From Global to Local Similarities:
A Graph-Based Contextualization Method using Distributional Thesauri

Chris Biemann and **Martin Riedl**
Computer Science Department, Technische Universität Darmstadt
Hochschulstrasse 10, D-64289 Darmstadt, Germany
{riedl,biem}@cs.tu-darmstadt.de

Abstract

After recasting the computation of a distributional thesaurus in a graph-based framework for term similarity, we introduce a new contextualization method that generates, for each term occurrence in a text, a ranked list of terms that are semantically similar and compatible with the given context. The framework is instantiated by the definition of term and context, which we derive from dependency parses in this work. Evaluating our approach on a standard data set for lexical substitution, we show substantial improvements over a strong non-contextualized baseline across all parts of speech. In contrast to comparable approaches, our framework defines an unsupervised generative method for similarity in context and does not rely on the existence of lexical resources as a source for candidate expansions.

1 Introduction

Following (de Saussure, 1916) we consider two distinct viewpoints: *syntagmatic* relations consider the assignment of values to a linear sequence of terms, and the *associative (also: paradigmatic)* viewpoint assigns values according to the commonalities and differences to other terms in the reader's memory. Based on these notions, we automatically expand terms in the linear sequence with their paradigmatically related terms.

Using the distributional hypothesis (Harris, 1951), and operationalizing similarity of terms (Miller and Charles, 1991), it became possible to compute term similarities for a large vocabulary (Ruge, 1992). Lin (1998) computed a *distributional thesaurus (DT)* by comparing context features defined over grammatical dependencies with an appropriate similarity measure for all reasonably frequent words in a large collection of text, and evaluated these automatically computed word similarities against lexical resources. Entries in the DT consist of a ranked list of the globally most similar terms for a *target* term. While the similarities are dependent on the instantiation of the context feature as well as on the underlying text collection, they are global in the sense that the DT aggregates over all occurrences of target and its similar elements. In our work, we will use a DT in a graph representation and move from a global notion of similarity to a contextualized version, which performs context-dependent text expansion for all word nodes in the DT graph.

2 Related Work

The need to model semantics just in the same way as local syntax is covered by the n-gram-model, i.e. trained from a background corpus sparked a large body of research on semantic modeling. This includes computational models for topicality (Deerwester et al., 1990; Hofmann, 1999; Blei et al., 2003), and language models that incorporate topical (as well as syntactic) information, see e.g. (Boyd-Graber and Blei, 2008; Tan et al., 2012). In the Computational Linguistics community, the vector space model (Schütze, 1993; Turney and Pantel, 2010; Baroni and Lenci, 2010; Pucci et al., 2009; de Cruys et al., 2013) is the prevalent metaphor for representing word meaning.

While the computation of semantic similarities on the basis of a background corpus produces a global model, which e.g. contains semantically similar words for different word senses, there are a number of works that aim at contextualizing the information held in the global model for particular occurrences. With his predication algorithm, Kintsch (2001) contextualizes LSA (Deerwester et al., 1990) for N-VP constructions by spreading activation over neighbourhood graphs in the latent space.

In particular, the question of operationalizing semantic compositionality in vector spaces (Mitchell

Proceedings of the TextGraphs-8 Workshop, pages 39–43,
Seattle, Washington, USA, 18 October 2013. ©2013 Association for Computational Linguistics

and Lapata, 2008) received much attention. The lexical substitution task (McCarthy and Navigli, 2009) (LexSub) sparked several approaches for contextualization. While LexSub participants and subsequent works all relied on a list of possible substitutions as given by one or several lexical resources, we describe a graph-based system that is knowledge-free and unsupervised in the sense that it neither requires an existing resource (we compute a DT graph for that), nor needs training for contextualization.

3 Method

3.1 Holing System

For reasons of generality, we introduce the holing operation (cf. (Biemann and Riedl, 2013)), to split any sort of observations on the syntagmatic level (e.g. dependency relations) into pairs of term and context features. These pairs are then both used for the computation of the global DT graph similarity and for the contextualization. This holing system is the only part of the system that is dependent on a pre-processing step; subsequent steps operate on a unified representation. The representation is given by a list of pairs <t,c> where t is the term (at a certain offset) and c is the context feature. The position of t in c is denoted by a hole symbol '@'. As an example, the dependency triple $(\texttt{nsub;gave}_2\texttt{;I}_1)$ could be transferred to $<\texttt{gave}_2\texttt{, (nsub;@;I}_1\texttt{)}>$ and $<\texttt{I}_1\texttt{, (nsub;gave}_2\texttt{;@)}>$.

