Constructing Semantic Space Models from Parsed Corpora

Sebastian Padó
Department of Computational Linguistics
Saarland University
PO Box 15 11 50
66041 Saarbrücken, Germany
pado@coli.uni-sb.de

Mirella Lapata
Department of Computer Science
University of Sheffield
Regent Court, 211 Portobello Street
Sheffield S1 4DP, UK
mlap@dcs.shef.ac.uk

Abstract

Traditional vector-based models use word co-occurrence counts from large corpora to represent lexical meaning. In this paper we present a novel approach for constructing semantic spaces that takes syntactic relations into account. We introduce a formalisation for this class of models and evaluate their adequacy on two modelling tasks: semantic priming and automatic discrimination of lexical relations.

1 Introduction

Vector-based models of word co-occurrence have proved a useful representational framework for a variety of natural language processing (NLP) tasks such as word sense discrimination (Schütze, 1998), text segmentation (Choi et al., 2001), contextual spelling correction (Jones and Martin, 1997), automatic thesaurus extraction (Grefenstette, 1994), and notably information retrieval (Salton et al., 1975). Vector-based representations of lexical meaning have been also popular in cognitive science and figure prominently in a variety of modelling studies ranging from similarity judgements (McDonald, 2000) to semantic priming (Lund and Burgess, 1996; Lowe and McDonald, 2000) and text comprehension (Landauer and Dumais, 1997).

In this approach semantic information is extracted from large bodies of text under the assumption that the context surrounding a given word provides important information about its meaning. The semantic properties of words are represented by vectors that are constructed from the observed distributional patterns of co-occurrence of their neighbouring words. Co-occurrence information is typically collected in a frequency matrix, where each row corresponds to a unique target word and each column represents its linguistic context.

Contexts are defined as a small number of words surrounding the target word (Lund and Burgess, 1996; Lowe and McDonald, 2000) or as entire paragraphs, even documents (Landauer and Dumais, 1997). Context is typically treated as a set of unordered words, although in some cases syntactic information is taken into account (Lin, 1998; Grefenstette, 1994; Lee, 1999). A word can be thus viewed as a point in an n-dimensional semantic space. The semantic similarity between words can be then mathematically computed by measuring the distance between points in the semantic space using a metric such as cosine or Euclidean distance.

In the variants of vector-based models where no linguistic knowledge is used, differences among parts of speech for the same word (e.g., *to drink* vs. *a drink*) are not taken into account in the construction of the semantic space, although in some cases word lexemes are used rather than word surface forms (Lowe and McDonald, 2000; McDonald, 2000). Minimal assumptions are made with respect to syntactic dependencies among words. In fact it is assumed that all context words within a certain distance from the target word are semantically relevant. The lack of syntactic information makes the building of semantic space models relatively straightforward and language independent (all that is needed is a corpus of written or spoken text). However, this entails that contextual information contributes indiscriminately to a word's meaning.

Some studies have tried to incorporate syntactic information into vector-based models. In this view, the semantic space is constructed from words that bear a syntactic relationship to the target word of interest. This makes semantic spaces more flexible, different types of contexts can be selected and words do not have to physically co-occur to be considered contextually relevant. However, existing models either concentrate on specific relations for constructing the semantic space such as objects (e.g., Lee, 1999) or collapse all types of syntactic relations available for a given target word (Grefenstette, 1994; Lin, 1998). Although syntactic information is now used to select a word's appropriate contexts, this information is not explicitly captured in the contexts themselves (which are still represented by words) and is therefore not amenable to further processing.

A commonly raised criticism for both types of semantic space models (i.e., word-based and syntax-based) concerns the notion of semantic similarity. Proximity between two words in the semantic space cannot indicate the nature of the lexical relations between them. Distributionally similar words can be antonyms, synonyms, hyponyms or in some cases semantically unrelated. This limits the application of semantic space models for NLP tasks which require distinguishing between lexical relations.

In this paper we generalise semantic space models by proposing a flexible conceptualisation of context which is parametrisable in terms of syntactic relations. We develop a general framework for vector-based models which can be optimised for different tasks. Our framework allows the construction of semantic space to take place over words or syntactic relations thus bridging the distance between word-based and syntax-based models. Furthermore, we show how our model can incorporate well-defined, informative contexts in a principled way which retains information about the syntactic relations available for a given target word.

