word in the graph. Curvature is a property of nodes in a graph which quantifies the interconnectedness of a node's neighbors. The curvature $\text{curv}(w)$ of a node w is defined by:

$$\text{curv}(w) = \frac{\#(\text{triangles } w \text{ participates in})}{\#(\text{triangles } w \text{ could participate in})} \qquad (1)$$

Curvature is the fraction of existing links among a node's neighbors out of all possible links between neighbors. It assumes values between 0 and 1. A value of 0 occurs if there is no link between any of the node's neighbors, and a node has a curvature of 1 if all its neighbors are linked (see Fig. 2). Curvature measures whether neighbors of a word are neighbors of each other. Very specific, unambiguous words have high curvature, because they usually live in small, semantically very cohesive

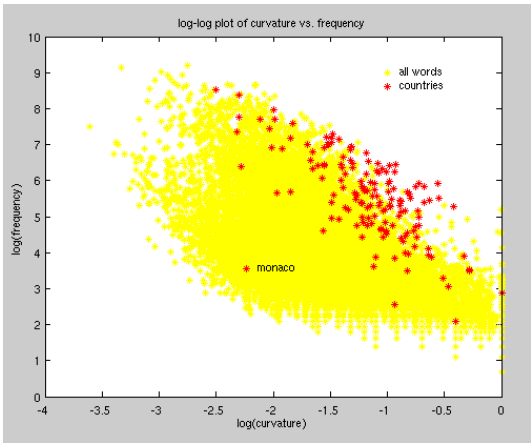

Figure 3: Curvature vs. frequency. Note that countries (black stars) have substantially higher curvature values than other words of similar frequencies, meaning that they are very specific. The outlier is *monaco*.

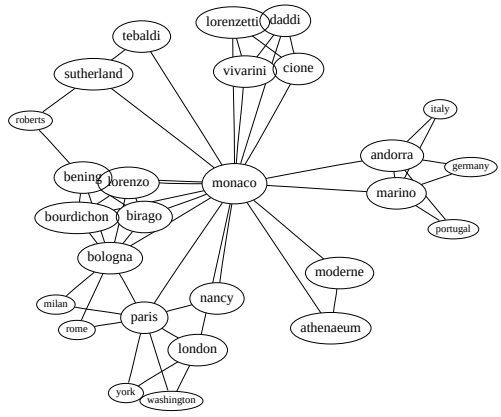

Figure 4: Local graph around *monaco* showing its several meanings.

communities in which many pairs of nodes have mutual neighbors. These communities thus contain a high density of triangles. Examples for tight word communities are the days of the week, the world religions, Greek gods, chemical elements, English counties, the planets, the members of a rock band, etc. Ambiguous words, on the other hand, are linked to members of different communities (corresponding to the different meanings of w) which do not know each other. An ambiguous word's neighborhood thus has a low density of triangles which results in a low curvature value.

In information theory, it is common to use the negative logarithm of relative word frequency to measure a word's information content ($\text{info}(w) = -\log(\text{rf}(w))$) (Shannon, 1948). The intuition is that very frequent words tend to be very general and uninformative, and that very infrequent words tend to be more specific. Among the most frequent words in coordinations are countries, which according to $\text{info}(\cdot)$ would be wrongly categorized as very uninformative, ambiguous words.

Figure 3 is a plot of curvature against frequency. The countries among the nodes are indicated by black stars. Very clearly, the curvatures of countries are considerably higher than the average curvature of words with similar frequency, suggesting that, despite their high frequency, they are all very informative, i.e., unambiguous. The outlier in the lower left corner of the plot is *monaco* which may not seem ambiguous, but which has several different meanings in the BNC: country, city, 14th century painter and 20th century tenor (cf. Fig. 4).
To check how well curvature is suited for detecting and assessing ambiguity, we took all words in our model which are listed in WordNet and checked how strongly curvature and the number of WordNet senses are related. Since the relationship does not have to be linear, we replace curvature and number of WordNet senses by their ranks before computing the Pearson correlation coefficient. We also checked whether and to which degree curvature better reflects ambiguity than a word's frequency or its *degree* (the number of links attached to it) in the graph. Table 1 lists the mutual correlations between any two quantities out of frequency, degree, curvature and number of WordNet senses. Our analysis shows that with a negative correlation of -0.538, curvature is more strongly related to the number of WordNet senses than frequency or degree[2]. This demonstrates that our combinatoric analysis does a significantly better job than raw frequency at predicting whether a word is ambiguous.