3.2 Distributional Similarity

Here, we present the computation of the distributional similarity between terms using three graphs. For the computation we use the Apache Hadoop Framework, based on (Dean and Ghemawat, 2004).

We can describe this operation using a bipartite "term"-"context feature" graph $TC(T, C, E)$ with T the set terms, C the set of context features and $e(t, c, f) \in E$ edges between $t \in T$, $c \in C$ with $f = count(t, c)$ frequency of co-occurrence. Additionally, we define $count(t)$ and $count(c)$ as the counts of the term, respectively as the count of the context feature. Based on the graph TC we can produce a first-order graph $FO(T, C, E)$, with $e(t, c, sig) \in E$. First, we calculate a significance score sig for each pair (t, c) using Lexicographer's Mutual Information (LMI): $score(t, c) = LMI(t, c,) = count(t, c) \log_2(\frac{count(t,c)}{count(t)count(c)})$ (Evert, 2004). Then, we remove all edges with $score(t, c) < 0$ and keep only the p most significant pairs per term t and remove the remaining edges. Additionally, we remove features which co-occur with more then 1000 words, as these features do not contribute enough to similarity to justify the increase of computation time (cf. (Rychlý and Kilgarriff, 2007; Goyal et al., 2010)). The second-order similarity graph between terms is defined as $SO(T, E)$ for $t_1, t_2 \in T$ with the similarity score $s = |\{c|e(t_1, c) \in FO, e(t_2, c) \in FO\}|$, which is the number of salient features two terms share. SO defines a distributional thesaurus.

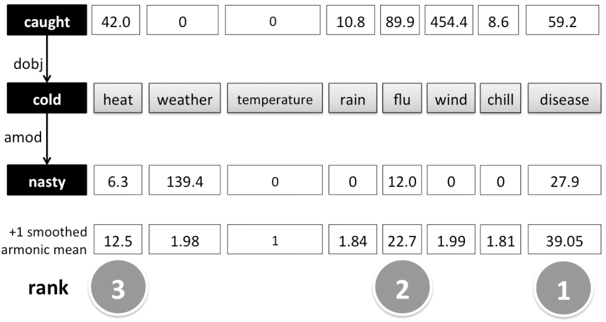
In contrast to (Lin, 1998) we do not count how often a feature occurs with a term (we use significance ranking instead), and do not use cosine or other similarities (Lee, 1999) to calculate the similarity over the feature counts of each term, but only count significant common features per term. This constraint makes this approach more scalable to larger data, as we do not need to know the full list of features for a term pair at any time. Seemingly simplistic, we show in (Biemann and Riedl, 2013) that this measure outperforms other measures on large corpora in a semantic relatedness evaluation.

3.3 Contextual Similarity

The contextualization is framed as a ranking problem: given a set of candidate expansions as provided by the SO graph, we aim at ranking them such that the most similar term in context will be ranked higher, whereas non-compatible candidates should be ranked lower.

First, we run the holing system on the lexical material containing our target word $tw \in T' \subseteq T$ and select all pairs $<tw, c_i>$ $c_i \in C' \subseteq C$ that are instantiated in the current context. We then define a new graph $CON(T', C', S)$ with context features $c_i \in C'$. Using the second-order similarity graph $SO(T, E)$ we extract the top n similar terms $T' = \{t_i, \ldots, t_n\} \subseteq T$ from the second-order graph SO for tw and add them to the graph CON. We add edges $e(t, c, sig)$ between all target words and context features and label the edge with the significance score from the first order graph FO. Edges $e(t, c, sig)$, not contained in FO, get a significance score of zero. We can then calculate a ranking score for each t_i with the harmonic mean, using a plus one smoothing: $rank(t_i) = \frac{\prod_{c_j}(sig(t_i,c_j)+1)/count(term(c_j))}{\sum_{c_j}(sig(t_i,c_j)+1)/count(term(c_j))}$ ($term(c_j)$ extracts the term out of the context notation). Using that ranking score we can re-order the entries $t_1, \ldots, t_n$ according to their ranking score.