We first evaluate our model on semantic priming, a phenomenon that has received much attention in computational psycholinguistics and is typically modelled using word-based semantic spaces. We next conduct a study that shows that our model is sensitive to different types of lexical relations.

2 Dependency-based Vector Space Models

Once we move away from words as the basic context unit, the issue of representation of syntactic information becomes pertinent. Information about the *dependency relations* between words abstracts over word order and can be considered as an intermediate layer between surface syntax and semantics. More formally, dependencies are asymmetric binary relationships between a head and a modifier (Tesnière, 1959). The structure of a sentence can be represented by a set of dependency relationships that form a tree as shown in Figure 1. Here the head of the sentence is the verb *carry* which is in turn modified by its subject *lorry* and its object *apples*.

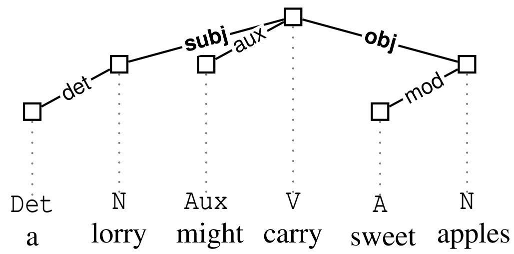

Figure 1: A dependency parse of a short sentence

It is the dependencies in Figure 1 that will form the context over which the semantic space will be constructed. The construction mechanism sets out by identifying the *local context* of a target word, which is a subset of all dependency *paths* starting from it. The paths consist of the dependency *edges* of the tree labelled with dependency relations such as subj, obj, or aux (see Figure 1). The paths can be ranked by a *path value function* which gives different weight to different dependency types (for example, it can be argued that subjects and objects convey more semantic information than determiners). Target words are then represented in terms of *syntactic features* which form the dimensions of the semantic space. Paths are mapped to features by the *path equivalence relation* and the appropriate cells in the matrix are incremented.

2.1 Definition of Semantic Space

We assume the semantic space formalisation proposed by Lowe (2001). A semantic space is a matrix whose rows correspond to target words and columns to dimensions which Lowe calls *basis elements*:

Definition 1. A Semantic Space Model is a matrix $K = B \times T$, where $b_i \in B$ denotes the basis element of column i, $t_j \in T$ denotes the target word of row j, and K_{ij} the cell (i, j).

T is the set of words for which the matrix contains representations; this can be either word *types* or word *tokens*. In this paper, we assume that co-occurrence counts are constructed over word types, but the framework can be easily adapted to represent word tokens instead.

In traditional semantic spaces, the cells K_{ij} of the matrix correspond to word co-occurrence counts. This is no longer the case for dependency-based models. In the following we explain how co-occurrence counts are constructed.

2.2 Building the Context

The first step in constructing a semantic space from a large collection of dependency relations is to construct a word's *local context*.

Definition 2. The *dependency parse* p of a sentence s is an undirected graph $p(s) = (V_p, E_p)$. The set of nodes corresponds to words of the sentence: $V_p = \{w_1, \ldots, w_n\}$. The set of edges is $E_p \subseteq V_p \times V_p$.

Definition 3. A *class* q is a three-tuple consisting of a POS-tag, a relation, and another POS-tag. We write Q for the set of all classes $Cat \times R \times Cat$. For each parse p, the *labelling function* $L_p : E_p \to Q$ assigns a class to every edge of the parse.

In Figure 1, the labelling function labels the left-most edge as $L_p((a, \textit{lorry})) = \langle \texttt{Det}, \textsf{det}, \texttt{N} \rangle$. Note that Det represents the POS-tag "determiner" and det the dependency relation "determiner".

In traditional models, the target words are surrounded by context words. In a dependency-based model, the target words are surrounded by *dependency paths*.

