Inducing Latent Semantic Relations for Structured Distributional Semantics

Sujay Kumar Jauhar
Language Technologies Institute
School of Computer Science
Carnegie Mellon University
sjauhar@cs.cmu.edu

Eduard Hovy
Language Technologies Institute
School of Computer Science
Carnegie Mellon University
hovy@cs.cmu.edu

Abstract

Structured distributional semantic models aim to improve upon simple vector space models of semantics by hypothesizing that the meaning of a word is captured more effectively through its relational — rather than its raw distributional — signature. In accordance, they extend the vector space paradigm by structuring elements with relational information that decompose distributional signatures over discrete relation dimensions. However, the number and nature of these relations remains an open research question, with most previous work in the literature employing syntactic dependencies as surrogates for truly semantic relations. In this paper we propose a novel structured distributional semantic model with latent relation dimensions, and instantiate it using latent relational analysis. Evaluation of our model yields results that significantly outperform several other distributional approaches on two semantic tasks and performs competitively on a third relation classification task.

1 Introduction

The distributional hypothesis, articulated by Firth (1957) in the popular dictum "You shall know the word by the company it keeps", has established itself as one of the most popular models of modern computational semantics. With the rise of massive and easily-accessible digital corpora, computation of co-occurrence statistics has enabled researchers in NLP to build distributional semantic models (DSMs) that have found relevance in many application areas. These include information retrieval (Manning et al., 2008), question answering (Tellex et al., 2003), word-sense disambiguation (McCarthy et al., 2004) and selectional preference modelling (Erk, 2007), to name only a few.

The standard DSM framework, which models the semantics of a word by co-occurrence statistics computed over its neighbouring words, has several known short-comings. One severe short-coming derives from the fundamental nature of the vector space model, which characterizes the semantics of a word by a single vector in a high dimensional space (or some lower dimensional embedding thereof).

Such a modelling paradigm goes against the grain of the intuition that the semantics of a word is neither unique nor constant. Rather, it is composed of many facets of meaning, and similarity (or dissimilarity) to other words is an outcome of the aggregate harmony (or dissonance) between the individual facets under consideration. For example, a shirt may be similar along one facet to a balloon in that they are both coloured blue, at the same time being similar to a shoe along another facet for both being articles of clothing, while being dissimilar along yet another facet to a t-shirt because one is stitched from linen while the other is made from polyester.

Structured distributional semantic models (SDSMs) aim to remedy this fault with DSMs by decomposing distributional signatures over discrete relation dimensions, or facets. This leads to a representation that characterizes the semantics of a word by a distributional tensor, rather than a vector. Previous attempts in the literature include the work of Padó and Lapata (2007), Baroni and Lenci (2010) and Goyal et al. (2013). However, all these approaches assume a simplified representation in which truly semantic relations are substituted by syntactic relations obtained from a dependency parser.

Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers,
pages 698–708, Dublin, Ireland, August 23-29 2014.

We believe that there are limiting factors to this approximation. Most importantly, the set of syntactic relations, while relatively uncontroversial, is unable to capture the full extent of semantic nuance encountered in natural language text. Often, syntax is ambiguous and leads to multiple semantic interpretations. Conversely, passivization and dative shift are common examples of semantic invariance in which multiple syntactic realizations are manifested. Additionally, syntax falls utterly short in explaining more complex phenomena – such as the description of buying and selling – in which implicit semantics are tacit from complex interactions between multiple participants.

While it is useful to consider relations that draw their origins from semantic roles such as Agent, Patient and Recipient, it remains unclear what this set of semantic roles should be. This problem is one that has long troubled linguists (Fillmore, 1967; Sowa, 1991), and has been previously noted by researchers in NLP as well (Màrquez et al., 2008). Proposed solutions range from a small set of generic Agent-like or Patient-like roles in Propbank (Kingsbury and Palmer, 2002) to an effectively open-ended set of highly specific and fine-grained roles in Framenet (Baker et al., 1998). In addition to the theoretic uncertainty of the set of semantic relations there is the very real problem of the lack of high-performance, robust semantic parsers to annotate corpora. These issues effectively render the use of pre-defined, linguistically ordained semantic relations intractable for use in SDSM.

In this paper we propose a novel approach to structuring distributional semantic models with *latent* relations that are automatically discovered from corpora. This approach effectively solves the conceptual dilemma of selecting the most expressive set of semantic relations. To the best of our knowledge this is the first paper to propose latent relation dimensions for SDSMs. The intuition for generating these latent relation dimensions leads to a generic framework, which — in this paper — is instantiated with embeddings obtained from latent relational analysis (Turney, 2005).