In Figure 1, we exemplify this, using the target word tw= ”cold” in the sentence ”I caught a nasty cold.”. Our dependency parse-based holing system produced the following pairs for ”cold”: <cold$_5$,(amod;@;nasty$_4$)>, <cold$_5$,(dobj;caught$_2$;@)>. The top 10 candidates for ”cold” are T'={heat, weather, temperature, rain, flue, wind, chill, disease}. The scores per pair are e.g. <heat, (dobj;caught;@)> with an LMI score of 42.0, <weather ,(amod;@;nasty)> with a score of 139.4. The pair <weather, (dobj;caught;@)> was not contained in our first-order data. Ranking the candidates by their overall scores as given in the figure, the top three contextualized expansions are ”disease, flu, heat”, which are compatible with both pairs. For the top 200 words, the ranking of fully compatible candidates is: ”virus, disease, infection, flu, problem, cough, heat, water”, which is clearly preferring the disease-related sense of ”cold” over the temperature-related sense.



cold	heat	weather	temperature	rain	flu	wind	chill	disease
caught	42.0	0	0	10.8	89.9	454.4	8.6	59.2
nasty	6.3	139.4	0	0	12.0	0	0	27.9
+1 smoothed harmonic mean	12.5	1.98	1	1.84	22.7	1.99	1.81	39.05
rank	3				2			1

Figure 1: Contextualized ranking for target ”cold” in the sentence ”I caught a nasty cold” for the 10 most similar terms from the DT.

In this way, each candidate t′ gets as many scores as there are pairs containing c′ in the holing system output. An overall score per t' then given by the harmonic mean of the add-one-smoothed single scores – smoothing is necessary to rank candidates t′ that are not compatible to all pairs. This scheme can easily be extended to expand all words in a given sentence or paragraph, yielding a two-dimensional contextualized text, where every (content) word is expanded by a list of globally similar words from the distributional thesaurus that are ranked according to their compatibility with the given context.

4 Evaluation

The evaluation of contextualizing the thesaurus (CT) was performed using the LexSub dataset, introduced in the Lexical Substitution task at Semeval 2007 (McCarthy and Navigli, 2009). Following the setup provided by the task organizers, we tuned our approach on the 300 trial sentences, and evaluate it on the official remaining 1710 test sentences. For the evaluation we used the out of ten (oot) precision and oot mode precision. Both measures calculate the number of detected substitutions within ten guesses over the complete subset. Whereas entries in the oot precision measures are considered correct if they match the gold standard, without penalizing non-matching entries, the oot mode precision includes also a weighting as given in the gold standard[1]. For comparison, we use the results of the DT as a baseline to evaluate the contextualization. The DT was computed based on newspaper corpora (120 million sentences), taken from the Leipzig Corpora Collection (Richter et al., 2006) and the Gigaword corpus (Parker et al., 2011). Our holing system uses collapsed Stanford parser dependencies (Marneffe et al., 2006) as context features. The contextualization uses only context features that contain words with part-of-speech prefixes V,N,J,R. Furthermore, we use a threshold for the significance value of the LMI values of 50.0, p=1000, and the most similar 30 terms from the DT entries.

5 Results

Since out contextualization algorithm is dependent on the number of context features containing the target word, we report scores for targets with at least two and at least three dependencies separately. In the Lexical Substitution Task 2007 dataset (LexSub) test data we detected 8 instances without entries in the gold standard and 19 target words without any

[1] The oot setting was chosen because it matches the expansions task better than e.g. precision@1

dependency, as they are collapsed into the dependency relation. The remaining entries have at least one, 49.2% have at least two and 26.0% have at least three dependencies. Furthermore, we also evaluated the results broken down into separate part-of-speeches of the target. The results on the LexSub test set are shown in Table 1.

