One distributional memory, many semantic spaces

Marco Baroni
University of Trento
Trento, Italy
marco.baroni@unitn.it

Alessandro Lenci
University of Pisa
Pisa, Italy
alessandro.lenci@ilc.cnr.it

Abstract

We propose an approach to corpus-based semantics, inspired by cognitive science, in which different semantic tasks are tackled using the same underlying repository of distributional information, collected once and for all from the source corpus. Task-specific semantic spaces are then built on demand from the repository. A straightforward implementation of our proposal achieves state-of-the-art performance on a number of unrelated tasks.

1 Introduction

Corpus-derived distributional *semantic spaces* have proved valuable in tackling a variety of tasks, ranging from concept categorization to relation extraction to many others (Sahlgren, 2006; Turney, 2006; Padó and Lapata, 2007). The typical approach in the field has been a "local" one, in which each semantic task (or set of closely related tasks) is treated as a separate problem, that requires its own corpus-derived model and algorithms. Its successes notwithstanding, the "one task – one model" approach has also some drawbacks.

From a cognitive angle, corpus-based models hold promise as simulations of how humans acquire and use conceptual and linguistic information from their environment (Landauer and Dumais, 1997). However, the common view in cognitive (neuro)science is that humans resort to a multipurpose *semantic memory*, i.e., a database of interconnected concepts and properties (Rogers and McClelland, 2004), adapting the information stored there to the task at hand. From an engineering perspective, going back to the corpus to train a different model for each application is inefficient and it runs the risk of overfitting the model to a specific task, while losing sight of its adaptivity – a highly desirable feature for any intelligent system. Think, by contrast, of WordNet, a single network of semantic information that has been adapted to all sorts of tasks, many of them certainly not envisaged by the resource creators.

In this paper, we explore a different approach to corpus-based semantics. Our model consists of a *distributional semantic memory* – a graph of weighted links between concepts - built once and for all from our source corpus. Starting from the tuples that can be extracted from this graph, we derive multiple semantic spaces to solve a wide range of tasks that exemplify various strands of corpus-based semantic research: measuring semantic similarity between concepts, concept categorization, selectional preferences, analogy of relations between concept pairs, finding pairs that instantiate a target relation and spotting an alternation in verb argument structure. Given a graph like the one in Figure 1 below, adaptation to all these tasks (and many others) can be reduced to two basic operations: 1) building semantic spaces, as co-occurrence matrices defined by choosing different units of the graph as row and column elements; 2) measuring similarity in the resulting matrix either between specific rows or between a row and an average of rows whose elements share a certain property.

After reviewing some of the most closely related work (Section 2), we introduce our approach (Section 3) and, in Section 4, we proceed to test it in various tasks, showing that its performance is always comparable to that of task-specific methods. Section 5 draws the current conclusions and discusses future directions.

2 Related work

Turney (2008) recently advocated the need for a uniform approach to corpus-based semantic tasks. Turney recasts a number of semantic challenges in terms of relational or analogical similarity. Thus, if an algorithm is able to tackle the latter, it can

Proceedings of the EACL 2009 Workshop on GEMS: GEometical Models of Natural Language Semantics, pages 1–8,
Athens, Greece, 31 March 2009.

also be used to address the former. Turney tests his system in a variety of tasks, obtaining good results across the board. His approach amounts to picking a task (analogy recognition) and reinterpreting other tasks as its particular instances. Conversely, we assume that each task may keep its specificity, and unification is achieved by designing a sufficiently general distributional structure, from which semantic spaces can be generated on demand. Currently, the only task we share with Turney is finding SAT analogies, where his method outperforms ours by a large margin (cf. Section 4.2.1). However, Turney uses a corpus that is 25 times larger than ours, and introduces negative training examples, whereas we dependency-parse our corpus – thus, performance is not directly comparable. Besides the fact that our approach does not require labeled training data like Turney's one, it provides, we believe, a more intuitive measure of taxonomic similarity (taxonomic neighbours are concepts that share similar contexts, rather than concepts that co-occur with patterns indicating a taxonomic relation), and it is better suited to model *productive* semantic phenomena, such as the selectional preferences of verbs with respect to unseen arguments (*eating topinambur* vs. *eating ideas*). Such tasks will require an extension of the current framework of Turney (2008) beyond evidence from the direct co-occurrence of target word pairs.