We conduct experiments on three different semantic tasks to evaluate our model. On a similarity scoring task and another synonym ranking task the model significantly outperforms other distributional semantic models, including a standard window-based model, a syntactic SDSM based on previous approaches proposed in the literature, and a state-of-the-art semantic model trained using recursive neural networks. On a relation classification task, our model performs competitively, outperforming all but one of the models it is compared against.

2 Related Work

Since the distributional hypothesis was first proposed by Firth (1957), a number of different research initiatives have attempted to extend and improve the standard distributional vector space model of semantics. Insensitivity to the multi-faceted nature of semantics has been one of the focal points of several papers. Earlier work in this regard is a paper by Turney (2012), who proposes that the semantics of a word is not obtained along a single distributional axis but simultaneously in two different spaces. He proposes a DSM in which co-occurrence statistics are computed for neighbouring nouns and verbs separately to yield independent domain and function spaces of semantics.

This intuition is taken further by a stance which proposes that a word's semantics is distributionally decomposed over *many* independent spaces – each of which is a unique relation dimension. Authors who have endorsed this perspective are Erk and Padó (2008), Goyal et al. (2013), Reisinger and Mooney (2010) and Baroni and Lenci (2010). Our work relates to these papers in that we subscribe to the multiple space semantics view. However, we crucially differ from them by structuring our semantic space with information obtained from latent semantic relations rather than from a syntactic parser. In this paper the instantiation of the SDSM with latent relation dimensions is obtained using LRA (Turney, 2005), which is an extension of LSA (Deerwester et al., 1990) to induce relational embeddings for pairs of words.

From a modelling perspective, SDSMs characterize the semantics of a word by a distributional tensor. Other notable papers on tensor based semantics or semantics of compositional structures are the simple additive and multiplicative models of Mitchell and Lapata (2009), the matrix-vector neural network approach of Socher et al. (2012), the physics inspired quantum view of semantic composition of Grefenstette and Sadrzadeh (2011) and the tensor-factorization model of Van de Cruys et al. (2013).

A different, partially overlapping strain of research attempts to induce word embeddings using methods from deep learning, yielding state-of-the-art results on a number of different tasks. Notable research papers on this topic are the ones by Collobert et al. (2011), Turian et al. (2010) and Socher et al. (2010).

Other related work to note is the body of research concerned with semantic relation classification, which is one of our evaluation tasks. Research community wide efforts in the SemEval-2007 task 4 (Girju et al., 2007), the SemEval-2010 task 8 (Hendrickx et al., 2009) and the SemEval-2012 task 2 (Jurgens et al., 2012) are notable examples. However, different from our work, most previous attempts at semantic relation classification operate on the basis of feature engineering and contextual cues (Bethard and Martin, 2007).

3 Structured Distributional Semantics and Latent Semantic Relation Induction

In this section we formalize the notion of SDSM as an extension of DSM and present a novel SDSM with latent relation dimensions.

A DSM is a vector space V that contains $|\Sigma|$ elements in $\mathbb{R}^n$, where $\Sigma = \{w_1, w_2, ..., w_k\}$ is a vocabulary of k distinct words. Every vocabulary word w_i has an associated semantic vector $\vec{v_i}$ representing its distributional signature. Each of the n elements of $\vec{v_i}$ is associated with a single dimension of its distribution. This dimension may correspond to another word — that may or may not belong to Σ — or a latent dimension as might be obtained from an SVD projection or an embedding learned via a deep neural network. Additionally, each element in $\vec{v_i}$ is typically a normalized co-occurrence frequency count, a PMI score, or a number obtained from an SVD or RNN transformation. The semantic similarity between two words w_i and w_j in a DSM is the vector distance defined by $cos(\vec{v_i}, \vec{v_j})$ on their associated distributional vectors.

An SDSM is an extension of DSM. Formally, it is a space U that contains $|\Sigma|$ elements in $\mathbb{R}^{d \times n}$, where $\Sigma = \{w_1, w_2, ..., w_k\}$ is a vocabulary of k distinct words. Every vocabulary word w_i has an associated semantic tensor $\vec{\vec{u_i}}$, which is itself composed of d vectors $\vec{u_{i1}}, \vec{u_{i2}}, ..., \vec{u_{id}}$ each having n dimensions. Every vector $\vec{u_{il}} \in \vec{\vec{u_i}}$ represents the distributional signature of the word w_i in a relation (or along a facet) r_l. The d relations of the SDSM may be syntactic, semantic, or latent (as in this paper). The n dimensional relational vector $\vec{u_{il}}$ is configurationally the same as a vector $\vec{v_i}$ of a DSM. This definition of an SDSM closely relates to an alternate view of Distributional Memories (DMs) (Baroni and Lenci, 2010) where the semantic space is a third-order tensor, whose modes are Word × Link × Word.