Table 1: Rank correlations between any two out of number of WordNet senses, word frequency, degree and curvature. The number of WordNet senses is more strongly correlated with curvature.

	senses	freq	deg	curv
senses	1.000	0.475	0.480	-0.538
freq		1.000	0.963	-0.865
deg			1.000	-0.884
curv				1.000

4 Inducing classes of similar words

A semantic category (also referred to as a semantic field) is a grouping of vocabulary within a language, organizing words which are interrelated and define each other in various ways. The acquisition of semantic categories from text has been addressed in several different ways: Work in this direction can be

[2]Recall that the sign of the correlation is irrelevant.

found in (Pereira et al. (1993), Schütze (1998), Pantel and Lin (2002), Dorow and Widdows (2003)).

Word clustering techniques differ in the way they assign words to clusters, either allowing words to belong to several clusters (soft clustering), or assigning words to one and only one cluster (hard clustering). Hard clustering techniques cannot detect the multiple meanings of a word. We therefore concentrate on soft clustering.

4.1 Graph clustering

We now describe two approaches to soft clustering of words in our graph.

Curvature clustering: In our word graph, ambiguous words function as bridges between different word communities, e.g., *cancer* is the meeting point of the animal community, the set of lethal diseases and the signs of the zodiac. By removing these "semantic hubs", the graph decomposes into small pieces corresponding to cohesive semantic categories. In detail, the method for extracting clusters of similar words is the following:

1. Compute the curvature of each node in the graph.
2. Remove all nodes whose curvature falls below a certain threshold, set to 0.5 from here on.[3]
3. The resulting connected components constitute clusters of semantically similar words.

Application of this algorithm to our word graph results in 700 clusters of size ≥ 2. The resulting clustering covers $2,306$ of the nouns in our model with $21,218$ of the nodes not making the curvature threshold and $25,203$ dangling nodes.

This method produces a hard clustering of the high curvature words. Since high curvature words have a well-defined meaning, we expect a hard clustering approach to detect the (unique) semantic category each of these words belongs to. Inspection of the clusters obtained shows that this is indeed the case to a very high degree of accuracy.

Curvature clustering in this form cannot give information on the semantically fuzzy low curvature words. Therefore, we augment each of the clusters with the nodes directly attached to it (including also the nodes which are not part of a triangle). Table 2 lists some of the enriched clusters. The original cluster (the core of the extended cluster) is printed in bold font, cluster neighbors which did not pass the curvature threshold are highlighted in italics, and dangling neighbors (neighbors which do not occur in a triangle) are printed in normal font. It is worth noticing that the core words of high curvature (bold) are quite specific and unambiguous, suggesting that high curvature is a desirable property for 'seed words' (cf. Roark and Charniak (1998)). By extending the core clusters to their neighbors, coverage could be increased to $9,962$ nodes in the graph.

Table 2: Clusters resulting from the curvature approach.

applewood fruitwood *cherry ivory pine oak*
jainism sikhism vaisnavism *islam buddhism hinduism christianity judaism*
horseflies lacewings *butterfly mosquito beetle centipedes ladybird bird moth*
freestyle backstroke *butterfly race medley*
printmaker ceramicist *sculptor painter draughtsman artist*
pomelo papaya *banana potato pineapple mango peach palm pear parsnip*
poliomyelitis tetanus *tb kinase cough polio diphtheria malaria disease tuberculosis pertussis* anthrax
thiamin niacin *riboflavin fibre protein iron calcium*
oratorio cantata *concert baroque opera aria motet play*
morphine methadone *chloroform heroin caffeine length phosphate cocaine lsd* librium metabolite
hypnotherapy autosuggestion *psychotherapy exercise meditation therapy counselling analysis*
stepsister stepbrother *friend father sister stepmother brother*
cosine tangent *area sine torsion factor*

