Discovering Synonyms and Other Related Words

Krister LINDÉN and Jussi PIITULAINEN
Helsinki University, Department of General Linguistics,
P.O.Box 9 (Siltavuorenpenger 20 A),
FIN-00014 University of Helsinki,
Finland,
Krister.Linden@helsinki.fi, Jussi.Piitulainen@helsinki.fi

Abstract

Discovering synonyms and other related words among the words in a document collection can be seen as a clustering problem, where we expect the words in a cluster to be closely related to one another. The intuition is that words occurring in similar contexts tend to convey similar meaning.

We introduce a way to use translation dictionaries for several languages to evaluate the rate of synonymy found in the word clusters. We also apply the information radius to calculating similarities between words using a full dependency syntactic feature space, and introduce a method for similarity recalculation during clustering as a fast approximation of the high-dimensional feature space. Finally, we show that 69-79 % of the words in the clusters we discover are useful for thesaurus construction.

1 Introduction

Finding related words among the words in a document collection can be seen as a clustering problem, where we expect the words in a cluster to be closely related to the same sense or to be distributional substitutes or proxies for one another. A number of language-technology tasks can benefit from such word clusters, e.g. document classification applications, language modelling, resolving prepositional phrase attachment, conjunction scope identification, word sense disambiguation, word sense separation, automatic thesaurus generation, information retrieval, anaphor resolution, text simplification, topic identification, spelling correction (Weeds, 2003).

At present, synonyms and other related words are available in manually constructed ontologies, such as synonym dictionaries, thesauri, translation dictionaries and terminologies. Manually constructing ontologies is time-consuming even for a single domain. On the world-wide web there are documents on many topics in different languages that could benefit from having an ontology. For many of them some degree of automation is eventually needed.

Humans often infer the meaning of an unknown word from its context. Lets look at a less well-known word like *blopping*. We look it up on the Web. Some of the hits are: *Blopping through some of my faves*, i.e. leafing through favourite web links, *A blop module emits strange electronic blopping noises*, i.e. an electronic sound, *The volcano looked like something off the cover of a Tolkien novel - perfectly conical, billowing smoke and blopping out chunks of bright orange lava*, i.e. spluttering liquid. At first we find all of them different and perhaps equally important. When looking at further links, we get an intuition that the first instance is perhaps a spurious creative metonym, whereas the two others can be regarded as more or less conventional and represent two distinct senses of *blopping*. However, the meaning of all three seems to be related to a sound, which is either clicking or spluttering in nature.

The intuition is that words occurring in the same or similar contexts tend to convey similar meaning. This is known as the Distributional Hypothesis (Harris, 1968). There are many approaches to computing semantic similarity between words based on their distribution in a corpus. For a general overview of similarity measures, see (Manning and Schütze, 1999), and for some recent and extensive overviews and evaluations of similarity measures for i.a. automatic thesaurus construction, see (Weeds, 2003; Curran, 2003; Lee, 2001; Dagan et al., 1999). They show that the information radius and the α-skew distance are among the best for finding distributional proxies for words.

If we assume that a word w is represented as a sum of its contexts and that we can calculate the similarities between such word representations, we get a list L_w of words with quantifications of how similar they are to w. Each similarity

list L_w contains a mix of words related to the senses of the word w.

If we wish to identify groups of synonyms and other related words in a list of similarity-rated words, we need to find clusters of similar words that are more similar to one another than they are to other words. For a review of general clustering algorithms, see (Jain et al., 1999) and for a recent evaluation of clustering algorithms for finding word categories, see (Pantel, 2003). (Pantel, 2003) shows that among the standard algorithms the average-link and the k-means clustering perform the best when trying to discover meaningful word groups.

In order to evaluate the quality of the discovered clusters three methods can be used, i.e. measuring the internal coherence of clusters, embedding the clusters in an application, or evaluating against a manually generated answer key. The first method is generally used by the clustering algorithms themselves. The second method is especially relevant for applications that can deal with noisy clusters and avoids the need to generate answer keys specific to the word clustering task. The third method requires a gold standard such as WordNet or some other ontological resource. For an overview of evaluation methodologies for word clustering, see (Weeds, 2003; Curran, 2003; Pantel, 2003).