The semantic similarity between two words w_i and w_j in an SDSM is the similarity function defined by $sim(\vec{\vec{u_i}}, \vec{\vec{u_j}})$ on their associated semantic tensors. We use the following decomposition of the similarity function:

$$sim(\vec{\vec{u_i}}, \vec{\vec{u_j}}) = \frac{1}{d} \sum_{l=1}^{d} cos(\vec{u_{il}}, \vec{u_{jl}}) \tag{1}$$

Mathematically, this corresponds to the ratio of the normalized Frobenius product of the two matrices representing $\vec{\vec{u_i}}$ and $\vec{\vec{u_j}}$ to the number of rows in both matrices. Intuitively it is simply the average relation-wise similarity between the two words w_i and w_j.

3.1 Latent Relation Induction for SDSM

The intuition behind our approach for inducing latent relation dimensions revolves around the simple observation that SDSMs, while representing semantics as distributional signatures over relation dimensions, also effectively encode relational vectors between pairs of words. Our method thus works backwards from this observation — beginning with a relational embedding for pairs of words, that are subsequently transformed to yield an SDSM.

Concretely, given a vocabulary $\Gamma = \{w_1, w_2, ..., w_k\}$ and a list of word pairs of interest from the vocabulary $\Sigma_V \subseteq \Gamma \times \Gamma$, we assume that we have some method for inducing a DSM V' that has a vector representation $\vec{v'_{ij}}$ of length d for every word pair $w_i, w_j \in \Sigma_V$, which intuitively embeds the distributional signature of the relation binding the two words in d latent dimensions. We then construct an SDSM U where $\Sigma_U = \Gamma$. For every word $w_i \in \Gamma$ a tensor $\vec{\vec{u_i}} \in \mathbb{R}^{d \times k}$ is generated. The tensor $\vec{\vec{u_i}}$

has d unique k dimensional vectors $\vec{u_{i1}}, \vec{u_{i2}}, ..., \vec{u_{id}}$. For a given relational vector $\vec{u_{il}}$, the value of the jth element is taken from the lth element of the vector $\vec{v'_{ij}}$ belonging to the DSM V'. If the vector $\vec{v'_{ij}}$ does not exist in V' – as is the case where the pair $w_i, w_j \notin \Sigma_V$ – the value of the jth element of $\vec{u_{il}}$ is set to 0. By applying this mapping to generate semantic tensors for every word in Γ, we are left with an SDSM U that effectively embed latent relation dimensions. From the perspective of DMs we matricize the third-order tensor and perform truncated SVD, before restoring the resulting matrix to a third-order tensor.

3.1.1 Latent Relational Analysis

In what follows, we present our instantiation of this model with an implementation that is based on Latent Relational Analysis (LRA) (Turney, 2005) to generate the DSM V'. While other methods (such as RNNs) are equally applicable in this scenario, we use LRA for its operational simplicity as well as proven efficacy on semantic tasks such as analogy detection. The parameter values we chose in our experiments are not fine-tuned and are guided by recommended values from Turney (2005), or scaled suitably to accommodate the size of Σ_V.

The input to LRA is a vocabulary $\Gamma = \{w_1, w_2, ..., w_k\}$ and a list of word pairs of interest from the vocabulary $\Sigma_V \subseteq \Gamma \times \Gamma$. While one might theoretically consider a large vocabulary with all possible pairs, for computational reasons we restrict our vocabulary to approximately 4500 frequent English words and only consider about 2.5% word pairs with high PMI (as computed on the whole of English Wikipedia) in $\Gamma \times \Gamma$. For each of the word pairs $w_i, w_j \in \Sigma_V$ we extract a list of contexts by querying a search engine indexed over the combined texts of the whole of English Wikipedia and Gigaword corpora (approximately 5.8×10^9 tokens). Suitable query expansion is performed by taking the top 4 synonyms of w_i and w_j using Lin's thesaurus (Lin, 1998). Each of these contexts must contain both w_i, w_j (or appropriate synonyms) and optionally some intervening words, and some words to either side.

Given such contexts, patterns for every word pair are generated by replacing the two target words w_i and w_j with placeholder characters X and Y, and replacing none, some or all of the other words by their associated part-of-speech tag or a wildcard symbol. For example, if w_i and w_j are "eat" and "pasta" respectively, and the queried context is "I eat a bowl of pasta with a fork", one would generate patterns such as "* X * NN * Y IN a *", "* X DT bowl IN Y with DT *", etc. For every word pair, only the 5000 most frequent patterns are stored.

Once the set of all relevant patterns $P = p_1, p_2, ..., p_n$ have been computed a DSM V is constructed. In particular, the DSM constitutes a Σ_V based on the list of word pairs of interest, and every word pair w_i, w_j of interest has an associated vector $\vec{v_{ij}}$. Each element m of the vector $\vec{v_{ij}}$ is a count pertaining to the number of times that the pattern p_m was generated by the word pair w_i, w_j.