Markov Clustering: A very intuitive graph clustering algorithm is *Markov Clustering*[4] developed by van Dongen (2000). Markov Clustering (MCL) partitions a graph via simulation of random walks. The idea is that random walks on a graph are likely to get stuck within dense subgraphs rather than shuttle between dense subgraphs via sparse connections.

MCL computes a hard clustering. The nodes in the graph are divided into non-overlapping clusters. Thus, nodes between dense regions will appear in a single cluster only, although they are attracted by different communities. Inspired by Schütze's method (Schütze, 1998) we next replace clustering of word *strings* by clustering of word *contexts*.

4.2 Clustering the *link graph*

We consider pairs of words which we linked earlier, as word contexts. For example, *organ* occurs in contexts (*organ*, *piano*), (*organ*, *harpsichord*), (*organ*, *tissue*) and (*organ*, *muscle*). In contrast to the semantic "fuzziness" of *organ*, each of its contexts has a sharp-cut meaning and refers to exactly one of the senses of *organ*. By clustering word contexts as opposed to clustering the words themselves, a word's different meanings can be distributed across different clusters which are then interpreted as word senses. For example, we can assign (*organ*, *piano*) and (*organ*, *harpsichord*) to one context cluster, and

[3] Variation of the curvature threshold leads to clusterings of different granularity.

[4] `http://micans.org/mcl/`

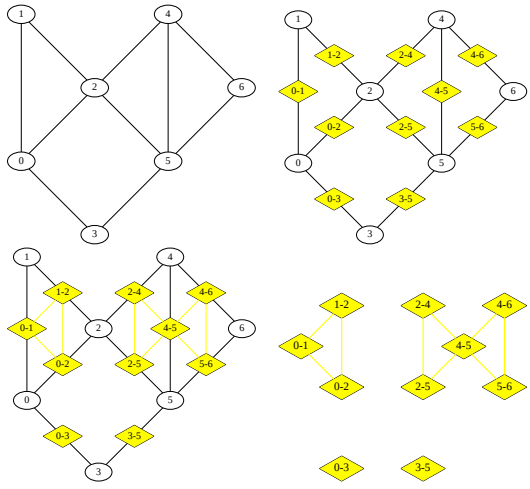

Figure 5: From G to G'. The original graph, new nodes n_l, new links, the graph G'.

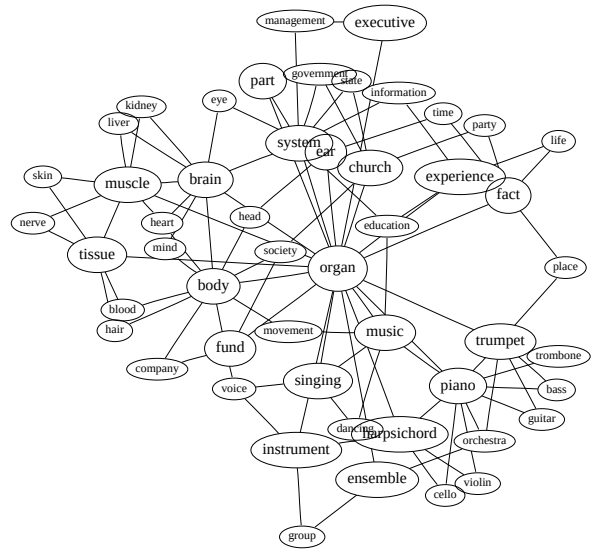

Figure 6: Local word graph around *organ* based on the original graph G. Unrelated areas of meaning (*body parts*, *musical instruments*, *administrative unit*) are connected to one another.