Definition 4. A *path* ϕ is an ordered tuple of edges $\langle e_1, \ldots, e_n \rangle \in E_p^n$ so that

$$\forall i : (e_{i-1} = (v_1, v_2) \wedge e_i = (v_3, v_4)) \Rightarrow v_2 = v_3$$

Definition 5. A *path anchored at a word* w is a path $\langle e_1, \ldots, e_n \rangle$ so that $e_1 = (v_1, v_2)$ and $w = v_1$. Write Φ_w for the set of all paths over E_p anchored at w.

In words, a path is a tuple of connected edges in a parse graph and it is anchored at w if it starts at w. In Figure 1, the set of paths anchored at *lorry*[1] is:

$$\{\langle(lorry, carry)\rangle, \langle(lorry, carry), (carry, apples)\rangle, \\ \langle(lorry, a)\rangle, \langle(lorry, carry), (carry, might)\rangle, \ldots\}$$

The local context of a word is the set or a subset of its anchored paths. The class information can always be recovered by means of the labelling function.

Definition 6. A *local context* of a word w from a sentence s is a subset of the anchored paths at w. A function $c : W \to 2^{\Phi_w}$ which assigns a local context to a word is called a *context specification function*.

[1]For the sake of brevity, we only show paths up to length 2.

The context specification function allows to eliminate paths on the basis of their classes. For example, it is possible to eliminate all paths from the set of anchored paths but those which contain immediate subject and direct object relations. This can be formalised as:

$$c(w) = \{\phi \in \Phi_w \mid \phi = \langle e \rangle \wedge \\ (L_p(e) = \langle \texttt{V}, \textsf{obj}, \texttt{N} \rangle \vee L_p(e) = \langle \texttt{V}, \textsf{subj}, \texttt{N} \rangle)\}$$

In Figure 1, the labels of the two edges which form paths of length 1 and conform to this context specification are marked in boldface. Notice that the local context of *lorry* contains only one anchored path ($c(\textit{lorry}) = \{\langle(lorry, carry)\rangle\}$).

2.3 Quantifying the Context

The second step in the construction of the dependency-based semantic models is to specify the relative importance of different paths. Linguistic information can be incorporated into our framework through the *path value function*.

Definition 7. The *path value function* v assigns a real number to a path: $v : \Phi \to \mathbb{R}$.

For instance, the path value function could penalise longer paths for only expressing indirect relationships between words. An example of a *length-based path value function* is $v(\phi) = \frac{1}{n}$ where $\phi = \langle e_1, \ldots, e_n \rangle$. This function assigns a value of 1 to the one path from $c(\textit{lorry})$ and fractions to longer paths.

Once the value of all paths in the local context is determined, the dimensions of the space must be specified. Unlike word-based models, our contexts contain syntactic information and dimensions can be defined in terms of *syntactic features*. The *path equivalence relation* combines functionally equivalent dependency paths that share a syntactic feature into equivalence classes.

Definition 8. Let $\sim$ be the *path equivalence relation* on Φ. The partition induced by this equivalence relation is the set of basis elements B.

For example, it is possible to combine all paths which end at the same word: A path which starts at w_i and ends at w_j, irrespectively of its length and class, will be the co-occurrence of w_i and w_j. This word-based equivalence function can be defined in the following manner:

$$\langle(v_1, v_2), \ldots, (v_{n-1}, v_n)\rangle \sim \langle(v'_1, v'_2), \ldots, (v'_{m-1}, v'_m)\rangle \\ \text{iff } v_n = v'_m$$

This means that in Figure 1 the set of basis elements is the set of words at which paths end. Although co-occurrence counts are constructed over words like in traditional semantic space models, it is only words which stand in a syntactic relationship to the target that are taken into account.

Once the value of all paths in the local context is determined, the *local observed frequency* for the co-occurrence of a basis element b with the target *word* w is just the sum of values of all paths ϕ in this context which express the basis element b. The *global observed frequency* is the sum of the local observed frequencies for all occurrences of a target word type t and is therefore a measure for the co-occurrence of t and b over the whole corpus.