3.1.2 SVD Transformation

The resulting DSM V is noisy and very sparse. Two transformations are thus applied to V. Firstly all co-occurrence counts between word pairs and patterns are transformed to PPMI scores (Bullinaria and Levy, 2007). Then given the matrix representation of V — where rows correspond to word pairs and columns correspond to patterns — SVD is applied to yield $V = M\Delta N$. Here M and N are matrices that have unit-length orthogonal columns and Δ is a matrix of singular values. By selecting the d top singular values, we approximate V with a lower dimension projection matrix that reduces noise and compensates for sparseness: $V' = M_d \Delta_d$. This DSM V' in d latent dimensions is precisely the one we then use to construct an SDSM, using the transformation described above.

Since the large number of patterns renders it effectively impossible to store the entire matrix V in memory we use a memory friendly implementation[1] of a multi-pass stochastic algorithm to directly approximate the projection matrix (Halko et al., 2011; Rehurek, 2010). A detailed analysis to see how change in the parameter d effects the quality of the model is presented in section 4.

The optimal SDSM embeddings we trained and used in the experiments detailed below are available for download at `http://www.cs.cmu.edu/~sjauhar/Software_files/LR-SDSM.tar.`

[1] `http://radimrehurek.com/gensim/`

Model	Spearman's ρ
Random	0.000
DSM	0.179
synSDSM	0.315
SENNA	0.510
LR-SDSM (300)	0.567
LR-SDSM (130)	**0.586**

Table 1

Model	Acc.
Random	0.25
DSM	0.28
synSDSM	0.27
SENNA	0.38
LR-SDSM (300)	0.47
LR-SDSM (130)	**0.51**

Table 2

Results on the WS-353 similarity scoring task and the ESL synonym selection task. LRA-SDSM significantly outperforms other structured and non-structured distributional semantic models.

gz. This SDSM contains a vocabulary of 4546 frequent English words with 130 latent relation dimensions.

4 Evaluation

Section 3 has described a method for embedding latent relation dimensions in SDSMs. We now turn to the problem of evaluating these relations within the scope of the distributional paradigm in order to address two research questions: 1) Are latent relation dimensions a viable and empirically competitive solution for SDSM? 2) Does structuring lead to a semantically more expressive model than a non-structured DSM? In order to answer these questions we evaluate our model on two generic semantic tasks and present comparative results against other structured and non-structured distributional models. We show that we outperform all of them significantly, thus answering both research questions affirmatively.

While other research efforts have produced better results on these tasks (Jarmasz and Szpakowicz, 2003; Gabrilovich and Markovitch, 2007; Hassan and Mihalcea, 2011), they are either lexicon or knowledge based, or are driven by corpus statistics that tie into auxiliary resources such as multi-lingual information and structured ontologies like Wikipedia. Hence they are not relevant to our experimental validation, and are consequently ignored in our comparative evaluation.

4.1 Word-Pair Similarity Scoring Task

The first task consists in using a semantic model to assign similarity scores to pairs of words. The dataset used in this evaluation setting is the WS-353 dataset from Finkelstein et al. (2002). It consists of 353 pairs of words along with an averaged similarity score on a scale of 1.0 to 10.0 obtained from 13–16 human judges. Word pairs are presented as-is, without any context. For example, an item in this dataset might be "book, paper → 7.46".

System scores are obtained by using the standard cosine similarity measure between distributional vectors in a non-structured DSM. In the case of a variant of SDSM, these scores can be found by using the cosine-based similarity functions in Equation 1 of the previous section. System generated output scores are evaluated against the gold standard using Spearman's rank correlation coefficient.

4.2 Synonym Selection Task

In the second task, the same set of semantic space representations is used to select the semantically closest word to a target from a list of candidates. The ESL dataset from Turney (2002) is used for this task, and was selected over the slightly larger TOEFL dataset (Landauer and Dumais, 1997). The reason for this choice was because the latter contained more complex vocabulary words — several of which were not present in our simple vocabulary model. The ESL dataset consists of 50 target words that appear with 4 candidate lexical substitutes each. While disambiguating context is also given in this dataset, we discard it in our experiments. An example item in this dataset might be "rug → sofa, ottoman, carpet, hallway", with "carpet" being the most synonym-like candidate to the target.