(*organ*, *tissue*) and (*organ*, *muscle*) to another different context cluster.

In the setting of Sect. 2, *words* correspond to *nodes* in the word graph and *word contexts* coincide with the graph's *edges* (with each edge being a context of the two nodes it joins). We now consider *edges* as the fundamental nodes of the *link graph* G', and define the edges of G' as follows: We construct the word graph's associated *link graph*, G', by (see Fig. 5):

1. Introducing a node n_l for each link l in the original graph G.
2. Connecting any two nodes n_{l_1} and n_{l_2} in G' if l_1 and l_2 co-occurred in a triangle in G.

The two component words u and v of a context $l = (u, v)$ disambiguate each other, e.g. in the (*organ*, *harpsichord*) context, both *organ* and *harpsichord* are *instruments*, since this is the intersection of all the possible meanings of *organ* and all the possible meanings of *harpsichord*. The nodes n_l introduced in step 1 therefore have a much narrower meaning than the nodes in G.

The links of a triangle in G constitute mutually overlapping word contexts. We therefore expect the links in such a context triangle to have the same "topic", and the nodes at the corners of the triangle to have the same meaning. This means, step 2 connects two nodes n_{l_1} and n_{l_2} if the corresponding contexts l_1 and l_2 are semantically similar.

Figure 6 shows the local word graph around *organ*. Its associated link graph is illustrated in Fig. 7 (only the connected components containing *organ* and consisting of more than one node are displayed). Note that in the link graph, neighbors which correspond to different senses of *organ* are no longer linked. Transition to the link graph elegantly sliced the initially fuzzy graph into semantically consistent pieces.

Instead of clustering words by partitioning the original graph G, we cluster word contexts by partitioning G's associated link graph G'. The nodes n_l in G' are built with contextual information, and thus typically have a clear-cut meaning. With little (if any) ambiguity left in the link graph, a hard clustering algorithm, such as MCL, is fit for dividing the contexts into (non-overlapping) similarity classes. In detail, our algorithm is:

1. Start with the original graph.
2. Construct the associated link graph G'.
3. Apply Markov Clustering to G'.
4. Merge clusters whose overlap in information exceeds a certain threshold.

The clustering resulting from step 3 is a bit too fine-grained. Several of the context clusters describe the same "topic". We merge these multiple clusters via another application of MCL, this time applied to a graph of context clusters which are linked if their shared information content (the negative logarithm of the probability of the words they have in common) exceeds 50% of the information contained in the smaller of the two clusters. Step 4 reduced the $12,786$ clusters resulting from step 3 to a total of $5,849$ clusters.

5 Conclusions

We have found empirical methods which are capable of recognizing very coherent classes of words,

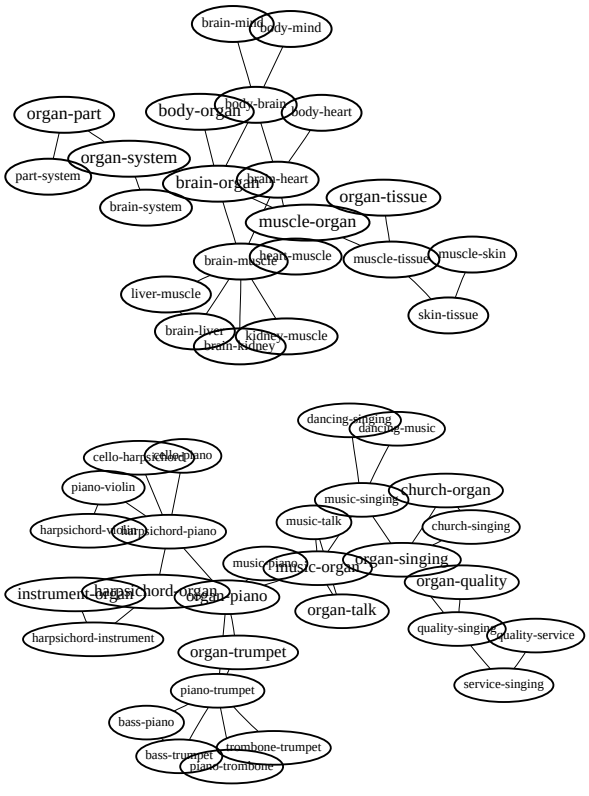

Figure 7: *Organ*'s associated link graph. Contexts belonging to the *body part* and *musical instrument* meaning are neatly separated.

recognizing ambiguity, and automatically splitting a word up into its senses. We have done this with simple regular expressions and combinatoric measures that are easy to understand and implement.