Definition 9. Global observed frequency:

$$\hat{f}(b,t) = \sum_{w \in W(t)} \sum_{\phi \in C(w) \wedge \phi \sim b} v(\phi)$$

As Lowe (2001) notes, raw frequency counts are likely to give misleading results. Due to the Zipfian distribution of word types, words occurring with similar frequencies will be judged more similar than they actually are. A *lexical association function* can be used to explicitly factor out chance co-occurrences.

Definition 10. Write A for the *lexical association function* which computes the value of a cell of the matrix from a co-occurrence frequency:

$$K_{ij} = A(\hat{f}(b_i,t_j))$$

3 Evaluation

3.1 Parameter Settings

All our experiments were conducted on the British National Corpus (BNC), a 100 million word collection of samples of written and spoken language (Burnard, 1995). We used Lin's (1998) broad coverage dependency parser MINIPAR to obtain a parsed version of the corpus. MINIPAR employs a manually constructed grammar and a lexicon derived from WordNet with the addition of proper names (130,000 entries in total). Lexicon entries contain part-of-speech and subcategorization information. The grammar is represented as a network of 35 nodes (i.e., grammatical categories) and 59 edges (i.e., types of syntactic (dependency) relationships). MINIPAR uses a distributed chart parsing algorithm. Grammar rules are implemented as constraints associated with the nodes and edges.

Cosine distance	$cos(\vec{x},\vec{y}) = \frac{\sum_i x_i y_i}{\sqrt{\sum_i x_i^2}\sqrt{\sum_i y_i^2}}$
Skew divergence	$s_\alpha(\vec{x},\vec{y}) = \sum_i x_i \log \frac{x_i}{\alpha x_i + (1-\alpha) y_i}$

Figure 2: Distance measures

The dependency-based semantic space was constructed with the word-based path equivalence function from Section 2.3. As basis elements for our semantic space the 1000 most frequent words in the BNC were used. Each element of the resulting vector was replaced with its log-likelihood value (see Definition 10 in Section 2.3) which can be considered as an estimate of how surprising or distinctive a co-occurrence pair is (Dunning, 1993).

We experimented with a variety of distance measures such as cosine, Euclidean distance, L_1 norm, Jaccard's coefficient, Kullback-Leibler divergence and the Skew divergence (see Lee 1999 for an overview). We obtained the best results for cosine (Experiment 1) and Skew divergence (Experiment 2). The two measures are shown in Figure 2. The Skew divergence represents a generalisation of the Kullback-Leibler divergence and was proposed by Lee (1999) as a linguistically motivated distance measure. We use a value of $\alpha = .99$.

We explored in detail the influence of different types and sizes of context by varying the context specification and path value functions. Contexts were defined over a set of 23 most frequent dependency relations which accounted for half of the dependency edges found in our corpus. From these, we constructed four context specification functions: (a) `minimum` contexts containing paths of length 1 (in Figure 1 *sweet* and *carry* are the `minimum` context for *apples*), (b) `np` context adds dependency information relevant for noun compounds to `minimum` context, (c) `wide` takes into account paths of length longer than 1 that represent meaningful linguistic relations such as argument structure, but also prepositional phrases and embedded clauses (in Figure 1 the `wide` context of *apples* is *sweet*, *carry*, *lorry*, and *might*), and (d) `maximum` combined all of the above into a rich context representation.

Four path valuation functions were used: (a) *plain* assigns the same value to every path, (b) *length* assigns a value inversely proportional to a path's length, (c) *oblique* ranks paths according to the obliqueness hierarchy of grammatical relations (Keenan and Comrie, 1977), and (d) *oblength*

	context specification	path value function
1	`minimum`	*plain*
2	`minimum`	*oblique*
3	`np`	*plain*
4	`np`	*length*
5	`np`	*oblique*
6	`np`	*oblength*
7	`wide`	*plain*
8	`wide`	*length*
9	`wide`	*oblique*
10	`wide`	*oblength*
11	`maximum`	*plain*
12	`maximum`	*length*
13	`maximum`	*oblique*
14	`maximum`	*oblength*

Table 1: The fourteen models

combines *length* and *oblique*. The resulting 14 parametrisations are shown in Table 1. Length-based and length-neutral path value functions are collapsed for the `minimum` context specification since it only considers paths of length 1.