While our unified framework is, as far as we know, novel, the specific ways in which we tackle the different tasks are standard. Concept similarity is often measured by vectors of co-occurrence with context words that are typed with dependency information (Lin, 1998; Curran and Moens, 2002). Our approach to selectional preference is nearly identical to the one of Padó et al. (2007). We solve SAT analogies with a simplified version of the method of Turney (2006). Detecting whether a pair expresses a target relation by looking at shared connector patterns with model pairs is a common strategy in relation extraction (Pantel and Pennacchiotti, 2008). Finally, our method to detect verb slot similarity is analogous to the "slot overlap" of Joanis et al. (2008) and others. Since we aim at a unified approach, the lack of originality of our task-specific methods should be regarded as a positive fact: our general framework can naturally reproduce, locally, well-tried ad-hoc solutions.

3 Distributional semantic memory

Many different, apparently unrelated, semantic tasks resort to the same underlying information, a "distributional semantic memory" consisting of weighted *concept+link+concept* tuples extracted from the corpus. The *concepts* in the tuples are typically content words. The *link* contains corpus-derived information about how the two words are connected in context: it could be for example a dependency path or a shallow lexico-syntactic pattern. Finally, the *weight* typically derives from co-occurrence counts for the elements in a tuple, rescaled via entropy, mutual information or similar measures. The way in which the tuples are identified and weighted when populating the memory is, of course, of fundamental importance to the quality of the resulting models. However, once the memory has been populated, it can be used to tackle many different tasks, without ever having to go back to the source corpus.

Our approach can be compared with the typical organization of databases, in which multiple alternative "views" can be obtained from the same underlying data structure, to answer different information needs. The data structure is virtually independent from the way in which it is accessed. Similarly, the structure of our repository only obeys to the distributional constraints extracted from the corpus, and it is independent from the ways it will be "queried" to address a specific semantic task. Different tasks can simply be defined by how we split the tuples from the repository into row and column elements of a matrix whose cells are filled by the corresponding weights. Each of these derived matrices represents a particular *view* of distributional memory: we will discuss some of these views, and the tasks they are appropriate for, in Section 4.

Concretely, we used here the web-derived, 2-billion word ukWaC corpus,[1] dependency-parsed with MINIPAR.[2] Focusing for now on modeling noun-to-noun and noun-to-verb connections, we selected the 20,000 most frequent nouns and 5,000 most frequent verbs as target concepts (minus stop lists of very frequent items). We selected as target links the top 30 most frequent direct verb-noun dependency paths (e.g., *kill+obj+victim*), the top 30 preposition-mediated noun-to-noun or

