UNIBA: Distributional Semantics for Textual Similarity

Annalina Caputo **Pierpaolo Basile** **Giovanni Semeraro**
Department of Computer Science
University of Bari "Aldo Moro"
Via E. Orabona, 4 - 70125 Bari, Italy
{acaputo, basilepp, semeraro}@di.uniba.it

Abstract

We report the results of UNIBA participation in the first SemEval-2012 Semantic Textual Similarity task. Our systems rely on distributional models of words automatically inferred from a large corpus. We exploit three different semantic word spaces: Random Indexing (RI), Latent Semantic Analysis (LSA) over RI, and vector permutations in RI. Runs based on these spaces consistently outperform the baseline on the proposed datasets.

1 Background and Related Research

SemEval-2012 Semantic Textual Similarity (STS) task (Agirre et al., 2012) aims at providing a general framework to "*examine the degree of semantic equivalence between two sentences.*"

We propose an approach to Semantic Textual Similarity based on distributional models of words, where the geometrical metaphor of meaning is exploited. Distributional models are grounded on the *distributional hypothesis* (Harris, 1968), according to which the meaning of a word is determined by the set of textual contexts in which it appears. These models represent words as vectors in a high dimensional vector space. Word vectors are built from a large corpus in such a way that vector dimensions reflect the different uses (or *contexts*) of a word in the corpus. Hence, the meaning of a word is defined by its use, and words used in similar contexts are represented by vectors near in the space. In this way, semantically related words like "basketball" and "volleyball", which occur frequently in similar contexts, say with words "court, play, player", will be represented by near points. Different definitions of contexts give rise to different (semantic) spaces. A context can be a document, a sentence or a fixed window of surrounding words. Contexts and words can be stored through a co-occurrence matrix, whose columns correspond to contexts, and rows to words. Therefore, the strength of the semantic association between words can be computed as the cosine similarity of their vector representations.

Latent Semantic Analysis (Deerwester et al., 1990), BEAGLE (Jones and Mewhort, 2007), Random Indexing (Kanerva, 1988), Hyperspace Analogue to Language (Burgess et al., 1998), WordSpace (Schütze and Pedersen, 1995) are all techniques conceived to build up semantic spaces. However, all of them intend to represent semantics at a word scale. Although vectors addition and multiplication are two well defined operations suitable for composing words in semantic spaces, they miss taking into account the underlying syntax, which regulates the compositionality of words. Some efforts toward this direction are emerging (Clark and Pulman, 2007; Clark et al., 2008; Mitchell and Lapata, 2010; Coecke et al., 2010; Basile et al., 2011; Clarke, 2012), which resulted in theoretical work corroborated by empirical evaluation on how small fragments of text compose (e.g. noun-noun, adjective-noun, and verb-noun pairs).

2 Methodology

Our approach to STS is inspired by the latest developments about semantic compositionality and distributional models. The general methodology is based on the construction of a semantic space endowed

*First Joint Conference on Lexical and Computational Semantics (*SEM)*, pages 591–596,
Montréal, Canada, June 7-8, 2012.

with a vector addition operator. The vector addition sums the word vectors of each pair of sentences involved in the evaluation. The result consists of two vectors whose similarity can be computed by cosine similarity. However, this simple methodology translates a text into a mere bag-of-word representation, depriving the text of its syntactic construction, which also influences the overall meaning of the sentence. In order to deal with this limit, we experiment two classical methods for building a semantic space, namely Random Indexing and Latent Semantic Analysis, along with a new method based on vector permutations, which tries to encompass syntactic information directly into the resulting space.

2.1 Random Indexing

Our first method is based on Random Indexing (RI), introduced by Kanerva (Kanerva, 1988). This technique allows us to build a semantic space with no need for (either term-document or term-term) matrix factorization, because vectors are inferred by using an incremental strategy. Moreover, it allows us to solve efficiently the problem of reducing dimensions, which is one of the key features used to uncover the "latent semantic dimensions" of a word distribution.