We further compare in Experiments 1 and 2 our dependency-based model against a state-of-the-art vector-based model where context is defined as a "bag of words". Note that considerable latitude is allowed in setting parameters for vector-based models. In order to allow a fair comparison, we selected parameters for the traditional model that have been considered optimal in the literature (Patel et al., 1998), namely a symmetric 10 word window and the most frequent 500 content words from the BNC as dimensions. These parameters were similar to those used by Lowe and McDonald (2000) (symmetric 10 word window and 536 content words). Again the log-likelihood score is used to factor out chance co-occurrences.

3.2 Experiment 1: Priming

A large number of modelling studies in psycholinguistics have focused on simulating semantic priming studies. The semantic priming paradigm provides a natural test bed for semantic space models as it concentrates on the semantic similarity or dissimilarity between a prime and its target, and it is precisely this type of lexical relations that vector-based models capture.

In this experiment we focus on Balota and Lorch's (1986) mediated priming study. In semantic priming transient presentation of a *prime word* like *tiger* directly facilitates pronunciation or lexical decision on a *target word* like *lion*. Mediated priming extends this paradigm by additionally allowing indirectly related words as primes – like *stripes*, which is only related to *lion* by means of the intermediate concept *tiger*. Balota and Lorch (1986) obtained small mediated priming effects for pronunciation tasks but not for lexical decision. For the pronunciation task, reaction times were reduced significantly for both direct and mediated primes, however the effect was larger for direct primes.

There are at least two semantic space simulations that attempt to shed light on the mediated priming effect. Lowe and McDonald (2000) replicated both the direct and mediated priming effects, whereas Livesay and Burgess (1997) could only replicate direct priming. In their study, mediated primes were farther from their targets than unrelated words.

3.2.1 Materials and Design

Materials were taken form Balota and Lorch (1986). They consist of 48 target words, each paired with a related and a mediated prime (e.g., *lion-tiger-stripes*). Each related-mediated prime tuple was paired with an unrelated control randomly selected from the complement set of related primes.

3.2.2 Procedure

One stimulus was removed as it had a low corpus frequency (less than 100), which meant that the resulting vector would be unreliable. We constructed vectors from the BNC for all stimuli with the dependency-based models and the traditional model, using the parametrisations given in Section 3.1 and cosine as a distance measure. We calculated the distance in semantic space between targets and their direct primes (TarDirP), targets and their mediated primes (TarMedP), targets and their unrelated controls (TarUnC) for both models.

3.2.3 Results

We carried out a one-way Analysis of Variance (ANOVA) with the distance as dependent variable (TarDirP, TarMedP, TarUnC). Recall from Table 1 that we experimented with fourteen different context definitions. A reliable effect of distance was observed for all models ($p < .001$). We used the η^2 statistic to calculate the amount of variance accounted for by the different models. Figure 3 plots η^2 against the different contexts. The best result was obtained for model 7 which accounts for 23.1% of the variance ($F(2,140) = 20.576$, $p < .001$) and corresponds to the `wide` context specification and the *plain* path value function. A reliable distance effect was also observed for the traditional vector-based model ($F(2,138) = 9.384$, $p < .001$).

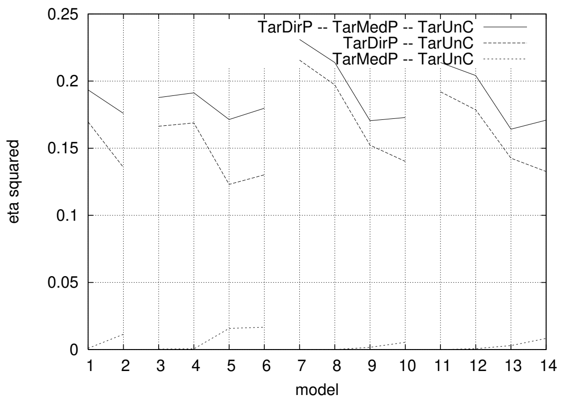

Figure 3: η^2 scores for mediated priming materials

Model	TarDirP – TarUnC	TarMedP – TarUnC
Model 7	$F = 25.290$ ($p < .001$)	$F = .001$ ($p = .790$)
Traditional	$F = 12.185$ ($p = .001$)	$F = .172$ ($p = .680$)
L & McD	$F = 24.105$ ($p < .001$)	$F = 13.107$ ($p < .001$)