		Precision			Mode Precision		
min. # dep.		1	2	3	1	2	3
POS	Alg.						
noun	CT	**26.64**	**26.55**	**28.36**	**38.68**	**38.24**	**37.68**
noun	DT	25.35	25.09	28.07	34.96	34.31	36.23
verb	CT	**23.39**	**23.75**	**23.05**	**32.05**	**33.09**	**33.33**
verb	DT	22.46	22.13	21.32	29.17	28.78	28.25
adj.	CT	**32.65**	**34.75**	**36.08**	**45.09**	**48.24**	**46.43**
adj.	DT	32.13	33.25	35.02	43.56	43.53	42.86
adv.	CT	20.47	**29.46**	**36.23**	30.14	**40.63**	**100.00**
adv.	DT	**28.91**	26.75	29.88	**41.63**	34.38	66.67
ALL	CT	26.46	**26.43**	**26.61**	**37.21**	**37.40**	**37.38**
ALL	DT	**27.06**	24.83	25.24	36.96	33.06	33.11

Table 1: Results of the LexSub test dataset.

Inspecting the results for all POS (denoted as ALL), we only observe a slight decline for the precision score with at least only one dependency, which is caused by adverbs. For targets with more than one dependency, we observe overall improvements of 1.6 points in precision and more than 4 points in mode precision.

Regarding the results of different part-of-speech tags, we always improve over the DT ranking, except for adverbs with only one dependency. Most notably, the largest relative improvements are observed on verbs, which is a notoriously difficult word class in computational semantics. For adverbs, at least two dependencies seem to be needed; there are only 7 adverb occurrences with more than two dependencies in the dataset. Regarding performance on the original lexical substitution task (McCarthy and Navigli, 2009), we did not come close to the performance of the participating systems, which range between 32–50 precision points, respectively 43–66 mode precision points (only taking systems without duplicate words in the result set into account). However, all participants used one or several lexical resources for generating substitution candidates, as well as a large number of features. Our system, on the other hand, merely requires a holing system – in this case based on a dependency parser – and a large amount of unlabeled text, and a very small number of contextual clues.

For an insight of the coverage for the entries delivered by the DT graph, we extended the oot precision measure, to consider not only the first 10 entries, but the first X={1,10,50,100,200} entries (see Figure 2). Here we also show the coverage for different sized datasets (10 and 120 million sentences). Amongst the 200 most similar words from the DT, a coverage of up to 55.89 is reached. DT quality improves with corpus size, especially due to increased coverage. This shows that there is considerable headroom for optimization for our contextualization method, but also shows that our automatic candidate expansions can provide a coverage that is competitive to lexical resources.

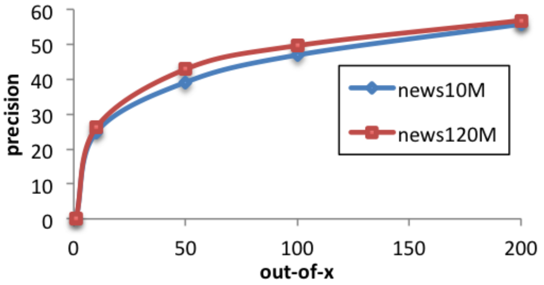


Figure 2: Coverage on the LexSub test dataset for different DT graphs, using *out of X* entries.

6 Conclusion

We have provided a way of operationalizing semantic similarity by splitting syntagmatic observations into terms and context features, and representing them a first-order and second-order graph. Then, we introduced a conceptually simple and efficient method to perform a contextualization of semantic similarity. Overall, our approach constitutes an unsupervised generative model for lexical expansion in context. We have presented a generic method on contextualizing distributional information, which retrieves the lexical expansions from a target term from the DT graph, and ranks them with respect to their context compatibility. Evaluating our method on the LexSub task, we were able to show improvements, especially for expansion targets with many informing contextual elements. For further work, we will extend our holing system and combine several holing systems, such as e.g. n-gram contexts.

Additionally, we would like to adapt more advanced methods for the contextualization (Viterbi, 1967; Lafferty et al., 2001) that yield an all-words simultaneous expansion over the whole sequence, and constitutes a probabilistic model of lexical expansion.

References

M. Baroni and A. Lenci. 2010. Distributional memory: A general framework for corpus-based semantics. *Computational Linguistics*, 36(4):673–721.