RI[1] (Widdows and Ferraro, 2008) is based on the concept of Random Projection according to which high dimensional vectors chosen randomly are "nearly orthogonal".

Formally, given an $n \times m$ matrix A and an $m \times k$ matrix R made up of k m-dimensional random vectors, we define a new $n \times k$ matrix B as follows:

$$B^{n,k} = A^{n,m} \cdot R^{m,k} \quad k << m \tag{1}$$

The new matrix B has the property to preserve the distance between points scaled by a multiplicative factor (Johnson and Lindenstrauss, 1984).

Specifically, RI creates the semantic space $B^{n,k}$ in two steps (we consider a fixed window w of terms as context):

1. A *context vector* is assigned to each term. This vector is sparse, high-dimensional and ternary, which means that its elements can take values in {-1, 0, 1}. A context vector contains a small number of randomly distributed non-zero elements, and the structure of this vector follows the hypothesis behind the concept of Random Projection;

2. Context vectors are accumulated by analyzing co-occurring terms in a window w. The *semantic vector* for a term is computed as the sum of the context vectors for terms which co-occur in w.

2.2 Latent Semantic Analysis

Latent Semantic Analysis (Deerwester et al., 1990) relies on the Singular Value Decomposition (SVD) of a term-document co-occurrence matrix. Given a matrix $\mathbf{M}$, it can be decomposed in the product of three matrices $\mathbf{U\Sigma V}^\top$, where $\mathbf{U}$ and $\mathbf{V}$ are the orthonormal matrices and $\mathbf{\Sigma}$ is the diagonal matrix of *singular values* of $\mathbf{M}$ placed in decreasing order. Computing the LSA on the co-occurrence matrix $\mathbf{M}$ can be a computationally expensive task, as a corpus can contain thousands of terms. Hence, we decided to apply LSA to the reduced approximation generated by RI. It is important to point out that no truncation of singular values is performed. Since computing the similarity between any two words is equal to taking the corresponding entry in the $\mathbf{MM}^\top$ matrix, we can exploit the relation

$$\mathbf{MM}^\top = \mathbf{U\Sigma V}^\top \mathbf{V\Sigma}^\top \mathbf{U}^\top = \mathbf{U\Sigma\Sigma}^\top \mathbf{U}^\top = (\mathbf{U\Sigma})(\mathbf{U\Sigma})^\top$$

Hence, the application of LSA to RI makes possible to represent each word in the $\mathbf{U\Sigma}$ space.

A similar approach was investigated by Sellberg and Jönsson (2008) for retrieval of similar FAQs in a Question Answering system. Authors showed that halving the matrix dimension by applying the RI resulted in a drastic reduction of LSA computation time. Certainly there was also a performance price to be paid, however general performance was better than VSM and RI respectively. We also experimented LSA computed on RI versus LSA applied to the original matrix during the tuning of our systems. Surprisingly, we found that LSA applied on the reduced matrix gives better results than LSA. However, these results are not reported as they are not the focus of this evaluation.

[1] An implementation of RI can be found at: http://code.google.com/p/semanticvectors/

2.3 Vector Permutations in RI

The classical distributional models can handle only one definition of context at a time, such as the whole document or the window w. A method to add information about context in RI is proposed in (Sahlgren et al., 2008). The authors describe a strategy to encode word order in RI by the permutation of coordinates in *context vector*. When the coordinates are shuffled using a random permutation, the resulting vector is nearly orthogonal to the original one. That operation corresponds to the generation of a new random vector. Moreover, by applying a predetermined mechanism to obtain random permutations, such as elements rotation, it is always possible to reconstruct the original vector using the reverse permutations. By exploiting this strategy it is possible to obtain different random vectors for each context in which the term occurs.

Our idea is to encode syntactic dependencies using vector permutations. A syntactic dependency between two words is defined as $dep(head, dependent)$, where dep is the syntactic link which connects the $dependent$ word to the $head$ word. Generally speaking, $dependent$ is the modifier, object or complement, while $head$ plays a key role in determining the behavior of the link. For example, $subj(eat, cat)$ means that "cat" is the subject of "eat". In that case the $head$ word is "eat", which plays the role of verb.