The contribution of this article is four-fold. The first contribution is to *apply the information radius in a full dependency syntactic feature space* when calculating the similarities between words. Previously, only a restricted set of dependency relations has been applied. The second contribution is a *similarity recalculation during clustering*, which we introduce as a fast approximation of high-dimensional feature space and study its effect on some standard clustering algorithms. The third contribution is a simple but efficient way to *evaluate the synonym content of clusters by using translation dictionaries for several languages*. Finally we show that *69-79 % of the words in the discovered clusters are useful* for thesaurus construction.

The rest of this article is organized as follows. Section 2 presents the corpus data and the the feature extraction. Section 3 introduces the discovery methodology. Section 4 presents the evaluation methodology. In Section 5 we present the experiments and evaluate the results and their significance. Sections 6 and 7 contain the discussion and conclusion, respectively.

2 Corpus Data

Our corpus consists of nouns in a sentence context. We used all the nouns (in base form) that occurred more than 100 times (in any inflected form) in a corpus of Finnish newspaper text. The corpus contained 245 000 documents totaling 48 million words of the Finnish newspaper *Helsingin sanomat* from 1995–1997. Excluding TV and radio listings, there were 196 000 documents with 42 million words. As corpus data we selected all the 17 835 nouns occurring more than 100 times comprising 14 million words of the corpus.

3 Methodology

First we present the types of features we have extracted from the corpus. Then we briefly describe the similarity measure which we use in order to calculate the similarity between the nouns in the corpus data. We also introduce a method for creating derived similarity information in a low-dimensional space. Finally we present the clustering algorithms which we apply to the similarity information.

3.1 Feature extraction

The present experiments aim at discovering the nouns that are most similar in meaning to a given noun. The assumption is that words occurring in similar syntactic contexts belong to the same semantic categories (Harris, 1968). In order to determine the similarity of the syntactic contexts, we represent a word w as a probability distribution over a set of features a occurring in the context of w: $P(a|w)$. The context features a are the major class words w' (nouns, adjectives and verbs) with direct dependency links to the word w. The context feature is the word w' in base form labeled with the dependency relation r. For example, the noun might occur as an object of a verb and with an adjective modifier; both the verb and the adjective including their dependency relations are context features.

We used Connexor's dependency parser FDG for Finnish (Connexor, 2002) for parsing the corpus. A sample of the parser output is shown in Table 1. Tokens of each sentence are numbered starting from zero, each token is on its own line, the token number first, the actual word form second and the base form in the third field. The fourth field links dependent tokens to their heads using a grammatical label and the

#	Token	Base form	Dependency	Morphosyntax	Gloss
0					
1	Toisessa	toinen		&NH PRON SG INE	In the other
2	esitetään	esittää	main:>0	&+MV V PASS IND PRES	they showed
3	videoita	video	obj:>2	&NH N PL PTV	videos
4	ja	ja	cc:>3	&CC CC	and
5	filmejä	filmi	cc:>3	&NH N PL PTV	films
6	.	.			
7	<s>	<s>	>6		

Table 1: Sample output from the FDG parser for Finnish (with an English gloss added).

number of the head token. The fifth field contains morphosyntactic information.

Two tokens, 3 and 5, are labeled as nouns `N`. The noun `video` is a direct object to the verb `esittää`, and the noun `filmi` is coordinated with `video`, so `video` gets two feature occurrences from this sentence:

```
esittää-obj
cc-filmi.
```

Also, `filmi` gets

```
video-cc
```

as a feature occurrence. The pronoun `toinen` is not a potential feature because of its word class and because it is not linked. The coordinating conjunction `ja` is not a potential feature because of its word class.