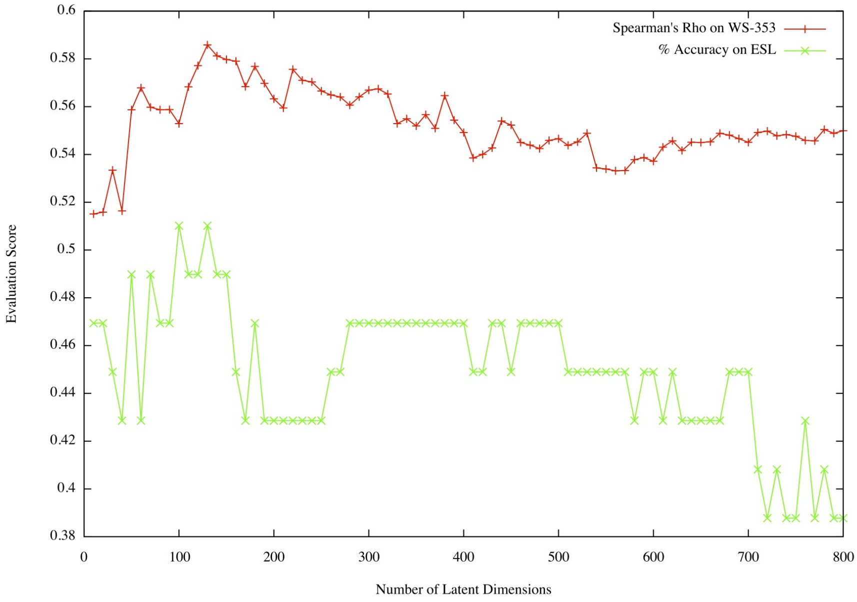

Figure 1: Evaluation results on WS-353 and ESL with varying number of latent dimensions. Generally high scores are obtained in the range of 100-150 latent dimensions, with optimal results on both datasets at 130 latent dimensions.

Similarity scores — which are obtained in the same manner as for the previous evaluation task — are extracted between the target and each of the candidates in turn. These scores are then sorted in descending order, with the top-ranking score yielding the semantically closest candidate to the target. Systems are evaluated on the basis of their accuracy at discriminating the top-ranked candidate.

4.3 Results

We compare our model (LR-SDSM) to several other distributional models in these experiments. These include a standard distributional vector space model (DSM) trained on the combined text of English Wikipedia and Gigaword with a window-size of 3 words to either side of a target, a syntax-based SDSM (Goyal et al., 2013; Baroni and Lenci, 2010) (synSDSM) trained on the same corpus parsed with a dependency parser (Tratz and Hovy, 2011) and the state-of-the-art neural network embeddings from Collobert et al. (2011) (SENNA). We also give the expected evaluation scores from a random baseline, for comparison.

An important factor to consider when constructing an SDSM using LRA is the number of latent dimensions selected in the SVD projection. In Figure 1 we investigate the effects of selecting different number of latent relation dimensions on both semantic evaluation tasks, starting with 10 dimensions up to a maximum of 800 (which was the maximum that was computationally feasible), in increments of 10. We note that optimal results on both datasets are obtained at 130 latent dimensions. In addition to the SDSM obtained in this setting we also give results for an SDSM with 300 latent dimensions (which has been a recommended value for SVD projections in the literature (Landauer and Dumais, 1997)) in our comparisons against other models. Comparative results on the Finkelstein WS-353 similarity scoring task are given in Table 1, while those on the ESL synonym selection task are given in Table 2.

4.4 Discussion

The results in Tables 1 and 2 show that LR-SDSM outperforms the other distributional models by a considerable and statistically significant margin (p-value < 0.05) on both types of semantic evaluation tasks. It should be noted that we do not tune to the test sets. While the 130 latent dimension SDSM yields the best results, 300 latent dimensions also gives comparable performance and moreover outperforms all the other baselines. In fact, it is worth noting that the evaluation results in figure 1 are almost all better

	Random	SENNA-Mik	DSM		SENNA		**LR-SDSM**
			AVC	MVC	AVC	MVC	
Prec.	0.111	0.273	0.419	0.382	**0.489**	0.416	0.431
Rec.	0.110	0.343	0.449	0.443	**0.516**	0.457	0.475
F-1.	0.110	0.288	0.426	0.383	**0.499**	0.429	0.444
% Acc.	11.03	34.30	44.91	44.26	**51.55**	45.65	47.48

Table 3: Results on Relation Classification Task. LR-SDSM scores competitively, outperforming all but the SENNA-AVC model.

than the results of the other models on either datasets.

We conclude that structuring of a semantic model with latent relational information in fact leads to performance gains over non-structured variants. Also, the latent relation dimensions we propose offer a viable and empirically competitive alternative to syntactic relations for SDSMs.

Figure 1 shows the evaluation results on both semantic tasks as a function of the number of latent dimensions. The general trend of both curves on the figure indicate that the expressive power of the model quickly increases with the number of dimensions until it peaks in the range of 100–150, and then decreases or evens out after that. Interestingly, this falls roughly in the range of the 166 frequent (those that appear 50 times or more) frame elements, or fine-grained relations, from FrameNet that O'Hara and Wiebe (2009) find in their taxonomization and mapping of a number of lexical resources that contain semantic relations.

