Exploring Knowledge Bases for Similarity

Eneko Agirre[†], Montse Cuadros[‡] German Rigau[†], Aitor Soroa[†]

[†] IXA NLP Group, University of the Basque Country, Donostia, Basque Country,
e.agirre@ehu.es, german.rigau@ehu.es, a.soroa@ehu.es
[‡] TALP center, Universitat Politècnica de Catalunya, Barcelona, Catalonia, cuadros@lsi.upc.edu

Abstract

Graph-based similarity over WordNet has been previously shown to perform very well on word similarity. This paper presents a study of the performance of such a graph-based algorithm when using different relations and versions of Wordnet. Some of the relations are part of the official release of WordNet, and others have been derived automatically. The results show that using the adequate relations the performance improves over previously published WordNet-based results on the WordSim353 dataset. The similarity software and some graphs used in this paper are publicly available at `http://ixa2.si.ehu.es/ukb`.

1. Introduction

Measuring semantic similarity and relatedness between terms is an important problem in lexical semantics (Budanitsky and Hirst, 2006). It has applications in many natural language processing tasks, such as Textual Entailment, Word Sense Disambiguation or Information Extraction, and other related areas like Information Retrieval. Nevertheless, most of the proposed techniques are evaluated over manually curated word similarity datasets like WordSim353 (Finkelstein et al., 2002), in which the weights returned by the systems for word pairs are compared with human ratings.

The techniques used to solve this problem can be roughly classified into two main categories: those relying on preexisting knowledge resources (thesauri, semantic networks, taxonomies or encyclopedias) (Alvarez and Lim, 2007; Yang and Powers, 2005; Hughes and Ramage, 2007; Agirre et al., 2009) and those inducing distributional properties of words from corpora (Sahami and Heilman, 2006; Chen et al., 2006; Bollegala et al., 2007; Agirre et al., 2009).

(Hughes and Ramage, 2007) presented a random walk algorithm over WordNet, with good results on a similarity dataset. In (Agirre et al., 2009) we improved their results and provided the best results among WordNet-based algorithms on the Wordsim353 dataset. Those results are comparable to a distributional method over four billion documents, also presented in (Agirre et al., 2009).

In (Agirre et al., 2009) we already mentioned that different combinations of WordNet relations provide different results. This paper explores in detail a wider range of combinations of relations and improve previous WordNet-based results. The similarity software and graphs used are publicly available under the GPL license.

The paper is structured as follows. We first present the WordNet versions and relations used in this work. Section 3 presents the graph-based algorithm. Section 4 presents the results. Finally, Section 5 presents the conclusions and outlines future work.

2. WordNet relations and versions

WordNet (Fellbaum, 1998b) is a lexical database of English, which groups nouns, verbs, adjectives and adverbs into sets of synonyms (synsets), each expressing a distinct concept. Synsets are interlinked with conceptual-semantic and lexical relations, including hypernymy, meronymy, causality, etc.

The WordNet versions that we use in this work are those integrated into the Multilingual Central Repository or MCR (Atserias et al., 2004) (which includes English WordNet version 1.6 and wordnets for several other languages like Spanish, Italian, Catalan and Basque), and WordNet version 3.0[1].

The version of the MCR used in our experiments comprises relations from WordNet 1.6, WordNet 2.0 relations mapped to 1.6 synsets, eXtended WordNet relations (Mihalcea and Moldovan, 2001), selectional preference relations for subjects and objects of verbs (Agirre and Martinez, 2002) and semantic coocurrence relations. The latter two types of relations are extracted from SemCor, a semantically hand-tagged corpus (Miller et al., 1993). They are thus essentially different from the other relations of the MCR, as they are extracted from a hand-tagged corpus.

Selectional preferences were acquired for subjects and objects from SemCor (Agirre and Martinez, 2002). Semantic co-occurrences were obtained using SemCor measuring the association between word-senses co-occurring in the same sentence (Cuadros et al., 2007).

We have tried three main versions of the MCR in our experiments, as follows:

mcr16.all: all relations in the MCR are used, including SemCor related relations.

mcr16.all_wout_sc: all relations except semantic cooccurrence relations.

mcr16.all_wout_semcor: all relations except semantic cooccurrences and selectional preferences.