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

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

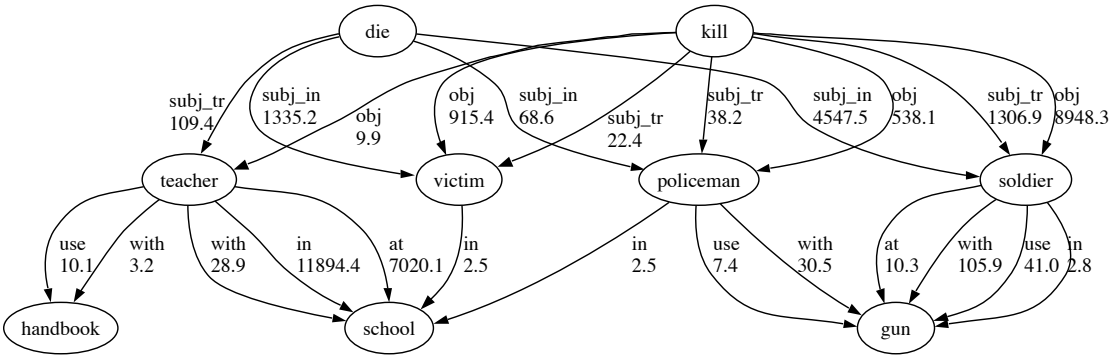

Figure 1: A fragment of distributional memory

verb-to-noun paths (e.g., *soldier+with+gun*) and the top 50 transitive-verb-mediated noun-to-noun paths (e.g., *soldier+use+gun*). We extracted all tuples in which a target link connected two target concepts. We computed the weight (strength of association) for all the tuples extracted in this way using the local MI measure (Evert, 2005), that is theoretically justified, easy to compute for triples and robust against overestimation of rare events. Tuples with local MI $\leq$ 0 were discarded. For each preserved tuple $c1+l+c2$, we added a same-weight $c1 + l^{-1} + c2$ tuple. In graph-theoretical terms (treating concepts as nodes and labeling the weighted edges with links), this means that, for each edge directed from $c1$ to $c2$, there is an edge from $c2$ to $c1$ with the same weight and inverse label, and that such inverse edges constitute the full set of links directed from $c2$ to $c1$. The resulting database (DM, for *Distributional Memory*) contains about 69 million tuples. Figure 1 depicts a fragment of DM represented as a graph (assume, for what we just said, that for each edge from x to y there is a same-weight edge from y to x with inverse label: e.g., the *obj* link from *kill* to *victim* stands for the tuples *kill+obj+victim* and *victim+obj*$^{-1}$*+kill*, both with weight 915.4; *subj_in* identifies the subjects of intransitive constructions, as in *The victim died*; *subj_tr* refers to the subjects of transitive sentences, as in *The policeman killed the victim*).

We also trained 3 closely comparable models that use the same source corpus, the same target concepts (in one case, also the same target links) and local MI as weighting method, with the same filtering threshold. The myPlain model implements a classic "flat" co-occurrence approach (Sahlgren, 2006) in which we keep track of verb-to-noun co-occurrence within a window that can include, maximally, one intervening noun, and noun-to-noun co-occurrence with no more than 2 intervening nouns. The myHAL model uses the same co-occurrence window, but, like HAL (Lund and Burgess, 1996), treats left and right co-occurrences as distinct features. Finally, myDV uses the same dependency-based target links of DM as filters. Like in the DV model of Padó and Lapata (2007), only pairs connected by target links are preserved, but the links themselves are not part of the model. Since none of these alternative models stores information about the links, they are only appropriate for the concept similarity tasks, where links are not necessary.

4 Semantic views and experiments

We now look at three views of the DM graph: *concept-by-link+concept* (CxLC), *concept+concept-by-link* (CCxL), and *concept+link-by-concept* (CLxC). Each view will be tested on one or more semantic tasks and compared with alternative models. There is a fourth possible view, *links-by-concept+concept* (LxCC), that is not explored here, but would lead to meaningful semantic tasks (finding links that express similar semantic relations).

4.1 The CxLC semantic space

Much work in computational linguistics and related fields relies on measuring similarity among words/concepts in terms of their patterns of co-occurrence with other words/concepts (Sahlgren, 2006). For this purpose, we arrange the information from the graph in a matrix where the concepts (nodes) of interest are rows, and the nodes they are connected to by outgoing edges are columns, typed with the corresponding edge label. We refer to this view as the *concept-by-link+concept*

(CxLC) semantic space. From the graph in Figure 1, we can for example construct the matrix in Table 1 (here and below, showing only some rows and columns of interest). By comparing the row vectors of such matrix using standard geometrical techniques (e.g., measuring the normalized cosine distance), we can find out about concepts that tend to share similar properties, i.e., are taxonomically similar (synonyms, antonyms, cohyponyms), e.g., soldiers and policemen, that both kill, are killed and use guns.

	subj_in^{-1} die	subj_tr^{-1} kill	obj^{-1} kill	with gun	use gun
teacher	109.4	0.0	9.9	0.0	0.0
victim	1335.2	22.4	915.4	0.0	0.0
soldier	4547.5	1306.9	8948.3	105.9	41.0
policeman	68.6	38.2	538.1	30.5	7.4

Table 1: A fragment of the CxLC space

We use the CxLC space in three taxonomic similarity tasks: modeling *semantic similarity judgments*, *noun categorization* and *verb selectional restrictions*.