Table 2: Size of direct and mediated priming effects

Pairwise ANOVAs were further performed to examine the size of the direct and mediated priming effects individually (see Table 2). There was a reliable direct priming effect ($F(1,94) = 25.290$, $p < .001$) but we failed to find a reliable mediated priming effect ($F(1,93) = .001$, $p = .790$). A reliable direct priming effect ($F(1,92) = 12.185$, $p = .001$) but no mediated priming effect was also obtained for the traditional vector-based model. We used the η^2 statistic to compare the effect sizes obtained for the dependency-based and traditional model. The best dependency-based model accounted for 23.1% of the variance, whereas the traditional model accounted for 12.2% (see also Table 2).

Our results indicate that dependency-based models are able to model direct priming across a wide range of parameters. Our results also show that larger contexts (see models 7 and 11 in Figure 3) are more informative than smaller contexts (see models 1 and 3 in Figure 3), but note that the `wide` context specification performed better than `maximum`. At least for mediated priming, a uniform path value as assigned by the `plain` path value function outperforms all other functions (see Figure 3).

Neither our dependency-based model nor the traditional model were able to replicate the mediated priming effect reported by Lowe and McDonald (2000) (see L & McD in Table 2). This may be due to differences in lemmatisation of the BNC, the parametrisations of the model or the choice of context words (Lowe and McDonald use a special procedure to identify "reliable" context words). Our results also differ from Livesay and Burgess (1997) who found that mediated primes were further from their targets than unrelated controls, using however a model and corpus different from the ones we employed for our comparative studies. In the dependency-based model, mediated primes were virtually indistinguishable from unrelated words.

In sum, our results indicate that a model which takes syntactic information into account outperforms a traditional vector-based model which simply relies on word occurrences. Our model is able to reproduce the well-established direct priming effect but not the more controversial mediated priming effect. Our results point to the need for further comparative studies among semantic space models where variables such as corpus choice and size as well as preprocessing (e.g., lemmatisation, tokenisation) are controlled for.

3.3 Experiment 2: Encoding of Relations

In this experiment we examine whether dependency-based models construct a semantic space that encapsulates different lexical relations. More specifically, we will assess whether word pairs capturing different types of semantic relations (e.g., hyponymy, synonymy) can be distinguished in terms of their distances in the semantic space.

3.3.1 Materials and Design

Our experimental materials were taken from Hodgson (1991) who in an attempt to investigate which types of lexical relations induce priming collected a set of 142 word pairs exemplifying the following semantic relations: (a) synonymy (words with the same meaning, *value* and *worth*), (b) superordination and subordination (one word is an instance of the kind expressed by the other word, *pain* and *sensation*), (c) category coordination (words which express two instances of a common superordinate concept, *truck* and *train*), (d) antonymy (words with opposite meaning, *friend* and *enemy*), (e) conceptual association (the first word subjects produce in free association given the other word, *leash* and *dog*), and (f) phrasal association (words which co-occur in phrases *private* and *property*). The pairs were selected to be unambiguous examples of the relation type they instantiate and were matched for frequency. The pairs cover a wide range of parts of speech, like adjectives, verbs, and nouns.