The abbreviations in bold will be used across the paper to refer to each version. Regarding WordNet 3.0, we set up two versions.

wn30: all relations in WordNet 3.0.

[1] `http://adimen.si.ehu.es/web/MCR`

Source	#relations
MCR1.6 all	1,650,110
Princeton WN1.6	138,091
Princeton WN3.0	235,402
Princeton WN3.0 gloss relations	409,099
Selectional Preferences from SemCor	203,546
eXtended WN	550,922
Co-occurring relations from SemCor	932,008
KnowNet-5	231,163
KnowNet-10	689,610

Table 1: Number of relations between synsets in each resource.

wn30g: all relations in WordNet 3.0, plus the relation between a synset and the disambiguated words in its gloss[2]

In addition, we have also incorporated relations from KnowNet[3] (Cuadros and Rigau, 2008). KnowNet is an extensible, large and accurate knowledge base, which has been derived by semantically disambiguating small portions of the Topic Signatures acquired from the web[4] (Agirre and Lopez de Lacalle, 2004).
The disambiguation process was performed using WordNet 1.6 as word-sense repository and the SSI-Dijkstra algorithm. SSI-Dijkstra is a knowledge-based graph algorithm which has been shown to be useful for disambiguating topically related terms. KnowNets were created using, as a knowledge source, a large graph containing the relations from the mcr16.all_wout_semcor. SSI-Dijkstra has been also used for assigning WordNet word senses to the Lexical Units associated to a particular FrameNet frame (Laparra and Rigau, 2009). When necessary, KnowNets were ported to WordNet 3.0 using the automatically generated mappings among WordNet versions (Daudé et al., 2003).
KnowNets are available in different sizes, depending on how many words have been included for each Topic Signature (see (Cuadros and Rigau, 2008) for details). We have used two versions in combination with **wn30** and **wn30g**:

k5: KnowNet-5, obtained by disambiguating only the first five words from each Topic Signature.

k10: KnowNet-10, obtained by disambiguating only the first ten words from each Topic Signature.

Table 1 compares the different volumes of semantic relations between synset pairs as used in this work.
For illustrative purposes, we show below some examples of a few relations between WordNet senses, and their corresponding knowledge source:

- **WN** (Fellbaum, 1998a): tree#n#1 –hyponym–> teak#n#2
- **XWN** (Mihalcea and Moldovan, 2001): teak#n#2 –gloss–> wood#n#1
- **spSemCor** (Agirre and Martinez, 2002): read#v#1–tobj–>book#n#1.
- **KnowNet** (Cuadros and Rigau, 2008): woodwork#n#2 –relatedto–> craft#n#1

[2] http://wordnet.princeton.edu/glosstag
[3] http://adimen.si.ehu.es/web/KnowNet
[4] http://ixa.si.ehu.es/Ixa/resources/sensecorpus

3. Personalized PageRank for similarity

We represent WordNet as a graph $G = (V, E)$ as follows: graph nodes represent WordNet concepts (synsets) and dictionary words; relations among synsets are represented by undirected edges; and dictionary words are linked to the synsets associated to them by directed edges.
Given a pair of words and a graph-based representation of WordNet, our method has basically two steps: We first compute the personalized PageRank over WordNet separately for each of the words, producing a probability distribution over WordNet synsets. We then compare how similar these two discrete probability distributions are by encoding them as vectors and computing the cosine between the vectors. We present each step in turn.

3.1. PageRank and Personalized PageRank

The celebrated PageRank algorithm (Page et al., 1999) is a method for ranking the vertices in a graph according to their relative structural importance. The main idea of PageRank is that whenever a link from v_i to v_j exists in a graph, a vote from node i to node j is produced, and hence the rank of node j increases. Besides, the strength of the vote from i to j also depends on the rank of node i: the more important node i is, the more strength its votes will have. Alternatively, PageRank can also be viewed as the result of a random walk process, where the final rank of node i represents the probability of a random walk over the graph ending on node i, at a sufficiently large time.
Let G be a graph with N vertices $v_1, \ldots, v_N$ and d_i be the outdegree of node i; let M be a $N \times N$ transition probability matrix, where $M_{ji} = \frac{1}{d_i}$ if a link from i to j exists, and zero otherwise. Then, the calculation of the *PageRank vector* $\mathbf{Pr}$ over G is equivalent to resolving Equation (1).