The key idea is to encode in the semantic space information about syntactic dependencies which link words together. Rather than representing the kind of dependency, our focus is to encompass information about the existence of such a relation between words in the construction of the space. The method adopted to construct a semantic space that takes into account both syntactic dependencies and Random Indexing can be defined as follows:

1. a context vector is assigned to each term, as described in Section 2.1 (Random Indexing);

2. context vectors are accumulated by analyzing terms which are linked by a dependency. In particular the semantic vector for each term t_i is computed as the sum of the inverse-permuted context vectors for the terms t_j which are dependents of t_i, and the permuted vectors for the terms t_j which are heads of t_i. Moreover, the context vector of t_i, and those of t_j terms which appears in a dependency relation with it, are sum to the final semantic vector in order to provide distributional evidence of co-occurrence. Each permutation is computed as a forward/backward rotation of one element. If Π^1 is a permutation of one element, the inverse-permutation is defined as Π^{-1}: the elements rotation is performed by one left-shifting step. Formally, denoting with $\mathbf{x}$ the context vector for a term, we compute the semantic vector for the term t_i as follows:

$$\mathbf{s_i} = \mathbf{x_i} + \sum_{\substack{j \\ \forall dep(t_i,t_j)}} \left(\Pi^{-1}\mathbf{x_j} + \mathbf{x_j}\right) + \sum_{\substack{k \\ \forall dep(t_k,t_i)}} \left(\Pi^{1}\mathbf{x_k} + \mathbf{x_k}\right)$$

Adding permuted vectors to the head word and inverse-permuted vectors to the corresponding dependent word allows to encode the information about both heads and dependents into the space. This approach is similar to the one investigated by (Cohen et al., 2010) to encode relations between medical terms.

3 Evaluation

Dataset Description. SemEval-2012 STS is a first attempt to provide a "*unified framework for the evaluation of modular semantic components.*" The task consists in computing the similarity between pair of texts, returning a similarity score. Sentences are extracted from five publicly available datasets: MSR (Paraphrase Microsoft Research Paraphrase Corpus, 750 pairs), MSR (Video Microsoft Research Video Description Corpus, 750 pairs), SMTeuroparl (WMT2008 development dataset, Europarl section, 459 pairs), SMTnews (news conversation sentence pairs from WMT, 399 pairs), and OnWN (pairs of sentences from Ontonotes and WordNet definition, 750 pairs). Humans rated each pair with values from 0 to 5. The evaluation is performed by comparing humans scores against systems performance through Pearson's correlation. The organizers propose three different ways to aggregate values from the datasets:

	ALL	Rank-ALL	ALLnrm	Rank-ALLNrm	Mean	Rank-Mean
baseline	*.3110*	*87*	*.6732*	*85*	*.4356*	*70*
UNIBA-RI	.6285	41	.7951	43	.5651	45
UNIBA-LSARI	.6221	44	.8079	30	.5728	40
UNIBA-DEPRI	.6141	46	.8027	38	.5891	31

Table 1: Evaluation results of Pearson's correlation.

	MSRpar	MSRvid	SMT-eur	On-WN	SMT-news
baseline	*.4334*	*.2996*	*.4542*	*.5864*	*.3908*
UNIBA- RI	.4128	.7612	.4531	.6306	**.4887**
UNIBA- LSARI	.3886	**.7908**	.4679	**.6826**	.4238
UNIBA- DEPRI	**.4542**	.7673	**.5126**	.6593	.4636

Table 2: Evaluation results of Pearson's correlation for individual datasets.

ALL Pearson correlation with the gold standard for the five datasets.

ALLnrm Pearson correlation after the system outputs for each dataset are fitted to the gold standard using least squares.

Mean Weighted mean across the five datasets, where the weight depends on the number of pairs in the dataset.

