Measuring semantic relatedness with vector space models and random walks

Amaç Herdağdelen
Center for Mind/Brain Sciences
University of Trento
amac@herdagdelen.com

Katrin Erk
Linguistics Department
University of Texas at Austin
katrin.erk@mail.utexas.edu

Marco Baroni
Center for Mind/Brain Sciences
University of Trento
marco.baroni@unitn.it

Abstract

Both vector space models and graph random walk models can be used to determine similarity between concepts. Noting that vectors can be regarded as local views of a graph, we directly compare vector space models and graph random walk models on standard tasks of predicting human similarity ratings, concept categorization, and semantic priming, varying the size of the dataset from which vector space and graph are extracted.

1 Introduction

Vector space models, representing word meanings as points in high-dimensional space, have been used in a variety of semantic relatedness tasks (Sahlgren, 2006; Padó and Lapata, 2007). Graphs are another way of representing relations between linguistic entities, and they have been used to capture semantic relatedness by using both corpus-based evidence and the graph structure of WordNet and Wikipedia (Pedersen et al., 2004; Widdows and Dorow, 2002; Minkov and Cohen, 2008). We study the relationship between vector space models and graph random walk models by embedding vector space models in graphs. The flexibility offered by graph random walk models allows us to compare the vector space-based similarity measures to extended notions of relatedness and similarity. In particular, a random walk model can be viewed as smoothing direct similarity between two vectors using second-order and even higher-order vectors. This view leads to the second focal point of this paper: We investigate whether random walk models can simulate the smoothing effects obtained by methods like Singular Value Decomposition (SVD). To answer this question, we compute models on reduced (downsampled) versions of our dataset and evaluate the robustness of random walk models, a classic vector-based model, and SVD-based models against data sparseness.

2 Model definition and implementation

We use directed graphs with weighted edges, $G = (V, E, w)$ where V is a set of nodes, $E = V \times V$ is a set of edges and $w : E \rightarrow \mathbb{R}$ is the weighting function on edges. For simplicity, we assume that G is fully connected, edges with zero weights can be considered as non-existing in the graph. On these graphs, we perform random walks with an initial probability distribution $\mathbf{q}$ over the nodes (a $1 \times |V|$ vector). We then follow edges with probability proportional to their weights, so that the probability of walking from node v_1 to node v_2 is $w(v_1, v_2) / \sum_v w(v_1, v)$. A *fixed length* random walk ends after a predetermined number of steps. In *flexible* walks, there is a constant probability γ of stopping at each step. Thus, walk length follows a geometric distribution with parameter γ, the probability of a walk of length k is $\gamma(1-\gamma)^{k-1}$ and the expected walk length is $1/\gamma$. For example, a flexible walk with $\gamma = 1/2$ will produce 1-step, 2-step, and higher-step walks while the expected average length is 2.

Relating vectors and graphs. Corpus co-occurrence (e_1, e_2, a_{12}) of two entities e_1 and e_2 that co-occur with (potentially transformed) count a_{12} can be represented in either a vector or a graph. In a vector, it corresponds to a dimension value of a_{12} for the dimension e_2 of entity e_1. In a graph, it corresponds to two nodes labeled e_1 and e_2 connected by an edge with weight a_{12}.

Similarity measures. Let $R(\mathbf{q}) = \mathbf{p}$ denote a specific random walk process which transforms an

Proceedings of the 2009 Workshop on Graph-based Methods for Natural Language Processing, ACL-IJCNLP 2009, pages 50–53,
Suntec, Singapore, 7 August 2009. ©2009 ACL and AFNLP

initial probability distribution $\mathbf{q}$ to a final probability distribution $\mathbf{p}$ over the nodes. We write $q(m)$ for the probability assigned to the node m under $\mathbf{q}$. If the initial distribution $\mathbf{q}$ concentrates all probability on a single node n, i.e., $q(n) = 1$ and $q(x) = 0$ for all nodes $x \neq n$, we write $pr(n \rightarrow m)$ for the probability $p(m)$ of ending up at node m.