$$\mathbf{Pr} = cM\mathbf{Pr} + (1 - c)\mathbf{v} \quad (1)$$

In the equation, $\mathbf{v}$ is a $N \times 1$ vector whose elements are $\frac{1}{N}$ and c is the so called *damping factor*, a scalar value between 0 and 1. The first term of the sum on the equation models the voting scheme described in the beginning of the section. The second term represents, loosely speaking, the probability of a surfer randomly jumping to any node, e.g. without following any paths on the graph. The damping factor, usually set in the $[0.85..0.95]$ range, models the way in which these two terms are combined at each step.
The second term on Eq. (1) can also be seen as a smoothing factor that makes any graph fulfill the property of being aperiodic and irreducible, and thus guarantees that PageRank calculation converges to a unique stationary distribution.
In the traditional PageRank formulation the vector $\mathbf{v}$ is a stochastic normalized vector whose element values are all $\frac{1}{N}$, thus assigning equal probabilities to all nodes in the graph in case of random jumps. However, as pointed out by (Haveliwala, 2002), the vector $\mathbf{v}$ can be non-uniform

and assign stronger probabilities to certain kinds of nodes, effectively biasing the resulting PageRank vector to prefer these nodes. For example, if we concentrate all the probability mass on a unique node i, all random jumps on the walk will return to i and thus its rank will be high; moreover, the high rank of i will make all the nodes in its vicinity also receive a high rank. Thus, the importance of node i given by the initial distribution of $\mathbf{v}$ spreads along the graph on successive iterations of the algorithm. We call this variant of PageRank *Personalized PageRank*.

Given a target word, we compute the personalized PageRank over the WordNet graph for the word, that is, we initialize $\mathbf{v}$ in Eq. (1) with equal probabilities for all synsets corresponding to the target word, while the rest of the synsets are initialized to zero.

Regarding implementation details, Eq. (1) is solved applying an iterative algorithm, computing Eq. (1) successively until convergence below a given threshold is achieved, or, more typically, until a fixed number of iterations are executed. We chose a damping value of 0.85 and finish the calculation after 30 iterations. These are default values, and we did not optimize them.

3.2. Computing similarity

Once personalized PageRank is computed, it returns a probability distribution over WordNet synsets. The similarity between two words can thus be implemented as the similarity between the probability distributions. Alternatively, we can interpret the probability distribution for a word w as a vector $\vec{w}$ of weights w_i where each dimension i is a synset, and use the cosine to compute similarity, as in Eq 2.

$$\begin{aligned} \mathbf{similarity}(\vec{w},\vec{v}) &= \cos(\theta(\vec{w},\vec{v})) \\ &= \frac{\vec{w}\cdot\vec{v}}{\|\vec{w}\|\|\vec{v}\|} \\ &= \frac{\sum_{i=1}^{n} w_i v_i}{\sqrt{\sum_{i=1}^{n} w_i^2}\sqrt{\sum_{i=1}^{n} v_i^2}} \end{aligned} \quad (2)$$

4. Results

We have tested the various sets of relations on the WordSim353 dataset (Finkelstein et al., 2002)[5], which contains 353 word pairs, each associated with an average of 13 to 16 human judgements. Both similarity and relatedness are annotated without any distinction. Several studies indicate that the human scores consistently have very high correlations with each other (Miller and Charles, 1991; Resnik, 1995), thus validating the use of these kind of datasets for evaluating semantic similarity.

The results in Table 2 show the results as the Spearman correlation for several wordnet versions and relations. Not all words in the dataset are in WordNet[6], meaning that we are not able to return a result in 9 pairs out of the 353. The *Known-word* column in Table 2 reports the Spearman value for the rest of pairs. Given the wide confidence intervals, most of the differences are not statistically significant. Note that the similarity literature rarely reports statistical significances, and limit themselves to report performance differences.

The main conclusions from the results are the following:

- The best combinations for MCR1.6 are obtained ignoring selectional preferences and semantic occurrences.
- The disambiguated glosses improve the results by a large margin on wn30.
- KnowNet improves results in both datasets. The largest gains are for MCR1.6 with KnowNet-10 (k10), but the best overall results are for Wordnet3.0 with disambiguated glosses and KnowNet-5 (k5)

All relations sources added to WordNet seem to improve results, with the exception of semantic cooccurrences and selectional preferences, which actually degrade performance. While adding relations was expected to improve the results, we were greatly surprised to see that the semantic cooccurrences and selectional preferences were harmful. These relations come from a manually annotated corpus, and improved the results in an word sense disambiguation task (Agirre and Soroa, 2008). We are still investigating the cause of the poor performance of these relations.