The parsed corpus contained a total of 18 516 609 unambiguous noun occurrences, 69 314 noun/verb ambiguities, 39 104 noun/adjective ambiguities, 20 847 noun/adverb ambiguities and 11 739 noun/numeral ambiguities, i.e. the amount of remaining ambiguities was less than 0.8 %. When its analyses were underspecified with more than one morphological analysis remaining, we took the relatively small risk ($p < 0.008$) of committing to a noun analysis.

As a straightforward weighting of the context features of a word, we used the number of occurrences with all the instances of the word. In our choice of similarity formula, the representation of a word w must be a probability distribution. This is formally just a matter of normalizing the weights of the features. Thus, a word w is represented as $w : a \mapsto P(a|w)$, i.e. the conditional probability distribution of all features a given the word w, such that $\sum_a P(a|w) = 1$.

Extracting features only from direct dependency relations produces few feature occurrences for each instance of a noun. This keeps the number of distinct features tolerable for all but the most frequent words, and still retains the most promising co-occurring words. As we use only linear frequency weighting, very frequent features tend to get more weight than they should. Additionally, many rare features could have been dropped without much loss of information.

3.2 Similarity calculations

In (Weeds, 2003; Lee, 2001), i.a. the information radius is applied to finding words that can be used as proxies or substitutes for one another. Their tests show that the information radius is among the best for finding such words. Here we briefly recapitulate the details of the similarity estimate, which is rather an estimate of dissimilarity.

Two words are distributionally similar to the extent that they co-occur with the same words, i.e., to the extent that they share features. We define the dissimilarity of two words, p and q, as

$$J(p, q) = (D(p\|m) + D(q\|m))/2, \qquad (1)$$

where $D(p\|m) = \sum_a p(a)(\log_2 p(a) - \log_2 m(a))$ and $m(a) = (p(a) + q(a))/2$ for any feature a. This is the symmetrically weighted case of the Jensen–Shannon divergence (Lin, 1991), also known as the information radius or the mean divergence to the mean (Dagan et al., 1999). For complete identity, $J(p, p) = 0$. For completely disjoint feature sets, $J(p, q) = 1$. The formula is symmetric but does not satisfy the triangle inequality. For speed the estimate may be calculated from the shared features alone (Lee, 1999).

After calculating all the pairwise estimates, we retained lists of the 100 most similar nouns for each of the nouns in the corpus data. No other data is used in the similarity calculations.

3.3 Low-dimensional similarity measures

Performing all the calculations in high-dimensional feature space is time-consuming. Here we introduce a method that can be used as an approximation in low-dimensional feature space based on the initial similarity estimates.

Assume that we have lists of the words that are distributionally most similar to a given word w. Each list L_w contains 100 words with an estimate of their similarity to w. The words in L_w represent a mix of the different meanings of the word w. We create a similarity matrix dis_w for these words such that $dis_w(p,q) = J(p,q)$, where $p, q \in L_w$. The similarity matrix dis_w is a symmetric matrix of the dimensions 101 by 101, as we also include the word w in the matrix.

A vector $p_w = dis_w(p,.)$ in the similarity matrix dis_w is regarded as a projection of the word p from a high dimensional feature space onto a 101-dimensional space, i.e. p is projected onto the 101 most important dimensions of w. The new matrix is not orthogonal, so we apply single-value decomposition (SVD) $dis_w = T\,S\,D$ and use T to rotate the matrix so that the first axis runs along the direction of the largest variation among the word similarity estimates, the second dimension runs along the direction of the second largest variation and so forth. After this rotation we can cluster the new vectors $p_{w,T} = T^t\,p_w$ as low-dimensional representatives of the original high-dimensional feature space. Often SVD is used for dimensionality reduction, but here we use its left singular vectors only for rotating the matrix in order to achieve noise reduction during clustering.

In the new low-dimensional vector representation $p_{w,T}$ we apply the cosine distance $cosd(p_{w,T}, q_{w,T}) = 1 - \cos(p_{w,T}, q_{w,T})$ in order to calculate the similarity between words. As a comparison we also tried the squared Euclidean distance $eucld(p_{w,T}, q_{w,T}) = \|p_{w,T} - q_{w,T}\|^2$ between words in the low-dimensional space. We first normalize the vectors to unit length, which effectively makes the squared Euclidean distance the same as two times the cosine distance: $\|A-B\|^2 = \|A\|^2 + \|B\|^2 - 2\|A\|\,\|B\|\cos(A,B)$, and when $\|A\| = 1$ and $\|B\| = 1$, we have $\|A-B\|^2 = 2\,(1-\cos(A,B))$.