5 Semantic Relation Classification and Analysis of the Latent Structure of Dimensions

In this section we conduct experiments on the task of semantic relation classification. We also perform a more detailed analysis of the induced latent relation dimensions in order to gain insight into our model's perception of semantic relations.

5.1 Semantic Relation Classification

In this task, a relational embedding is used as a feature vector to train a classifier for predicting the semantic relation between previously unseen word pairs. The dataset used in this experiment is from the SemEval-2012 task 2 on measuring the degree of relational similarity (Jurgens et al., 2012), since it characterizes a number of very distinct and interesting semantic relations. In particular it consists of an aggregated set of 3464 word pairs evidencing 10 kinds of semantic relations. We prune this set to discard pairs that don't contain words in the vocabularies of the models we consider in our experiments. This leaves us with a dataset containing 933 word pairs in 9 classes (1 class was discarded altogether because it contained too few instances). The 9 semantic relation classes are: "Class Inclusion", "Part-Whole", "Similar", "Contrast", "Attribute", "Non-Attribute", "Case Relation", "Cause-Purpose" and "Space-Time". For example, an instance of a word pair that exemplifies the "Part-Whole" relationship is "engine:car". Note that, as with previous experiments, word pairs are given without any context.

5.2 Results

We compare LR-SDSM on the semantic relation classification task to several different models. These include the additive vector composition (AVC) and multiplicative vector composition methods (MVC) proposed by Mitchell and Lapata (2009); we present both DSM and SENNA based variants of these models. We also compare against the vector difference method of Mikolov et al. (2013) (SENNA-Mik) which sees semantic relations as a meaning preserving vector translation in an RNN embedded vector space. Finally, we note the performance of random classification as a baseline, for reference. We attempted to produce results of a syntactic SDSM on the task; however, the hard constraint imposed by syntactic adjacency meant that effectively all the word pairs in the dataset yielded zero feature vectors.

To avoid overfitting on all 130 original dimensions in our optimal SDSM, and also to render results comparable, we reduce the number of latent relation dimensions of LR-SDSM to 50. We similarly reduce

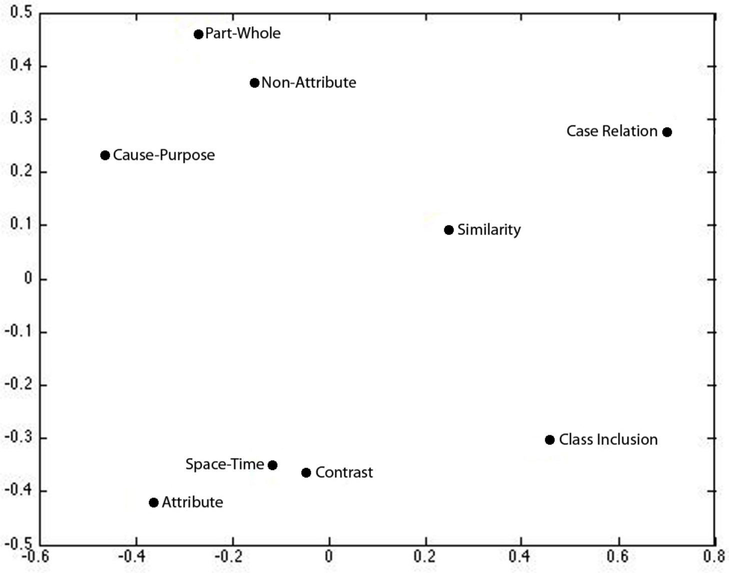

Figure 2: Correlation distances between semantic relations' classifier weights. The plot shows how our latent relations seem to perceive humanly interpretable semantic relations. Most points are fairly well spaced out, with opposites such as "Attribute" and "Non-Attribute" as well as "Similar" and "Contrast" being relatively further apart.

the feature vector dimension of DSM-AVC and DSM-MVC to 50 by feature selection. The dimensions of SENNA-AVC, SENNA-MVC and SENNA-Mik are already 50, and are not reduced further.

For each of the methods we train a logistic regression classifier. We don't perform any tuning of parameters and set a constant ridge regression value of 0.2, which seemed to yield roughly the best results for all models. The performance on the semantic relation classification task in terms of averaged precision, recall, F-measure and percentage accuracy using 10-fold cross-validation is given in Table 3.

Additionally, to gain further insight into the LR-SDSM's understanding of semantic relations, we conduct a secondary analysis. We begin by training 9 one-vs-all logistic regression classifiers for each of the 9 semantic relations under consideration. Then pairwise correlation distances are measured between all pairs of weight vectors of the 9 models. Finally, the distance adjacency matrix is projected into 2-d space using multidimensional scaling. The result of this analysis is presented in Figure 2.