The simplest way of measuring relatedness through random walks is to consider the probability $p(m)$ of a single node m as an endpoint for a walk starting with start probability distribution $\mathbf{q}$, that is, $\mathbf{p} = R(\mathbf{q})$. We call this a **direct, one-direction** measure of relatedness between $\mathbf{q}$ and m. Direct, one-direction measures are typically asymmetric. In case all start probability is concentrated on a single node n, we can also consider **direct, two-direction** measures, which will be a combination of $pr(m \rightarrow n)$ and $pr(n \rightarrow m)$. The point of using two-direction measures is that these can be made symmetric, which is an advantage when we are modeling undirected semantic similarity. In the experiments below we focus on the average of the two probabilities.

In addition to direct measures, we will use **indirect measures**, in which we compute the relatedness of endpoint probability distributions $\mathbf{p_1} = R(\mathbf{q_1})$ and $\mathbf{p_2} = R(\mathbf{q_2})$. As endpoint distributions can be viewed both as probability distributions and as vectors, we used three indirect measures: 1) Jensen/Shannon divergence, a symmetric variant of the Kullback/Leibler divergence between probability distributions. 2) cosine similarity, and 3) dot product. Dot product is a natural choice in a graph setting because we can view it as the probability of a pair of walks, one starting at a node determined by $\mathbf{q_1}$ and the other starting at a node governed by $\mathbf{q_2}$, ending at the same node.

Discussion. Direct and indirect relatedness measures together with variation in walk length give us a simple, powerful and flexible way to capture different kinds of similarity (with traditional vector-based approach as a special case). Longer walks or flexible walks will capture higher order effects that may help coping with data sparseness, similar to the use of second-order vectors. Dimensionality reduction techniques like Singular Value Decomposition (SVD) also capture these higher-order effects, and it has been argued that that makes them more resistant against sparseness (Schütze, 1997). To our knowledge, no systematic comparison of SVD and classical vector-based methods has been done on different corpus sizes. In our experiments, we will compare the performance of SVD and flexible-walk smoothing at different corpus sizes and for a variety of tasks.

Implementation: We extract tuples from the 2-billion word ukWaC corpus,[1] dependency-parsed with MINIPAR.[2] Following Padó and Lapata (2007), we only consider co-occurrences where two target words are connected by certain dependency paths, namely: the top 30 most frequent preposition-mediated noun-to-noun paths (*soldier+with+gun*), the top 50 transitive-verb-mediated noun-to-noun paths (*soldier+use+gun*), the top 30 direct or preposition-mediated verb-noun paths (*kill+obj+victim*, *kill+in+school*), and the modifying and predicative adjective-to-noun paths. Pairs (w_1, w_2) that account for 0.01% or less of the marginal frequency of w_1 were trimmed. The resulting tuple list, with raw counts converted to mutual information scores, contains about 25 million tuples.

To test how well graph-based and alternative methods "scale down" to smaller corpora, we sampled random subsets of tuples corresponding to 0.1%, 1%, 10%, and 100% of the full list. To put things into perspective, the full list was extracted from a corpus of about 2 billion words; so, the 10% list is on the order of magnitude of the BNC, and the 0.1% list is on the order of magnitude of the Brown corpus. From each of the 4 resulting datasets, we built one graph and two vector space models: one space with full dimensionality, and one space reduced to 300 dimensions using singular value decomposition.

3 Experiments

First, we report the results for all tasks obtained on the full data-set and then proceed with the comparison of different models on differing graph sizes to see the robustness of the models against data sparseness.

Human similarity ratings: We use the dataset of Rubenstein and Goodenough (1965), consisting of averages of subject similarity ratings for 65 noun pairs. We use the Pearson's coefficient between estimates and human judgments as our performance measure. The results obtained for