3.4 Clustering

When we wish to discover the potential senses of w by clustering, we are currently only interested in the 100 words in L_w with a similarity estimate for w. The other words are deemed to be too dissimilar to w to be relevant.

We cluster the words related to w with standard algorithms such as complete-link and average-link clustering (Manning and Schütze, 1999). Complete-link and average-link are hierarchical clustering methods. We compare them with flat clustering methods like k-means and self-organizing maps (SOM) (Kohonen, 1997). In k-means the clusters have no ordering. The potential benefit of using SOM with a two-dimensional display compared to k-means is that related data samples get assigned into nearby clusters as the SOM converges forming cluster areas with related content.

We use the MATLAB implementation (The MathWorks, Inc., 2002) of the algorithms. We use both the original similarity measures in dis_w and the distance measures *cosd* and *eucld*, which we defined on the low-dimensional space. In order to use methods like k-means and SOM, we need to be able to calculate the similarity between cluster centroids and words to be clustered each time a centroid is updated. We do this in the low-dimensional space $p_{w,T}$ using *cosd* and *eucld*.

For SOM, the MATLAB implementation supported only the squared Euclidean distance. It should be noted that the centroids are not necessarily of unit length, so the squared Euclidean distance is different from the cosine distance between the samples and the centroids, when the centroids are based on more than one sample.

Our clustering setup currently produces hard clusters, where each word w in L_w belong to one cluster, as opposed to soft clustering, where a word may belong to several clusters. We call the cluster containing the word w itself the key cluster.

4 Evaluation methodology

In order to evaluate the quality of the clusters we need a gold standard. English and a number of other languages have resources such as WordNet (Fellbaum, 1998; Vossen, 2001). For Finnish there is no WordNet and there are no large on-line synonym dictionaries available. In fact, our experiment can be seen as a feasibility study for automatically extracting information that could be used for building a WordNet for Finnish. The synsets of WordNet contain synonyms, so we can evaluate the feasibility of the clusters for WordNet development by rating the amount of synonyms and related words in the

Language	Target word	Back translation
English	deficit	*vaje, vajaus, alijäämä*; *tilivajaus*
	shortfall	*vaje, alijäämä*
German	Defizit	*vajaus, vaje, alijäämä*; kassavajaus, tappio; *tilivajaus*; puutos, puute
	Unterbilanz	*alijäämä, vajaus, vaje*, kauppavaje
	Fehlbetrag	*vajaus, alijäämä*, tappio, virhemaksu
French	déficit	*alijäämä*, miinus, *tilivajaus*; *vajaus, vaje*; tappio

Table 2: Translations of the Finnish source word *alijäämä* into English, German and French with the back translations into Finnish. The shared back translations *vaje, vajaus, alijäämä, tilivajaus* are highlighted.

discovered clusters.

We note that when translating a word from the source language the meaning of the word is rendered in a target language. Such meaning preserving relations are available in translation dictionaries. If we translate into the target language and back we end up i.a. with the synonyms of the original source language word. In addition, we may also get some spurious words that are related to other meanings of the target language words. If we assume that the other words represent spurious cases of polysemy or homonymy in the target language, we can reduce the impact of these spurious words by considering several target languages and for each source word we use only the back-translated source words that are common to all the target languages. We call such a group of words a source word synonym set. For an example, see Table 2.

In addition to the mechanical rating of the synonym content we also manually classified the words of some cluster samples into synonymy, antonymy, hyperonymy, hyponymy, complementarity and other relations.