Experimental Setting. For the evaluation, we built Distributional Spaces using the WaCkypedia_EN corpus[2]. WaCkypedia_EN is based on a 2009 dump of the English Wikipedia (about 800 million tokens) and includes information about: part-of-speech, lemma and a full dependency parsing performed by MaltParser (Nivre et al., 2007). The three spaces described in Section 2 are built exploiting information about term windows and dependency parsing supplied by WaCkypedia. The total number of dependencies amounts to about 200 million.

The RI system is implemented in Java and relies on some portions of code publicly available in the Semantic Vectors package (Widdows and Ferraro, 2008), while for LSA we exploited the publicly available C library SVDLIBC[3].

We restricted the vocabulary to the 50,000 most frequent terms, with stop words removal and forcing the system to include terms which occur in the dataset. Hence, the dimension of the original matrix would have been 50,000×50,000.

[2]http://wacky.sslmit.unibo.it/doku.php?id=corpora

[3]http://tedlab.mit.edu/ dr/SVDLIBC/

Our approach involves some parameters. In particular, each semantic space needs to set up the dimension k of the space. All spaces use a dimension of 500 (resulting in a 50,000×500 matrix). The number of non-zero elements in the random vector is set to 10. When we apply LSA to the output space generated by the Random Indexing we hold all the 500 dimensions since during the tuning we observed a drop in performance when a lower dimension was set. The co-occurrence distance w between terms was set up to 4.

In order to compute the similarity between the vector representations of sentences we used the cosine similarity, and then we multiplied by 5 the obtained value.

Results. Table 1 shows the overall results obtained exploiting the different semantic spaces. We report the three proposed evaluation measures with the corresponding overall ranks with respect to the 89 runs submitted by participants. We submitted three different runs, each exploring a different semantic space: UNIBA-RI (based on Random Indexing), UNIBA-LSARI (based on LSA performed over RI outcome), and UNIBA-DEPRI (based on Random Indexing and vector permutations). Each proposed measure stresses different aspects. ALL is the Pearson's correlation computed over the concatenated dataset. As a consequence this measure ranks higher systems which obtain consistent better results. Conversely, ALLNrm normalizes results by scaling values obtained from each dataset, in this way it tries to give emphasis to systems trained on each dataset.

The result of these different perspective is that our three spaces rank differently according to each measure. It seems that UNIBA-RI is able to work better across all datasets, while UNIBA-LSARI gives the best results on specific datasets, even though all our methods are unsupervised and do not need training steps. A deeper analysis on each dataset is reported on Table 2. Here results seem to be at odds with Table 1.

Considering individual datasets, UNIBA-RI gives only once the best result, while UNIBA-LSARI and UNIBA-DEPRI are able to provide the best results twice. Generally, all results outperform the baseline, based on a simple keyword overlap. Lower results are obtained in MSRpar, we ascribe this result to the notably long sentences here involved. In particular, UNIBA-LSARI gives a result lower than the baseline, and in line with the one obtained by LSA during the tuning. Hence, we ascribe this low performance to the application of LSA method to this specific dataset. Only UNIBA-DEPRI was able to outperform the baseline in this dataset. This shows the usefulness of encoding syntactic features in semantic word space where longer sentences are involved. Generally, it is interesting to be noticed that our spaces perform rather well on short and similarly structured sentences, such as MSRvid and On-WN.

4 Conclusion

We reported evaluation results of our participation in Semantic Textual Similarity task. Our systems exploit distributional models to represent the semantics of words. Two of such spaces are based on a classical definition of context, such as a fixed window of surrounding words. A third spaces tries to encompass more definitions of context at once, as the syntactic structure that relates words in a corpus. Although simple, our methods have achieved generally good results, outperforming the baseline provided by the organizers.

References

Eneko Agirre, Daniel Cer, Mona Diab, and Aitor Gonzalez-Agirre. 2012. Semeval-2012 task 6: A pilot on semantic textual similarity. In *Proceedings of the 6th International Workshop on Semantic Evaluation (SemEval 2012), in conjunction with the First Joint Conference on Lexical and Computational Semantics (*SEM 2012)*.