[1] http://wacky.sslmit.unibo.it

[2] http://www.cs.ualberta.ca/~lindek/minipar.htm

	Direct (average)			Vector (cosine)		Indirect (dot product)			Previous
	0.5	1	2	svd	vector	0.5	1	2	
RG	0.409	0.326	0.571	0.798	0.689	0.634	0.673	0.400	BL: 0.70 CLW: 0.849
AAMP Purity	0.480	0.418	0.669	0.701	0.704	0.664	0.667	0.612	AP: 0.709 RS: 0.791
				Hodgson					
synonym	$2,563$	1.289	$5,408^{**}$	10.015^{**}	$6,623^{**}$	$5,462^{**}$	$5,954^{**}$	$5,537^{**}$	
coord	$4,275^{**}$	$3,969^{**}$	$6,319^{**}$	11.157^{**}	$7,593^{**}$	$8,466^{**}$	$8,477^{**}$	$4,854^{**}$	
antonym	$2,853*$	$2,237$	$5,319^{**}$	$7,724^{**}$	$5,455^{**}$	$4,589^{**}$	$4,859^{**}$	$6,810^{**}$	
conass	$9,209^{**}$	10.016^{**}	$5,889^{**}$	$9,299^{**}$	$6,950^{**}$	$5,993^{**}$	$5,455^{**}$	$4,994^{**}$	
supersub	$4,038^{**}$	$4,113^{**}$	$6,773^{**}$	10.422^{**}	$7,901^{**}$	$6,792^{**}$	$7,165^{**}$	$4,828^{**}$	
phrasacc	$4,577^{**}$	$4,718^{**}$	$2,911^{*}$	$3,532^{*}$	$3,023^{*}$	$3,506^{*}$	$3,612^{*}$	1.038	

Table 1: All datasets. * (**) indicates significance level $p < 0.01$ ($p < 0.001$). BL: (Baroni and Lenci, 2009), CLW: (Chen et al., 2006), AP: (Almuhareb, 2006), RS: (Rothenhäusler and Schütze, 2009)

	0.1%			1%			10%		
	cos svd	cos vector	dot 2	cos svd	cos vector	dot 2	cos svd	cos vector	dot 2
RG	0.219	0.244	**0.669**	0.676	0.700	**1.159**	0.911	0.829	**1.068**
AAMP	**0.379**	0.339	0.366	**0.723**	0.622	0.634	0.923	0.886	**0.948**
Synonym	0.369	0.464	**0.610**	0.493	0.590	**0.833**	0.857	0.770	**1.081**
Antonym	0.449	**0.493**	0.231	**0.768**	0.585	0.730	**1.044**	0.849	0.977
Conass	0.187	0.260	**0.261**	0.451	0.498	**0.942**	0.857	0.704	**1.062**
Coord	0.282	0.362	**0.456**	0.527	0.570	**1.050**	0.927	0.810	**1.187**
Phrasacc	0.268	0.132	**0.761**	0.849	0.610	**1.215**	0.920	0.868	**1.049**
Supersub	0.313	**0.353**	0.285	0.645	0.601	**1.029**	0.936	0.752	**1.060**

Table 2: Each cell contains the ratio of the performance of the corresponding model for the corresponding downsampling ratio to the performance of the same model on the full graph. The higher ratio means the less deterioration due to data sparseness.

the full graph are in Table 1, line 1. The SVD model clearly outperforms the pure-vector based approach and the graph-based approaches. Its performance is above that of previous models trained on the same corpus (Baroni and Lenci, 2009). The best model that we report is based on web search engine results (Chen et al., 2006). Among the graph-based random walk models, flexible walk with parameter 0.5 and fixed 1-step walk with indirect relatedness measures using dot product similarity achieve the highest performance.

Concept categorization: Almuhareb (2006) proposed a set of 402 nouns to be categorized into 21 classes of both concrete (animals, fruit…) and abstract (feelings, times…) concepts. Our results on this clustering task are given in Table 1 (line 2). The difference between SVD and pure-vector models is negligible and they both obtain the best performance in terms of both cluster entropy (not shown in the table) and purity. Both models' performances are comparable with the previously reported studies, and above that of random walks.

Semantic priming: The next dataset comes from Hodgson (1991) and it is of interest since it requires capturing different forms of semantic relatedness between prime-target pairs: synonyms (*synonym*), coordinates (*coord*), antonyms (*antonym*), free association pairs (*conass*), super- and subordinate pairs (*supersub*) and phrasal associates (*phrasacc*). Following previous simulations of this data-set (Padó and Lapata, 2007), we measure the similarity of each related target-prime pair, and we compare it to the average similarity of the target to all the other primes instantiating the same relation, treating the latter quantity as our surrogate of an unrelated target-prime pair. We report results in terms of differences between unrelated and related pairs, normalized to t-scores, marking significance according to two-tailed paired t-tests for the relevant degrees of freedom. Even though the SVD-based and pure-vector models are among the top achievers in general, we see that in different tasks different random walk models achieve comparable or even better performances. In particular, for phrasal associates and conceptual associates, the best results are obtained by random walks based on direct measures.