4.1 Evaluation data

In order to evaluate the clusters we picked a random sample of 1759 nouns from the corpus data, which represented approximately 10 % of the words we had clustered. For these words we extracted the translations in the Finnish-English, Finnish-German and Finnish-French MOT dictionaries (Kielikone, 2004) available in electronic form. We then translated each target language word back into Finnish using the same resources. The dictionaries are based on extensive hand-made dictionaries. The choice of words may be slightly different in each of them, which means that the words in common for all the dictionaries after the back translation tend to be only the core synonyms.

For evaluation purposes it would be unfair to demand that the clustering generate words into the clusters that are not in the corpus data, so we also removed those back translations from the source word synonym sets. Finally, only synonym sets that had more than one word remaining were interesting, i.e. they contained more than the original source word. There were 453 of the 1759 test words that met the qualifications. The average number of synonyms or back translations for these test words was 3.53 including the source word itself.

For manual classification we used a sample of 50 key clusters from the whole set of clusters and an additional sample of 50 key clusters from the words qualifying for the mechanical evaluation.

4.2 Evaluation method

The mechanical evaluation was performed by picking the key cluster produced by a clustering algorithm for each of the test words. The key cluster was the cluster which contained the original source word. The evaluation was a simple overlap calculation with the gold standard generated from the translation dictionaries. By counting the number of cluster words in a source word synonym set and dividing by the synonym set size, we get the recall R. By counting the number of of source word synonyms in a cluster and dividing by the cluster size, we get the precision P.

The manual evaluation was performed independently by the two authors and an external linguist. We then discussed the result in order to arrive at a common view.

5 Testing

First we did some initial experimenting with a preliminary test sample in order to tune the parameters. We then clustered the corpus data and evaluated the clusters against the gold standard, which gave an estimate of the synonym content of the clusters. In addition, we performed a manual evaluation of the result of the

Clustering method	Information radius		Cosine distance		Euclidean distance	
	R	P	R	P	R	P
Average link	47	42	43	41	43	41
Complete link	47	40	42	39	42	38
K-means	-	-	43	36	42	36
SOM	-	-	-	-	41	35

Table 3: Cluster synonym content as average recall (R) and precision (P) in per cent (%) with a standard deviation of 2 % using different clustering methods and similarity measures.

Clustering method	Cosine distance w rotated feature space		Cosine distance w/o rotation	
	R	P	R	P
Average link	43	41	42	32
Complete link	42	39	41	33

Table 4: Cluster synonym content as average recall (R) and precision (P) in per cent (%) with a standard deviation of 2 % using a denoised and a noisy low-dimensional feature space.

best clustering algorithm.

5.1 Parameter selection

We clustered the words in L_w with the complete-link and average-link clustering algorithms using the dis_w similarity information. The algorithms form hierarchical cluster trees which need to be split into clusters at some level. The inconsistency coefficient c characterizes each link in a cluster tree by comparing its length with the average size of other links at the same level of the hierarchy. The higher the value of this coefficient, the less similar the objects connected by the link (The MathWorks, Inc., 2002). We selected the inconsistency coefficient $c = 1$ by testing on a separate initial test set different from the final evaluation data.

Using the cosine distance $cosd(p_{w,T}, q_{w,T})$ as a similarity measure on the projected and rotated representation of the words we clustered with the above mentioned standard clustering algorithms as well as with the k-means algorithm. Using the euclidean distance $eucld(p_{w,T}, q_{w,T})$ we also produced self-organizing maps (SOM). For k-means and SOM an initial number of clusters need to be selected. We selected 35 clusters as this was close to the average of what the other algorithms produced, which we were comparing with. For k-means we used the best out of 10 iterations and for SOM we trained a 5×7 hexagonal gridtop for 10 epochs. We also tried a considerably longer training period for SOM but noticed only an insignificant improvement on the cluster precision.

We also tried a number of other algorithms in the MATLAB package, but they typically produced a result either containing only the word itself or clusters containing more than one fifth of the words in the key cluster. We deemed such clustering results a failure on our data without need for formal evaluation.