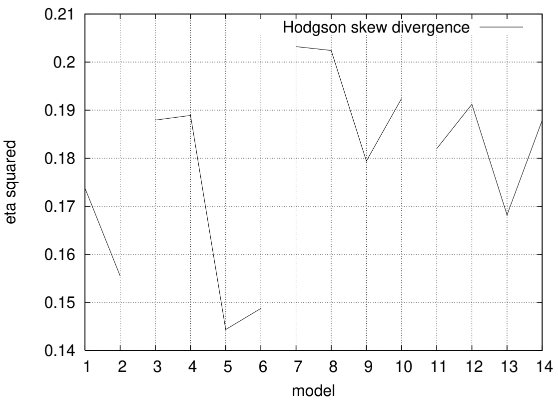

Figure 4: η^2 scores for the Hodgson materials

	Mean	PA	SUP	CO	ANT	SYN
CA	16.25		×	×	×	×
PA	15.13				×	×
SUP	11.04					
CO	10.45					
ANT	10.07					
SYN	8.87					

Table 3: Mean skew divergences and Tukey test results for model 7

3.3.2 Procedure

As in Experiment 1, six words with low frequencies (less than 100) were removed from the materials. Vectors were computed for the remaining 278 words for both the traditional and the dependency-based models, again with the parametrisations detailed in Section 3.1. We calculated the semantic distance for every word pair, this time using Skew divergence as distance measure.

3.3.3 Results

We carried out an ANOVA with the lexical relation as factor and the distance as dependent variable. The lexical relation factor had six levels, namely the relations detailed in Section 3.3.1. We found no effect of semantic distance for the traditional semantic space model ($F(5,141) = 1.481$, $p = .200$). The η^2 statistic revealed that only 5.2% of the variance was accounted for. On the other hand, a reliable effect of distance was observed for all dependency-based models ($p < .001$). Model 7 (`wide` context specification and *plain* path value function) accounted for the highest amount of variance in our data (20.3%). Our results can be seen in Figure 4.

We examined whether there are any significant differences among the six relations using Post-hoc Tukey tests. The pairwise comparisons for model 7 are given in Table 3. The mean distances for conceptual associates (CA), phrasal associates (PA), superordinates/subordinates (SUP), category coordinates (CO), antonyms (ANT), and synonyms (SYN) are also shown in Table 3. There is no significant difference between PA and CA, although SUP, CO, ANT, and SYN, are all significantly different from CA (see Table 3, where × indicates statistical significance, $a = .05$). Furthermore, ANT and SYN are significantly different from PA.

Kilgarriff and Yallop (2000) point out that manually constructed taxonomies or thesauri are typically organised according to synonymy and hyponymy for nouns and verbs and antonymy for adjectives. They further argue that for automatically constructed thesauri similar words are words that either co-occur with each other or with the same words. The relations SYN, SUP, CO, and ANT can be thought of as representing taxonomy-related knowledge, whereas CA and PA correspond to the word clusters found in automatically constructed thesauri. In fact an ANOVA reveals that the distinction between these two classes of relations can be made reliably ($F(1,136) = 15.347$, $p < .001$), after collapsing SYN, SUP, CO, and ANT into one class and CA and PA into another.

Our results suggest that dependency-based vector space models can, at least to a certain degree, distinguish among different types of lexical relations, while this seems to be more difficult for traditional semantic space models. The Tukey test revealed that category coordination is reliably distinguished from all other relations and that phrasal association is reliably different from antonymy and synonymy. Taxonomy related relations (e.g., synonymy, antonymy, hyponymy) can be reliably distinguished from conceptual and phrasal association. However, no reliable differences were found between closely associated relations such as antonymy and synonymy.

Our results further indicate that context encoding plays an important role in discriminating lexical relations. As in Experiment 1 our best results were obtained with the `wide` context specification. Also, weighting schemes such as the obliqueness hierarchy length again decreased the model's performance (see conditions 2, 5, 9, and 13 in Figure 4), showing that dependency relations contribute equally to the representation of a word's meaning. This points to the fact that rich context encodings with a wide range of dependency relations are promising for capturing lexical semantic distinctions. However, the performance for `maximum` context specification was lower, which indicates that collapsing all dependency relations is not the optimal method, at least for the tasks attempted here.

4 Discussion

In this paper we presented a novel semantic space model that enriches traditional vector-based models with syntactic information. The model is highly general and can be optimised for different tasks. It extends prior work on syntax-based models (Grefenstette, 1994; Lin, 1998), by providing a general framework for defining context so that a large number of syntactic relations can be used in the construction of the semantic space.