On one hand, curvature has turned out to be a quantity particularly suited for measuring the degree of ambiguity of words, as shown by Fig. 3 and Tab. 1; on the other hand, the clustering scheme based on curvature does especially well for words which are unambiguous. We thus expect curvature clustering to do well in recognizing the meanings of words unknown to WordNet. In contrast, link clustering is particularly suited for splitting ambiguous words into their different meanings. It confers importance to the contextual implication of two words rather than to the word itself. That appears to us to be an important step towards the assignment of meaning to words by contextual association alone.

Both the curvature method and the link clustering determine the number of word senses purely empirically. All we control is how tightly clustered a sense should be. This is a great support to a user who wants to decide what level of lumping or splitting is appropriate for a whole domain, rather than on a word-by-word basis.

In summary, we have shown that graphs can be learned directly from free text and used for ambiguity recognition and lexical acquisition. We introduced two new techniques, *graph curvature* and *link clustering*, combinatoric methods for analyzing the geometry and topology of graphs that can improve the automatized assignment of word meaning.

References

G. Chartrand. 1985. *Introductory Graph Theory*. Dover.

B. Dorow and D. Widdows. 2003. Discovering corpus-specific word-senses. In *Proceedings of EACL, Conference Companion*, pages 79–82, Budapest, Hungary, April.

J. Eckmann and E. Moses. 2002. Curvature of co-links uncovers hidden thematic layers in the world-wide web. In *Proceedings of the Natl. Acad. Sci. USA*, volume 99, pages 5825–5829.

M. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *COLING*, Nantes, France.

P. Pantel and D. Lin. 2002. Discovering word senses from text. In *Proceedings of ACM SIGKDD 2002*, Edmonton, Canada.

F. Pereira, N. Tishby, and L. Lee. 1993. Distributional clustering of english words. In *Proceedings of ACL*, pages 183–190, Columbus, Ohio.

E. Riloff and J. Shepherd. 1997. A corpus-based approach for building semantic lexicons. In *Proceedings of the Second Conference on Empirical Methods in NLP*, pages 117–124. ACL, Somerset, New Jersey.

B. Roark and E. Charniak. 1998. Noun-phrase co-occurence statistics for semi-automatic semantic lexicon construction. In *COLING-ACL*, pages 1110–1116.

H. Schütze. 1998. Automatic word sense discrimination. *Computational Linguistics*, 24(1):97–124.

Claude Shannon. 1948. A mathematical theory of communication. *Bell system technical journal*, 27:379–423, 623–656.

M. Sigman and G. Cecchi. 2002. The global organization of the wordnet lexicon. In *Proceedings of the Natl. Acad. Sci. USA*, volume 99, pages 1742–1747, February.

S. van Dongen. 2000. *Graph Clustering by Flow Simulation*. Ph.D. thesis, University of Utrecht, May.

D. Watts and S. Strogatz. 1998. Collective dynamics of 'small-world' networks. *Nature*, 393:440–442.

D. Widdows and B. Dorow. 2002. A graph model for unsupervised lexical acquisition. In *Proceedings of Coling*, pages 1093–1099, Taipei, Taiwan, August.

Acknowledgements

This research was supported by the National Science Foundation, the Deutsche Forschungsgemeinschaft, the Fonds National Suisse, the Einstein Minerva Center for Theoretical Physics and the Minerva Center for Nonlinear Dynamics.