5. Related work

Table 3 shows the results of some noteworthy systems in the literature. Compared to the systems in the literature, we outperform the best systems that only use Wordnet. The algorithm is the same as in (Agirre et al., 2009), with the only change in the use of new relations, which in the case of KnowNet-5 improves performance in 3 points. Note that the results over the pairs with both words in Wordnet are very close to (Gabrilovich and Markovitch, 2007) (0.72 vs. 0.75), which is the best system using a single knowledge source (Wikipedia). The best result overall is obtained using a supervised combination of distributional algorithms and our WordNet-based algorithm (Agirre et al., 2009).

6. Conclusions and future work

The study presented in this paper shows that choosing the right version of WordNet and the right set of relations is important to obtain good similarity results. The combination of all relations in WN3.0, the disambiguated glosses, and the automatically derived relations from KnowNets produces the best results to date using WordNet on this dataset. The similarity algorithm and needed resources are publicly available under the GPL license at `http://ixa2.si.ehu.es/ukb/`.

For the future, we would like to perform a similar study on WSD using a related algorithm(Agirre and Soroa, 2009), and compare which is the best setting on these closely interrelated tasks.

[5] `http://www.cs.technion.ac.il/~gabr/resources/data/wordsim353/wordsim353.html`

[6] As we assumed that all words are nouns in singular form, we missed the following words: *media*, *live*, *children*, *eat*, *earning*, *defeating* and *Maradona*.

Method	Spearman	Known-words	interval
mcr16.all	0.369690	0.395788	[0.275818, 0.456578]
mcr16.all_wout_sc	0.449606	0.479641	[0.362092, 0.529263]
mcr16.all_wout_semcor	0.525343	0.559497	[0.445263, 0.597086]
mcr16.all_wout_semcor+k5	0.553766	0.589597	[0.476836, 0.622276]
mcr16.all_wout_semcor+k10	0.565809	0.602374	[0.490275, 0.632907]
wn30	0.559087	0.588069	[0.482770, 0.626976]
wn30g	0.658218	0.692505	[0.594597, 0.713647]
wn30g+k5	**0.685184**	**0.720859**	[0.625450, 0.736934]
wn30g+k10	0.638901	0.672213	[0.572612, 0.696891]

Table 2: Wordsim353 results for various wordnet versions and relations. Spearman report the correlation with the gold standard. Known-words reports the Spearman correlation for pairs where both words are in WordNet. The last column corresponds to the 95% confidence interval. Please check Section 2 for the meaning of the abbreviations used in this table.

Method	Source	Spearman
(Hughes and Ramage, 2007)	WordNet	0.55
(Finkelstein et al., 2002)	LSA	0.56
(Finkelstein et al., 2002)	Combination	0.56
(Gabrilovich and Markovitch, 2007)	ODP	0.65
(Agirre et al., 2009)	Web Corpus	0.65
(Agirre et al., 2009)	WordNet	0.66
This work	WordNet	0.69
(Gabrilovich and Markovitch, 2007)	Wikipedia	0.75
(Agirre et al., 2009)	Combination	**0.78**

Table 3: Comparison with previous work for WordSim353, inclusing Spearman.

Acknowledgments

This work has been supported by KNOW (TIN2006-15049-C03-03, TIN2006-15049-C03-01), KNOW-2 (TIN2009-14715-C04-04) and KYOTO (ICT-2007-211423). We want to thank the anonymous reviewers for their comments.

7. References

Agirre, E. and Lopez de Lacalle, O. (2004). Publicly available topic signatures for all wordnet nominal senses. In *Proceedings of LREC*.

Agirre, E. and Martinez, D. (2002). Integrating selectional preferences in wordnet. In *Proceedings of the First International WordNet Conference*, Mysore, India.

Agirre, E. and Soroa, A. (2008). Using the multilingual central repository for graph-based word sense disambiguation. In *Proceedings of LREC*.