5.3 Discussion

Table 3 shows that LR-SDSM performs competitively on the relation classification task and outperforms all but one of the other models. The performance differences are statistically significant with a p-value < 0.5. We believe that some of the expressive power of the model is lost by compressing to 50 latent relation dimensions, and that a greater number of dimensions might improve performance. However, testing a model with a 130-length dense feature vector on a dataset containing 933 instances would likely lead to overfitting and also not be comparable to the SENNA-based models that operate on 50-length feature vectors.

Other points to note from Table 3 are that the AVC variants of the the DSM and SENNA composition models tend to perform better than their MVC counterparts. Also, SENNA-Mik performs surprisingly poorly. It is worth noting, however, that Mikolov et al. (2013) report results on fairly simple lexico-syntactic relations between words – such as plural forms, possessives and gender – while the semantic relations under consideration in the SemEval-2012 dataset are relatively more complex.

In the analysis of the latent structure of dimensions presented in Figure 2, there are few interesting points to note. To begin with, all the points (with the exception of one pair) are fairly well spaced out. At the weight vector level, this implies that different latent dimensions need to fire in different combinations to characterize distinct semantic relations, thus resulting in low correlation between their corresponding weight vectors. This indicates the fact that the latent relation dimensions seem to capture the intuition that each of the classes encodes a distinctly different semantic relation. The notable exception is “Space-Time”, which is very close to “Contrast”. This is probably due to the fact that distributional models are ineffective at capturing spatio-temporal semantics. Moreover, it is interesting to note that “Attribute" and “Non-Attribute” as well as “Similar” and “Contrast”, which are intuitively semantic inverses of each other are also (relatively) distant from each other in the plot.

These general findings indicate an interesting avenue for future research, which involves mapping the empirically learnt latent relations to hand-built semantic lexicons or frameworks. This could help to validate the empirical models at various levels of linguistic granularity, as well as establish correspondences between different views of semantic representation.

6 Conclusion and Future Work

In this paper we have proposed a novel paradigm for SDSMs, that allows for structuring via latent relational information. We have introduced a generic operational framework that allows for building such SDSMs and outlined an instantiation of the model with LRA. Experimental results of the model support our claim that the resulting SDSM captures the semantics of words more effectively than a number of other semantic models, and presents a viable — and empirically competitive — alternative to syntactic SDSMs. Additionally we have conducted experiments on a relation classification task and shown promising results, as well as performed analyses to investigate the structure of, and interactions between, the latent relation dimensions.

These findings motivate a number of future directions of research. Since our framework is fairly general we hope to explore techniques other than LRA (such as RNNs) to generate relational embeddings for word pairs. A desiderata for future techniques is scalability so that we can characterize vocabularies that are larger than the one in our current experiments. We also hope to explore mappings between our empirically learnt latent relations, and semantic lexicons and frameworks that catalog semantic relations. Finally, we hope to test our model on more realistic application task such as event coreference, recognizing textual entailment, and semantic parsing in future work.

Acknowledgments

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions to improve the quality of the paper. This work was supported in part by the following grants: NSF grant IIS-1143703, NSF award IIS-1147810, DARPA grant FA87501220342.

References

Collin F Baker, Charles J Fillmore, and John B Lowe. 1998. The berkeley framenet project. In *Proceedings of the 17th international conference on Computational linguistics-Volume 1*, pages 86–90. Association for Computational Linguistics.

Marco Baroni and Alessandro Lenci. 2010. Distributional memory: A general framework for corpus-based semantics. *Comput. Linguist.*, 36(4):673–721, December.

Steven Bethard and James H Martin. 2007. Cu-tmp: temporal relation classification using syntactic and semantic features. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 129–132. Association for Computational Linguistics.

John A Bullinaria and Joseph P Levy. 2007. Extracting semantic representations from word co-occurrence statistics: A computational study. *Behavior Research Methods*, 39(3):510–526.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*, 999888:2493–2537, November.

Scott C. Deerwester, Susan T Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. 1990. Indexing by latent semantic analysis. *JASIS*, 41(6):391–407.

Katrin Erk and Sebastian Padó. 2008. A structured vector space model for word meaning in context. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 897–906, Stroudsburg, PA, USA. Association for Computational Linguistics.

Katrin Erk. 2007. A simple, similarity-based model for selectional preferences.

Charles J Fillmore. 1967. The case for case.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2002. Placing search in context: The concept revisited. In *ACM Transactions on Information Systems*, volume 20, pages 116–131, January.

John R. Firth. 1957. A Synopsis of Linguistic Theory, 1930-1955. *Studies in Linguistic Analysis*, pages 1–32.

Evgeniy Gabrilovich and Shaul Markovitch. 2007. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *IJCAI*, volume 7, pages 1606–1611.

Roxana Girju, Preslav Nakov, Vivi Nastase, Stan Szpakowicz, Peter Turney, and Deniz Yuret. 2007. Semeval-2007 task 04: Classification of semantic relations between nominals. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 13–18. Association for Computational Linguistics.

