

DISCO: A Multilingual Database of Distributionally Similar Words

Peter Kolb

Abstract. This paper¹ presents DISCO, a tool for retrieving the distributional similarity between two given words, and for retrieving the distributionally most similar words for a given word. Pre-computed word spaces are freely available for a number of languages including English, German, French and Italian, so DISCO can be used off the shelf. The tool is implemented in Java, provides a Java API, and can also be called from the command line. The performance of DISCO is evaluated by measuring the correlation with WordNet-based semantic similarities and with human relatedness judgements. The evaluations show that DISCO has a higher correlation with semantic similarities derived from WordNet than latent semantic analysis (LSA) and the web-based PMI-IR.

1 Introduction

A growing number of applications in natural language processing rely on knowledge about the semantic similarity between words. These similarities are used for example in ontology learning (Cimiano et al. (2005)), information retrieval (Müller et al. (2007)), and word sense disambiguation (Patwardhan et al. (2007)).

One has to differentiate between semantic similarity and semantic relatedness (Budanitsky and Hirst (2006)). The first is a narrower concept that holds between lexical items having a similar meaning, like *palm* and *tree*. The broader concept semantic relatedness holds between lexical items that are connected by lexical relations like meronymy (e.g. *palm* – *leaf*) or belong to the same semantic field (e.g. *palm* – *coconut*). Thus, semantic similarity is a special case of semantic relatedness.

Unfortunately, measures of semantic similarity and relatedness rely on hand-crafted lexical resources like WordNet, which are not available for many languages and have limited coverage, particularly in specialized domains. Therefore, Kilgariff (2003) and others have argued for using distributional similarity as a proxy for semantic similarity. Distributional similarity is based on the assumption that words with similar meaning occur in similar contexts (Miller (1969)). Several successful methods to compute the distributional similarity of words from text corpora have been proposed,

1. Published in A. Storrer, A. Geyken, A. Siebert and K.-M. Würzner (Ed.), *KONVENS 2008 - Ergänzungsband: Textressourcen und lexikalisches Wissen*, Berlin 2008.

including Landauer and Dumais (1997), Grefenstette (1994), Dorow and Widdows (2003), and Rapp (2004).

Budanitsky and Hirst (2006) emphasize the difference between semantic and distributional similarity. Methods that measure the similarity of the distributional behaviour of words do not take into account the different senses a word has, and therefore mix up the similar words for all the word senses. While semantic similarity is a relation between concepts, distributional similarity is a relation between words. Weeds and Weir (2005) on the other hand point out that for many applications it is only important to know that some words behave the same way with respect to the problem at hand and not if they mean similar things.

Finally, Mohammad and Hirst (2005) differentiate between distributional relatedness and distributional similarity. Two words are distributionally similar if they have many common co-occurring words in the same syntactic relations. By contrast, distributional measures that use a bag-of-words context capture distributional relatedness. Kilgarriff and Yallop (2000) call these two variants *tight* and *loose* word similarities. As DISCO does not employ syntactic parsers, it should be subsumed under the methods for finding distributional relatedness between words. We believe, however, that the difference is a gradual one and that DISCO lies somewhere in between both ends of the scale, as we will further discuss in the next section. In section three we present DISCO's features and implementation details. Section four evaluates DISCO's performance against human relatedness judgements and similarities produced by other systems. In the last section we summarize our contributions.

2 Method

Our method for computing the distributional similarity between words works as follows. In a preprocessing step, the corpus at hand is tokenized and highly frequent function words are eliminated. Since we want to keep the method independent from language-specific resources, neither part of speech tagging nor lemmatization are performed, and we use a simple context window of size ± 3 words for counting co-occurrences. Our evaluations showed that it is beneficial to take the exact position within the window into account (Rapp (1999)). This can be seen as a crude approximation of syntactic dependency relations. Instead of dependency triples like $\langle \textit{donut}, \text{OBJ-OF}, \textit{eat} \rangle$ we get triples of the form $\langle \textit{donut}, -2, \textit{eat} \rangle$. Consequently, the features that describe a word's distribution are not just words as in a pure bag-of-words approach, but ordered pairs of word and window position. This kind of context leads to tighter similarities than a window without exact position. We therefore claim DISCO to lie somewhere in between distributional relatedness and similarity.