5.2 Experiments

After evaluating against the translation dictionary gold standard, the result of the experiment with complete-link, average-link, k-means and SOM clustering using different similarity measures is shown in Table 3. *The best recall with the best precision was achieved with the average-link clustering using the information radius on the original feature space* with 47 ± 2 % recall and and 42 ± 2 % precision. This produced clusters with an average size of 6.05 words.

The difference between complete-link and average-link clustering is not statistically significant even if the average-link is slightly better. *The recall is statistically significantly better in the original feature space than in the low-dimensional space at the risk level $p = 0.05$, whereas the precision remains roughly the same.* The average-link and complete-link clustering have a statistically significantly better precision than k-means and SOM, respectively, at the risk level $p < 0.05$. We can also see that there is hardly any difference in practice between the Euclidean distance on normalized word vectors and the cosine distance despite the fact that the centroids were not normalized when using the squared Euclidean distance with k-means.

As can be seen from Table 4 the rotation of the low-dimensional feature space using SVD has the effect of increasing precision statistically significantly at the risk level $p < 0.005$, i.e. the

Word	alijäämä/deficit	maatalous/agriculture	tuki/aid
Synonymy	vaje/deficiency vajaus/shortfall		avustus/subsidy apu/help
Antonymy	ylijäämä/surplus		
Complementarity		teollisuus/industry vientiteollisuus/export industry elintarviketeollisuus/food industry	
Hyperonymy		elinkeinoelämä/business talouselämä/economy	
Hyponymy			rahoitus/financing

Table 5: Semantic relations of the cluster content of some sample words (English glosses added)

Content Relations	Dictionary sample	All words sample
Synonymy	52 %	38 %
Antonymy	1 %	1 %
Complementarity	12 %	34 %
Hyperonymy	2 %	4 %
Hyponymy	1 %	3 %
Other	31 %	21 %
Total	100 %	100 %

Table 6: Manual evaluation of the percentage of different semantic relations in the cluster content in two different samples of 50 clusters each.

clusters become less noisy.

We then performed a manual evaluation of the output of the best clustering algorithm. We used one cluster sample from the 453 clusters qualifying for mechanical evaluation and one sample from the whole set of 1753 clusters. The results of the manual evaluation is shown in Table 6. The evaluation shows that 69-79 % of the material in the clusters is relevant for constructing a thesaurus.

The manual evaluation agrees with the mechanical evaluation, when the manual evaluation found a synonym content of 52 %, compared to the minimum synonym content of 42 % found by the mechanical evaluation. This means that the clusters actually contain a few more synonyms than those conservatively agreed on by the three translation dictionaries.

If we evaluate the sample of key clusters drawn from all the words in the test sample, we get a synonym content of 38 %. This figure is rather low, but can be explained by the fact that many of the words were compound nouns that had no synonyms, which is why the translation dictionaries either did not have them listed or contained no additional source word synonyms for them.

In Table 5, we see a few sample clusters whose words we rated during manual evaluation.

6 Discussion

The feature selection and the feature weighting radically influences the outcome of the results of any machine learning task. This has been noted in several evaluations of supervised machine learning algorithms (Voorhees et al., 1995; Yarowsky and Florian, 2002; Lindén, 2003). During clustering, i.e. unsupervised learning, the features extracted from the corpus are the only information guiding the machine learning in addition to the clustering principle, which makes successful feature extraction, good feature weighting and accurate similarity measurements crucial for the success of the clustering. The clustering algorithms only exploit and preserve the information provided by the features and the similarity measure.

In (Weeds, 2003; Lee, 2001; Dagan et al., 1999), the information radius is applied to find words that can be used as distributional proxies for one another. They extract features only from verb relations whereas we use the full range of dependency syntactic relations. One intention of this study was to evaluate whether the selected corpus and the features extracted provide a basis for forming linguistically meaningful clusters that are useful in thesaurus construction. The result showed that 69-79 % of the words found in the key clusters are useful, which is very encouraging. It turned out that the chosen features as such were useful, even if the over-all result probably could benefit from a more nuanced feature weighting scheme. We do not yet fully understand how the initial feature weighting affects the outcome of the clustering. Perhaps there are features that would contribute to a more fine-grained clustering if properly weighted.