3.1 Robustness against data sparseness

So far, we reported only the results obtained on the full graph. However, in order to see the response of the models to using smaller corpora

we ran another set of experiments on artificially down-sampled graphs as explained above. In this case, we are not interested in the absolute performance of the models per se but the relative performance. Thus, for ease of comparison we fixed each model's performance on the full graph to 1 for each task and linearly scaled its performance on smaller graphs. For example saying that the SVD-based model achieves a score of 0.911 on 10% graph for the Rubenstein and Goodenough dataset means that the ratio of the performance of SVD-based model on 10% graph to the performance of the same model on the full graph is 0.911. The results are given in Table 2, where the only random walk model we report is *dot 2*, i.e., a 2-step random walk coupled with the dot product-based indirect measure. This is by far the random walk model most robust to downsampling. In the 10% graph, we see that on all tasks but one, dot 2 is the model least affected by the data reduction. On the contrary, down-sampling has a positive effect on this model because on 6 tasks, it actually performs better than it does on the full graph! The same behavior is also observed on the 1% graph - as an example, for phrasal associates relations, dot 2 performance increases by a factor of around 1.2 when we use one hundredth of the graph instead of the full one. For the smallest graph we used, 0.1%, still dot 2 provides the highest relative performance in 5 out of the 8 tasks.

4 Conclusion

We compared graph-based random walk models and vector models. For this purpose, we showed how corpus co-occurrences could be represented both as a graph and a vector, and we identified two different ways to calculate relatedness based on the outcomes of random walks, by direct and indirect measures. The experiments carried out on 8 different tasks by using the full graph revealed that SVD-based model performs very well across all types of semantic relatedness. However, there is also evidence that -depending on the particular relation- some random walk models can achieve results as good as or even better than those of SVD-based models. Our second question was whether the random walk models would be able to simulate the smoothing effects obtained by SVD. While answering this question, we also carried out a systematic comparison of plain and SVD-based models on different tasks with different sizes of data. One interesting result is that an SVD-based model is not necessarily more robust to data sparseness than the plain vector model. The more interesting result is that a 2-step random walk model, based on indirect measures with dot product, consistently outperforms both SVD-based and plain vector models in terms of relative performance, thus it is able to achieve comparable results on very small datasets. Actually, the improvement on absolute performance measures of this random walk by making the dataset even smaller calls for future research.

References

A. Almuhareb. 2006. *Attributes in lexical acquisition*. Dissertation, University of Essex.

M. Baroni and A. Lenci. 2009. One distributional memory, many semantic spaces. In *Proceedings of GEMS*, Athens, Greece.

Hsin-Hsi Chen, Ming-Shun Lin, and Yu-Chuan Wei. 2006. Novel association measures using web search with double checking. In *Proceedings of ACL*, pages 1009–16.

J. Hodgson. 1991. Informational constraints on prelexical priming. *Language and Cognitive Processes*, 6:169–205.

Einat Minkov and William W. Cohen. 2008. Learning graph walk based similarity measures for parsed text. In *Proceedings of EMNLP'08*.

S. Padó and M. Lapata. 2007. Dependency-based construction of semantic space models. *Computational Linguistics*, 33(2):161–199.

T. Pedersen, S. Patwardhan, and J. Michelizzi. 2004. Wordnet::similarity: Measuring the relatedness of concepts. In *Proceedings of NAACL*, pages 38–41.

Klaus Rothenhäusler and Hinrich Schütze. 2009. Unsupervised classification with dependency based word spaces. In *Proceedings of GEMS*, pages 17–24.

H. Rubenstein and J.B. Goodenough. 1965. Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633.

M. Sahlgren. 2006. *The Word-Space Model*. Dissertation, Stockholm University.

H. Schütze. 1997. *Ambiguity Resolution in Natural Language Learning*. CSLI, Stanford.

Dominic Widdows and Beate Dorow. 2002. A graph model for unsupervised lexical acquisition. In *19th International Conference on Computational Linguistics*, pages 1093–1099.