Chris Biemann and Martin Riedl. 2013. Text: Now in 2D! a framework for lexical expansion with contextual similarity. *Journal of Language Modelling*, 1(1):55–95.

D. M. Blei, A. Y. Ng, and M. I. Jordan. 2003. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022.

J. Boyd-Graber and D. M. Blei. 2008. Syntactic topic models. In *Neural Information Processing Systems*, Vancouver, BC, USA.

T. Van de Cruys, T. Poibeau, and A. Korhonen. 2013. A tensor-based factorization model of semantic compositionality. In *Proc. NAACL-HLT 2013*, Atlanta, USA.

F. de Saussure. 1916. *Cours de linguistique générale*. Payot, Paris, France.

J. Dean and S. Ghemawat. 2004. MapReduce: Simplified Data Processing on Large Clusters. In *Proc. of Operating Systems, Design & Implementation*, pages 137–150, San Francisco, CA, USA.

S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407.

S. Evert. 2004. *The statistics of word cooccurrences: word pairs and collocations*. Ph.D. thesis, IMS, Universität Stuttgart.

A. Goyal, J. Jagarlamudi, H. Daumé, III, and T. Venkatasubramanian. 2010. Sketch techniques for scaling distributional similarity to the web. In *Proc. of the 2010 Workshop on GEometrical Models of Nat. Lang. Semantics*, pages 51–56, Uppsala, Sweden.

Z. S. Harris. 1951. *Methods in Structural Linguistics*. University of Chicago Press, Chicago, USA.

Thomas Hofmann. 1999. Probabilistic latent semantic indexing. In *Proc. 22nd ACM SIGIR*, pages 50–57, New York, NY, USA.

W. Kintsch. 2001. Predication. *Cognitive Science*, 25(2):173–202.

J. D. Lafferty, A. McCallum, and F. C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. of the 18th Int. Conf. on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA.

L. Lee. 1999. Measures of distributional similarity. In *Proc. of the 37th ACL*, pages 25–32, College Park, MD, USA.

D. Lin. 1998. Automatic retrieval and clustering of similar words. In *Proc. COLING'98*, pages 768–774, Montreal, Quebec, Canada.

M.-C. De Marneffe, B. Maccartney, and C. D. Manning. 2006. Generating typed dependency parses from phrase structure parses. In *Proc. of the Int. Conf. on Language Resources and Evaluation*, Genova, Italy.

D. McCarthy and R. Navigli. 2009. The english lexical substitution task. *Language Resources and Evaluation*, 43(2):139–159.

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

J. Mitchell and M. Lapata. 2008. Vector-based models of semantic composition. In *Proceedings of ACL-08: HLT*, pages 236–244, Columbus, OH, USA.

R. Parker, D. Graff, J. Kong, K. Chen, and K. Maeda. 2011. *English Gigaword Fifth Edition*. Linguistic Data Consortium, Philadelphia, USA.

D. Pucci, M. Baroni, F. Cutugno, and R. Lenci. 2009. Unsupervised lexical substitution with a word space model. In *Workshop Proc. of the 11th Conf. of the Italian Association for Artificial Intelligence*, Reggio Emilia, Italy.

M. Richter, U. Quasthoff, E. Hallsteinsdóttir, and C. Biemann. 2006. Exploiting the leipzig corpora collection. In *Proceesings of the IS-LTC 2006*, Ljubljana, Slovenia.

G. Ruge. 1992. Experiments on linguistically-based term associations. *Information Processing & Management*, 28(3):317 – 332.

P. Rychlý and A. Kilgarriff. 2007. An efficient algorithm for building a distributional thesaurus (and other sketch engine developments). In *Proc. 45th ACL*, pages 41–44, Prague, Czech Republic.

Hinrich Schütze. 1993. Word space. In *Advances in Neural Information Processing Systems 5*, pages 895–902. Morgan Kaufmann.

Ming Tan, Wenli Zhou, Lei Zheng, and Shaojun Wang. 2012. A scalable distributed syntactic, semantic, and lexical language model. *Computational Linguistics*, 38(3):631–671.

P. D. Turney and P. Pantel. 2010. From frequency to meaning: vector space models of semantics. *J. Artif. Int. Res.*, 37(1):141–188.

A. J. Viterbi. 1967. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13(2):260–269.