Moving the window over our corpus results in a co-occurrence matrix. Every row of the matrix describes a word, and is also called a first order word vector. The matrix

size is not $v \times f$ as usual (with v being the number of words for which word vectors are built, f being the number of words used as features), but $v \times f \cdot r$ (r is the window size). The next step is to transform the absolute counts in the matrix fields into more meaningful weights. For this feature weighting we found the measure proposed by Lin (1998b), which is based on mutual information, to be optimal:

$$\log \frac{(f(w, r, w') - 0,95)f(*, r, *)}{f(w, r, *)f(*, r, w')} \quad (1.1)$$

where w and w' stand for words and r for a window position (or a dependency relation, respectively), and f is the frequency of occurrence.

To arrive at a word's distributionally similar words the next step is to compare every word vector with all other word vectors. For vector comparison we use Lin's information theoretic measure (Lin (1998a)):

$$lin = \frac{\sum_{(r,w')} (w_m, *r, *w') + (w_n, *r, *w')}{\sum_{(r,w')} (w_m, *, *) + \sum_{(r,w')} (w_n, *, *)} \quad (1.2)$$

As an example of the outcome, the twelve distributionally most similar words for *palm* are listed here:

palms (0.1345) coconut (0.1059) olive (0.0870) pine (0.0823) citrus (0.0745) oak (0.0677) mango (0.0652) cocoa (0.0645) banana (0.0627) bananas (0.0623) trees (0.0570) fingers (0.0560)

Such a list of distributionally similar words can in turn be seen as the *second order* word vector of the given word, containing not only the words which occur together with it, but those that occur in similar contexts. We can now compare two words based on their second order word vectors, too. This use of higher-order co-occurrences is to some extent comparable to what is achieved in LSA by singular value decomposition (Kontostathis and Pottenger (2006)).

3 Features of the Software

DISCO is implemented in Java and consists of a pre-computed database of collocations and distributionally similar words. In essence, for all words the first and second order word vectors are stored. For this, DISCO uses the Lucene² index. If the similarity between two words is queried, DISCO retrieves their word vectors from the

2. <http://lucene.apache.org>

index and computes the similarity according to the measure described above. If the distributionally most similar words for an input word are requested, DISCO directly returns the second order word vector for the input word. The Java class provides the following functionalities:

- 1) Retrieve collocations for a word. This means that the first order word vector is returned.
- 2) Retrieve the distributionally most similar words, i.e. the second order word vector is returned.
- 3) Compute the first order similarity between two input words, based on their collocation sets (DISCO1).
- 4) Compute the second order similarity between two input words, based on their sets of distributionally similar words (this measure is called DISCO2 below).

On a modern PC, DISCO can compute the first order similarity scores for approximately 50 word pairs per second, and the second order similarity scores for about 25 word pairs per second. The biggest DISCO index we have built so far is based on a 700 million token corpus and provides the first and second order similarities for the 480,000 most frequent words. The index contains the equivalent of a sparse matrix with 1,110,629,329 non-zero fields, its size being nearly seven gigabytes.

In contrast to the web services at the Projekt Deutscher Wortschatz³ DISCO delivers not only collocations, but lists of distributionally similar words, and it computes the similarity between two arbitrary words. Also, DISCO's database can be downloaded for easier integration into the user's own applications.

4 Evaluation

4.1 Data

We built a DISCO word space according to the method outlined above. The corpus consisted of 300,000 articles from the English Wikipedia⁴, amounting to some 267 million tokens. We considered all words with a corpus frequency of at least 100, resulting in $v = 226,000$, and used the $f = 101,000$ most frequent words as feature words.

As mentioned above, there exist word spaces for other languages as well. For German, we downloaded the German version of the Wikipedia and extracted all articles that correspond to an article in the English Wikipedia, using Wikipedia's translation links. Then the German word space was built on this subset of approximately

3. <http://wortschatz.uni-leipzig.de>

4. <http://en.wikipedia.org>

300,000 German Wikipedia articles (which only amount to 171 million tokens). The same was done for French and Italian. For English, there also exists a word space that is based on 100,000 articles from the medical database PubMed⁵.

Finkelstein et al. (2001) prepared a list of 353 noun-noun pairs and employed 16 subjects to estimate their semantic relatedness on a scale from 0 to 10. We use this list as our evaluation data. As seven word pairs contained at least one word that was unknown to WordNet, we deleted them from the list, leaving 346 word pairs for testing.