Next we intend to identify more than a single key cluster for each word, which poses addi-

tional challenges for the evaluation. We also aim at evaluating the generated clusters in an information retrieval setting in order to see if they improve performance despite the fact that they contain more than synonyms. This would also shed some light on exactly how much synonymy we need to aim at in a practical application.

7 Conclusion

We have demonstrated that it is feasible to calculate similarities between words using a full dependency syntactic feature space. We have also introduced similarity recalculation during clustering as a fast approximation of the high-dimensional feature space. In addition we introduced a way to use translation dictionaries for evaluating the rate of synonymy found in the word clusters, which is useful for languages that do not yet have publicly available thesaurus resources like WordNet. Finally we have shown that 69-79 % of the words in the discovered clusters are useful for thesaurus construction.

Acknowledgements

The second author made the Jensen-Shannon similarity lists and the corpus processing described in Sections 2, 3.1 and 3.2, and the first author did the rest. We are grateful to Lauri Carlson, Kimmo Koskenniemi and Mathias Creutz for helpful comments on the manuscript and to Juhani Jokinen for manually evaluating the test samples.

References

Connexor. 2002. Machinese phrase tagger. [http://www.connexor.com/].

James Richard Curran. 2003. *From Distributional to Semantic Similarity*. Ph.D. thesis, Institute for Communicating and Collaborative Systems, School of Informatics, University of Edinburgh.

Ido Dagan, Lillian Lee, and Fernando C. N. Pereira. 1999. Similarity-based models of word co-occurrence probabilities. *Machine Learning*, 34(1–3):43–69.

Christiane Fellbaum. 1998. *WordNet, An Electronic Lexical Database*. The MIT Press.

Zellig Harris. 1968. Mathematical structures of language. *Interscience Tracts in Pure and Applied Mathematics*, 21(ix):230 pp.

A. K. Jain, M. N. Murty, and P. J. Flynn. 1999. Data clustering: a review. *ACM Computing Surveys*, 31(3):264–323.

Kielikone. 2004. Dictionary service mot - dictionaries and terminologies. [http://www.kielikone.fi/en/].

Teuvo Kohonen. 1997. *Self-Organizing Maps (Second Edition)*, volume 30 of *Springer Series in Information Sciences*. Springer, Berlin.

Lillian Lee. 1999. Measures of distributional similarity. In *37th Annual Meeting of the Association for Computational Linguistics*, pages 25–32.

Lillian Lee. 2001. On the effectiveness of the skew divergence for statistical language analysis. In *Artificial Intelligence and Statistics 2001*, pages 65–72.

Jianhua Lin. 1991. Divergence measures based on the Shannon entropy. *IEEE Transactions on Information Theory*, 37(1):145–151, January.

Krister Lindén. 2003. Word sense disambiguation with thessom. In *Proceedings of the WSOM'03 – Intelligent Systems and Innovational Computing*, Kitakuyshu, Japan, September.

Christopher D. Manning and Hinrich Schütze. 1999. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, Massachusetts.

Patrick André Pantel. 2003. *Clustering by Committee*. Ph.D. thesis, University of Alberta, Edmonton, Alberta, Canada.

The MathWorks, Inc. 2002. Matlab with statistics toolbox and neural network toolbox. [http://www.mathworks.com/], June 18. Version 6.5.0.180913a Release 13.

Ellen M. Voorhees, Claudia Leacock, and Geoffrey Towell, 1995. *Computational Learning Theory and Natural Language Learning Systems 3: Selecting Good Models*, chapter Learning context to disambiguate word senses, pages 279–305. MIT Press, Cambridge.

Piek Vossen. 2001. Eurowordnet. [http://www.hum.uva.nl/∼ewn/].

Julie Elisabeth Weeds. 2003. *Measures and Applications of Lexical Distributional Similarity*. Ph.D. thesis, University of Sussex, September.

David Yarowsky and Radu Florian. 2002. Evaluating sense disambiguation across diverse parameter spaces. *Natural Language Engineering*, 8(4):293–310, December.