Pierpaolo Basile, Annalina Caputo, and Giovanni Semeraro. 2011. Encoding syntactic dependencies by vector permutation. In *Proceedings of the EMNLP 2011 Workshop on GEometrical Models of Natural Language Semantics*, GEMS '11, pages 43–51, Stroudsburg, PA, USA. Association for Computational Linguistics.

Curt Burgess, Kay Livesay, and Kevin Lund. 1998. Explorations in context space: Words, sentences, discourse. *Discourse Processes*, 25(2-3):211–257.

Stephen Clark and Stephen Pulman. 2007. Combining symbolic and distributional models of meaning. In *Proceedings of the AAAI Spring Symposium on Quantum Interaction*, pages 52–55.

Stephen Clark, Bob Coecke, and Mehrnoosh Sadrzadeh. 2008. A compositional distributional model of meaning. In *Proceedings of the Second Quantum Interaction Symposium (QI-2008)*, pages 133–140.

Daoud Clarke. 2012. A context–theoretic framework for compositionality in distributional semantics. *Computational Linguistics*, 38(1):41–71.

Bob Coecke, Mehrnoosh Sadrzadeh, and Stephen Clark. 2010. Mathematical foundations for a compositional distributional model of meaning. *CoRR*, abs/1003.4394.

Trevor Cohen, Dominic Widdows, Roger W. Schvaneveldt, and Thomas C. Rindflesch. 2010. Logical leaps and quantum connectives: Forging paths through predication space. In *AAAI-Fall 2010 Symposium on Quantum Informatics for Cognitive, Social, and Semantic Processes*, pages 11–13.

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

Zellig Harris. 1968. *Mathematical Structures of Language*. New York: Interscience.

William B. Johnson and Joram Lindenstrauss. 1984. Extensions of Lipschitz mappings into a Hilbert space. *Conference on Modern Analysis and Probability, Contemporary Mathematics*, 26:189–206.

Michael N. Jones and Douglas J. K. Mewhort. 2007. Representing word meaning and order information in a composite holographic lexicon. *Psychological Review*, 114(1):1–37.

Pentti Kanerva. 1988. *Sparse Distributed Memory*. MIT Press.

Jeff Mitchell and Mirella Lapata. 2010. Composition in distributional models of semantics. *Cognitive Science*, 34(8):1388–1429.

Joakim Nivre, Johan Hall, Jens Nilsson, Atanas Chanev, Gülsen Eryigit, Sandra Kübler, Svetoslav Marinov, and Erwin Marsi. 2007. Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(02):95–135.

Magnus Sahlgren, Anders Holst, and Pentti Kanerva. 2008. Permutations as a means to encode order in word space. In V. Sloutsky, B. Love, and K. Mcrae, editors, *Proceedings of the 30th Annual Meeting of the Cognitive Science Society (CogSci'08), July 23-26, Washington D.C., USA*, pages 1300–1305. Cognitive Science Society, Austin, TX.

Hinrich Schütze and Jan O. Pedersen. 1995. Information retrieval based on word senses. In *Proceedings of the 4th Annual Symposium on Document Analysis and Information Retrieval*, pages 161–175.

Linus Sellberg and Arne Jönsson. 2008. Using random indexing to improve singular value decomposition for latent semantic analysis. In Nicoletta Calzolari, Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odjik, Stelios Piperidis, and Daniel Tapias, editors, *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC2008)*, pages 2335–2338, Marrakech, Morocco. European Language Resources Association (ELRA).

Dominic Widdows and Kathleen Ferraro. 2008. Semantic Vectors: A Scalable Open Source Package and Online Technology Management Application. In Nicoletta Calzolari, Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odjik, Stelios Piperidis, and Daniel Tapias, editors, *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC2008)*, pages 1183–1190, Marrakech, Morocco. European Language Resources Association (ELRA).