Kartik Goyal, Sujay Kumar Jauhar, Huiying Li, Mrinmaya Sachan, Shashank Srivastava, and Eduard Hovy. 2013. A structured distributional semantic model for event co-reference. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL – 2013)*.

Edward Grefenstette and Mehrnoosh Sadrzadeh. 2011. Experimental support for a categorical compositional distributional model of meaning. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 1394–1404, Stroudsburg, PA, USA. Association for Computational Linguistics.

Nathan Halko, Per-Gunnar Martinsson, and Joel A Tropp. 2011. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review*, 53(2):217–288.

Samer Hassan and Rada Mihalcea. 2011. Semantic relatedness using salient semantic analysis. In *AAAI*.

Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. 2009. Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the Workshop on Semantic Evaluations: Recent Achievements and Future Directions*, pages 94–99. Association for Computational Linguistics.

Mario Jarmasz and Stan Szpakowicz. 2003. Roget's thesaurus and semantic similarity. *Recent Advances in Natural Language Processing III: Selected Papers from RANLP*.

David A Jurgens, Peter D Turney, Saif M Mohammad, and Keith J Holyoak. 2012. Semeval-2012 task 2: Measuring degrees of relational similarity. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, pages 356–364. Association for Computational Linguistics.

Paul Kingsbury and Martha Palmer. 2002. From treebank to propbank. In *LREC*. Citeseer.

Thomas K Landauer and Susan T. Dumais. 1997. A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological review*, pages 211–240.

Dekang Lin. 1998. Automatic retrieval and clustering of similar words. In *Proceedings of the 17th international conference on Computational linguistics-Volume 2*, pages 768–774. Association for Computational Linguistics.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA.

Lluís Màrquez, Xavier Carreras, Kenneth C Litkowski, and Suzanne Stevenson. 2008. Semantic role labeling: an introduction to the special issue. *Computational linguistics*, 34(2):145–159.

Diana McCarthy, Rob Koeling, Julie Weeds, and John Carroll. 2004. Finding predominant word senses in untagged text. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, ACL '04, Stroudsburg, PA, USA. Association for Computational Linguistics.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013. Linguistic regularities in continuous space word representations. *Proceedings of NAACL-HLT*, pages 746–751.

Jeff Mitchell and Mirella Lapata. 2009. Language models based on semantic composition. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1 - Volume 1*, EMNLP '09, pages 430–439, Stroudsburg, PA, USA. Association for Computational Linguistics.

Tom O'Hara and Janyce Wiebe. 2009. Exploiting semantic role resources for preposition disambiguation. *Computational Linguistics*, 35(2):151–184.

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

Radim Rehurek. 2010. Fast and faster: A comparison of two streamed matrix decomposition algorithms. *NIPS 2010 Workshop on Low-rank Methods for Large-scale Machine Learning*.

Joseph Reisinger and Raymond J Mooney. 2010. Multi-prototype vector-space models of word meaning. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 109–117. Association for Computational Linguistics.

Richard Socher, Christopher D Manning, and Andrew Y Ng. 2010. Learning continuous phrase representations and syntactic parsing with recursive neural networks. *Proceedings of the NIPS-2010 Deep Learning and Unsupervised Feature Learning Workshop*.

Richard Socher, Brody Huval, Christopher D. Manning, and Andrew Y. Ng. 2012. Semantic compositionality through recursive matrix-vector spaces. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 1201–1211, Stroudsburg, PA, USA. Association for Computational Linguistics.

John F. Sowa. 1991. Principles of semantic networks. Morgan Kaufmann.

Stefanie Tellex, Boris Katz, Jimmy J. Lin, Aaron Fernandes, and Gregory Marton. 2003. Quantitative evaluation of passage retrieval algorithms for question answering. In *SIGIR*, pages 41–47.

Stephen Tratz and Eduard Hovy. 2011. A fast, accurate, non-projective, semantically-enriched parser. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 1257–1268, Stroudsburg, PA, USA. Association for Computational Linguistics.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 384–394. Association for Computational Linguistics.

Peter D. Turney. 2002. Mining the web for synonyms: Pmi-ir versus lsa on toefl. *CoRR*.

Peter Turney. 2005. Measuring semantic similarity by latent relational analysis. In *Proceedings of the 19th international Conference on Aritifical Intelligence*, pages 1136–1141.

Peter D Turney. 2012. Domain and function: A dual-space model of semantic relations and compositions. *Journal of Artificial Intelligence Research*, 44:533–585.

Tim Van de Cruys, Thierry Poibeau, and Anna Korhonen. 2013. A tensor-based factorization model of semantic compositionality. In *Proceedings of NAACL-HLT*, pages 1142–1151.