4.2 Description of the other Systems

LSA. Latent semantic analysis (Landauer and Dumais (1997)) is arguably the most popular variant of word space. Its core step is a dimension reduction technique called singular value decomposition (SVD). SVD computes the least mean square error projection of a matrix onto a lower dimensional matrix. It achieves a kind of generalization by combining columns that stand for words with similar meanings. In our experiments we used the LSA implementation accessible at <http://lsa.colorado.edu>.

PMI-IR (pointwise mutual information - information retrieval). Turney (2001) presents a method for computing the similarity between arbitrary words that utilizes the WWW search engine AltaVista⁶ according to the following formula, adapted from pointwise mutual information:

$$PMI-IR(w_1, w_2) = \log\left(\frac{H(w_1 NEAR w_2)}{H(w_1)H(w_2)}\right) \quad (1.3)$$

where $H(w)$ is the number of hits the search engine returns for the query w . The more often two words co-occur near each other on a web page, the higher is their PMI-IR score. We computed the PMI-IR similarity values for the 346 test pairs by querying AltaVista on 4/10/2008.

WordNet::Similarity. WordNet::Similarity (Pedersen et al. (2004)) is a Perl module based on WordNet that has been widely used in a variety of natural language processing tasks. It implements three measures of semantic relatedness (namely Hirst-St. Onge (hso), Lesk (lesk) and vector pairs (vp)) and six measures of semantic similarity (Jiang and Conrath (jcn), Leacock and Chodorow (lch), Lin (lin), path length (path), Resnik (res), and Wu and Palmer (wup)). The latter utilize the *is-a* relations

5. <http://www.ncbi.nlm.nih.gov/pubmed/>

6. <http://www.altavista.com>

		Vector-based	LSA	PMI-IR	DISCO1	DISCO2			
		0.41	0.56	0.63	0.39	0.51			
hso	lesk	vp	jcn	lch	lin	path	res	wup	
0.35	0.21	0.39	0.23	0.35	0.30	0.38	0.36	0.30	

Table 1. Correlation of several systems with the semantic relatedness values assigned by humans.

in WordNet. Since there are only *is-a* relations between nouns and between verbs in WordNet, the similarity measures cannot be applied to adjectives or across part of speech.

4.3 Correlation with Human Judgements of Semantic Relatedness

Our first experiment measures the correlation (according to the Pearson correlation coefficient) of the candidate systems with the averaged semantic relatedness scores assigned to the 346 word pairs by the human subjects. Table 1 shows the results. The first two correlation values in the first row of the table are taken from Finkelstein et al. (2001). Among the systems listed in the first row, DISCO1 shows the lowest correlation with the human judgements, comparable to that of Finkelstein et al.’s vector approach. DISCO2 performs much better, but is still worse than LSA. The best score is achieved by PMI-IR, which is in accordance with other results reported in the literature (Turney (2001)).

The WordNet-based measures (shown in the second row of the table) perform worse, which comes as no surprise for the six measures of similarity, since they are not intended to measure relatedness. But the three measures of relatedness (hso, lesk, and vp) do not perform much better. The best scoring vector pairs measure (vp) only achieves the same score as DISCO1.

We conclude that for computing semantic relatedness higher-order co-occurrences (like that of DISCO2) can substitute for SVD – not fully, but at least to a certain degree.

4.4 Correlation with WordNet::Similarity

We now take the semantic similarity values produced by the six WordNet similarity measures as gold standard and compare the correlation of the other test systems with these similarities. In this task, PMI-IR performs worst (cf. table 2), whereas DISCO1 shows the highest correlation on average. Note that DISCO1 compares words based on their collocation sets, while PMI-IR’s similarities *are* collocations. Obviously, the

	jcn	lch	lin	path	res	wup	avg.
PMI-IR	0.14	0.12	0.06	0.15	0.22	0.11	0.13
LSA	0.16	0.26	0.21	0.29	0.28	0.22	0.24
DISCO1	0.38	0.39	0.33	0.45	0.42	0.33	0.38
DISCO2	0.15	0.40	0.39	0.35	0.44	0.40	0.36

Table 2. Correlation between WordNet-based semantic similarity and four systems based on word distributions.

higher-order co-occurrences of DISCO2 are of no use here. SVD seems to be even worse, since LSA shows a much lower performance than DISCO1 and DISCO2. Our conclusion is that SVD and higher-order co-occurrences increase the performance when computing semantic relatedness, but they (and probably other techniques of dimension reduction and generalization) do not help in computing semantic similarity.