Our approach differs from Lin (1998) in three important ways: (a) by introducing dependency paths we can capture non-immediate relationships between words (i.e., between subjects and objects), whereas Lin considers only local context (dependency edges in our terminology); the semantic space is therefore constructed solely from isolated head/modifier pairs and their inter-dependencies are not taken into account; (b) Lin creates the semantic space from the set of dependency edges that are relevant for a given word; by introducing dependency labels and the path value function we can selectively weight the importance of different labels (e.g., subject, object, modifier) and parametrize the space accordingly for different tasks; (c) considerable flexibility is allowed in our formulation for selecting the dimensions of the semantic space; the latter can be words (see the leaves in Figure 1), parts of speech or dependency edges; in Lin's approach, it is only dependency edges (features in his terminology) that form the dimensions of the semantic space.

Experiment 1 revealed that the dependency-based model adequately simulates semantic priming. Experiment 2 showed that a model that relies on rich context specifications can reliably distinguish between different types of lexical relations. Our results indicate that a number of NLP tasks could potentially benefit from dependency-based models. These are particularly relevant for word sense discrimination, automatic thesaurus construction, automatic clustering and in general similarity-based approaches to NLP.

References

Balota, David A. and Robert Lorch, Jr. 1986. Depth of automatic spreading activation: Mediated priming effects in pronunciation but not in lexical decision. *Journal of Experimental Psychology: Learning, Memory and Cognition* 12(3):336–45.

Burnard, Lou. 1995. *Users Guide for the British National Corpus*. British National Corpus Consortium, Oxford University Computing Service.

Choi, Freddy, Peter Wiemer-Hastings, and Johanna Moore. 2001. Latent Semantic Analysis for text segmentation. In *Proceedings of EMNLP 2001*. Seattle, WA.

Dunning, Ted. 1993. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics* 19:61–74.

Grefenstette, Gregory. 1994. *Explorations in Automatic Thesaurus Discovery*. Kluwer Academic Publishers.

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

Jones, Michael P. and James H. Martin. 1997. Contextual spelling correction using Latent Semantic Analysis. In *Proceedings of the ANLP 97*.

Keenan, E. and B. Comrie. 1977. Noun phrase accessibility and universal grammar. *Linguistic Inquiry* (8):62–100.

Kilgarriff, Adam and Colin Yallop. 2000. What's in a thesaurus. In *Proceedings of LREC 2000*. pages 1371–1379.

Landauer, T. and S. Dumais. 1997. A solution to Platos problem: the latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review* 104(2):211–240.

Lee, Lillian. 1999. Measures of distributional similarity. In *Proceedings of ACL '99*. pages 25–32.

Lin, Dekang. 1998. Automatic retrieval and clustering of similar words. In *Proceedings of COLING-ACL 1998*. Montréal, Canada, pages 768–511.

Lin, Dekang. 2001. LaTaT: Language and text analysis tools. In J. Allan, editor, *Proceedings of HLT 2001*. Morgan Kaufmann, San Francisco.

Livesay, K. and C. Burgess. 1997. Mediated priming in high-dimensional meaning space: What is "mediated" in mediated priming? In *Proceedings of COGSCI 1997*. Lawrence Erlbaum Associates.

Lowe, Will. 2001. Towards a theory of semantic space. In *Proceedings of COGSCI 2001*. Lawrence Erlbaum Associates, pages 576–81.

Lowe, Will and Scott McDonald. 2000. The direct route: Mediated priming in semantic space. In *Proceedings of COGSCI 2000*. Lawrence Erlbaum Associates, pages 675–80.

Lund, Kevin and Curt Burgess. 1996. Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior Research Methods, Instruments, and Computers* 28:203–8.

McDonald, Scott. 2000. *Environmental Determinants of Lexical Processing Effort*. Ph.D. thesis, University of Edinburgh.

Patel, Malti, John A. Bullinaria, and Joseph P. Levy. 1998. Extracting semantic representations from large text corpora. In *Proceedings of the 4th Neural Computation and Psychology Workshop*. London, pages 199–212.

Salton, G, A Wang, and C Yang. 1975. A vector-space model for information retrieval. *Journal of the American Society for Information Science* 18(613–620).

Schütze, Hinrich. 1998. Automatic word sense discrimination. *Computational Linguistics* 24(1):97–124.

Tesnière, Lucien. 1959. *Elements de syntaxe structurale*. Klincksieck, Paris.