Agirre, E. and Soroa, A. (2009). Personalizing pagerank for word sense disambiguation. In *Proc. of EACL 2009*, Athens, Greece.

Agirre, E., Soroa, A., Alfonseca, E., Hall, K., Kravalova, J., and Pasca, M. (2009). A study on similarity and relatedness using distributional and WordNet-based approaches. In *Proceedings of annual meeting of the North American Chapter of the Association of Computational Linguistics (NAAC)*, Boulder, USA.

Alvarez, M. and Lim, S. (2007). A graph modeling of semantic similarity between words. *Proceedings of the Conference on Semantic Computing*, pages 355–362.

Atserias, J., Villarejo, L., Rigau, G., Agirre, E., Carroll, J., Magnini, B., and Vossen, P. (2004). The meaning multilingual central repository. In *Proc. of Global WordNet Conference*, Brno, Czech Republic.

Bollegala, D., Y., M., and Ishizuka, M. (2007). Measuring semantic similarity between words using web search engines. In *Proceedings of WWW'2007*.

Budanitsky, A. and Hirst, G. (2006). Evaluating WordNet-based Measures of Lexical Semantic Relatedness. *Computational Linguistics*, 32(1):13–47.

Chen, H., Lin, M., and Wei, Y. (2006). Novel association measures using web search with double checking. In *Proceedings of COCLING/ACL 2006*.

Cuadros, M. and Rigau, G. (2008). KnowNet: Building a Large Net of Knowledge from the Web. In *Proceedings of COLING*.

Cuadros, M., Rigau, G., and Castillo, M. (2007). Evaluating large-scale knowledge resources across languages. In *Proceedings of RANLP*.

Daudé, J., Padró, L., and Rigau, G. (2003). Making Wordnet Mappings Robust. In *Proceedings of the 19th Congreso de la Sociedad Espala para el Procesamiento del Lenguage Natural, SEPLN'03*, Universidad Universidad de Alcala de Henares. Madrid, Spain.

Fellbaum, C. (1998a). *WordNet. An Electronic Lexical Database*. Language, Speech, and Communication. The MIT Press.

Fellbaum, C., editor (1998b). *WordNet: An Electronic Lex-*

ical Database and Some of its Applications. MIT Press, Cambridge, Mass.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., and Ruppin, E. (2002). Placing Search in Context: The Concept Revisited. *ACM Transactions on Information Systems*, 20(1):116–131.

Gabrilovich, E. and Markovitch, S. (2007). Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis. *Proc of IJCAI*, pages 6–12.

Haveliwala, T. H. (2002). Topic-sensitive pagerank. In *WWW '02: Proceedings of the 11th international conference on World Wide Web*, pages 517–526.

Hughes, T. and Ramage, D. (2007). Lexical semantic relatedness with random graph walks. In *Proceedings of EMNLP-CoNLL-2007*, pages 581–589.

Laparra, E. and Rigau, G. (2009). Integrating wordnet and framenet using a knowledge-based word sense disambiguation algorithm. In *Proceedings of Recent Advances in Natural Language Processing (RANLP09)*, Borovets, Bulgaria.

Mihalcea, R. and Moldovan, D. (2001). extended wordnet: Progress report. In *NAACL Workshop* WordNet and Other Lexical Resources: Applications, Extensions and Customizations *(NAACL'2001)*., pages 95–100, Pittsburg, PA, USA.

Miller, G. and Charles, W. (1991). Contextual correlates of semantic similarity. *Language and Cognitive Processes*, 6(1):1–28.

Miller, G., Leacock, C., Tengi, R., and Bunker, R. (1993). A Semantic Concordance. In *Proceedings of the ARPA Workshop on Human Language Technology*.

Page, L., Brin, S., Motwani, R., and Winograd, T. (1999). The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab. Previous number = SIDL-WP-1999-0120.

Resnik, P. (1995). Using Information Content to Evaluate Semantic Similarity in a Taxonomy. *Proc. of IJCAI*, 14:448–453.

Sahami, M. and Heilman, T. (2006). A web-based kernel function for measuring the similarity of short text snippets. *Proc. of WWW*, pages 377–386.

Yang, D. and Powers, D. (2005). Measuring semantic similarity in the taxonomy of WordNet. *Proceedings of the Australasian conference on Computer Science*.