5 Summary

We have presented an off-the-shelf tool for retrieving the distributional similarity between arbitrary words in a number of languages. Two distinctive features of our tool are that it returns the distributionally most similar words for an input word, and that it computes similarity scores based on second order word vectors. Our evaluations have shown that DISCO has a higher correlation with semantic similarities derived from WordNet than LSA and the web-based PMI-IR. It also achieves a higher correlation with semantic relatedness judgements by human subjects than the WordNet-based measures of semantic relatedness.

DISCO and word spaces in several languages are freely available for research purposes at <http://www.linguatools.de/disco.html>.

References

- Budanitsky, A. and G. Hirst (2006). Evaluating WordNet-based Measures of Lexical Semantic Relatedness. *Computational Linguistics* 32(1).
- Cimiano, P., A. Hotho, and S. Staab (2005). Learning Concept Hierarchies from Text Corpora using Formal Concept Analysis. *Journal of Artificial Intelligence Research* 24:305–339.
- Dorow, B. and D. Widdows (2003). Discovering Corpus-specific Word Senses. In *Proceedings of EACL*, 79–82, Budapest.
- Finkelstein, L., E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, and E. Ruppin (2001). Placing search in context: the concept revisited. In *WWW '01: Proceedings of the 10th international conference on World Wide Web*, 406–414, New York, NY, USA: ACM.
- Grefenstette, G. (1994). *Explorations in Automatic Thesaurus Discovery*. Dordrecht: Kluwer.

- Kilgarriff, A. (2003). Thesauruses for Natural Language Processing. In *Proceedings of Natural Language Processing and Knowledge Engineering (NLPKE)*, Beijing.
- Kilgarriff, A. and C. Yallop (2000). What's in a thesaurus? In *Proceedings of the Second Conference on Language Resources and Evaluation*, 1371–1379, Athens.
- Kontostathis, A. and W. M. Pottenger (2006). A framework for understanding latent semantic indexing (LSI) performance. *Information Processing and Management* 42(1):56–73, doi:10.1016/j.ipm.2004.11.007.
- Landauer, T. K. and S. T. Dumais (1997). A Solution to Plato's Problem: The Latent Semantic Analysis Theory of Acquisition, Induction and Representation of Knowledge. *Psychological Review* 104(2):211–240.
- Lin, D. (1998a). Automatic Retrieval and Clustering of Similar Words. In *Proceedings of COLING-ACL 1998*, Montreal.
- Lin, D. (1998b). Extracting Collocations from Text Corpora. In *Workshop on Computational Terminology*, 57–63, Montreal, Kanada.
- Miller, G.A. (1969). The organization of lexical memory: Are word associations sufficient? In G.A. Talland and N.C. Waugh (eds.), *The pathology of memory*, 223–36, New York: Academic Press.
- Mohammad, S. and G. Hirst (2005). Distributional Measures as Proxies for Semantic Relatedness, unpublished.
- Müller, C., I. Gurevych, and M. Mühlhäuser (2007). Integrating Semantic Knowledge into Text Similarity and Information Retrieval. In *Proceedings of the First IEEE International Conference on Semantic Computing*, 57–63, Montreal, Kanada.
- Patwardhan, S., S. Banerjee, and T. Pedersen (2007). UMND1: Unsupervised Word Sense Disambiguation Using Contextual Semantic Relatedness. In *SemEval-2007: Proceedings of the 4th International Workshop on Semantic Evaluations*, 390–393, Prague, Czech Republic.
- Pedersen, T., S. Patwardhan, and J. Michelizzi (2004). WordNet::Similarity - Measuring the Relatedness of Concepts. In *Proceedings of Fifth Annual Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL-04)*, 38–41, Boston, MA.
- Rapp, R. (1999). Automatic Identification of Word Translations from Unrelated English and German Corpora. In *Proceedings of ACL*, 519–526.
- Rapp, R. (2004). A Freely Available Automatically Generated Thesaurus of Related Words. In *Proceedings of LREC 2004*, 395–398.
- Turney, P. D. (2001). Mining the Web for Synonyms: PMI-IR versus LSA on TOEFL. In *Proc. of the Twelfth European Conference on Machine Learning*, 491–502.
- Weeds, J. and D. Weir (2005). Co-occurrence Retrieval: A Flexible Framework for Lexical Distributional Similarity. *Computational Linguistics* 31(4):439–475.