4.1.1 Human similarity ratings

We use the dataset of Rubenstein and Goodenough (1965), consisting of 65 noun pairs rated by 51 subjects on a 0-4 similarity scale (e.g. *car-automobile* 3.9, *cord-smile* 0.0). The average rating for each pair is taken as an estimate of the perceived similarity between the two words. Following Padó and Lapata (2007), we use Pearson's r to evaluate how the distances (cosines) in the CxLC space between the nouns in each pair correlate with the ratings. Percentage correlations for DM, our other models and the best absolute result obtained by Padó and Lapata (DV+), as well as their best cosine-based performance (cosDV+), are reported in Table 2.

model	*r*	*model*	*r*
myDV	70	DV+	62
DM	**64**	myHAL	61
myPlain	63	cosDV+	47

Table 2: Correlation with similarity ratings

DM is the second-best model, outperformed only by DV when the latter is trained on comparable data (myDV in Table 2). Notice that, here and below, we did not try any parameter tuning (e.g., using a similarity measure different than cosine, feature selection, etc.) to improve the performance of DM.

4.1.2 Noun categorization

We use the concrete noun dataset of the ESSLLI 2008 Distributional Semantics shared task,[3] including 44 concrete nouns to be clustered into cognitively justified categories of increasing generality: 6-way (birds, ground animals, fruits, greens, tools and vehicles), 3-way (animals, plants and artifacts) and 2-way (natural and artificial entities). Following the task guidelines, we clustered the target row vectors in the CxLX matrix with CLUTO,[4] using its default settings, and evaluated the resulting clusters in terms of cluster-size-weighted averages of purity and entropy (see the CLUTO documentation). An ideal solution would have 100% purity and 0% entropy. Table 3 provides percentage results for our models as well as for the ESSLLI systems that reported all the relevant performance measures, indexed by first author. Models are ranked by a global score given by summing the 3 purity values and subtracting the 3 entropies.

model	*6-way*		*3-way*		*2-way*		*global*
	P	*E*	*P*	*E*	*P*	*E*	
Katrenko	89	13	100	0	80	59	197
Peirsman+	82	23	84	34	86	55	140
DM	**77**	**24**	**79**	**38**	**59**	**97**	**56**
myDV	80	28	75	51	61	95	42
myHAL	75	27	68	51	68	89	44
Peirsman–	73	28	71	54	61	96	27
myPlain	70	31	68	60	59	97	9
Shaoul	41	77	52	84	55	93	-106

Table 3: Concrete noun categorization

DM outperforms our models trained on comparable resources. Katrenko's system queries Google for patterns that cue the category of a concept, and thus its performance should rather be seen as an upper bound for distributional models. Peirsman and colleagues report results based on different parameter settings: DM's performance – not tuned to the task – is worse than their top model, but better than their worse.

4.1.3 Selectional restrictions

In this task we test the ability of the CxLC space to predict verbal selectional restrictions. We use the CxLC matrix to compare a concept to a "prototype" constructed by averaging a set of other concepts, that in this case represent typical fillers of

[3] http://wordspace.collocations.de/doku.php/esslli:start

[4] http://glaros.dtc.umn.edu/gkhome/cluto/cluto/overview

a verbal slot – for example, by averaging the vectors of the nouns that are, according to the underlying graph, objects of killing, we can build a vector for the typical "killee", and model selectional restrictions by measuring the similarity of other concepts (including concepts that have not been seen as objects of killing in the corpus) to this prototype. Note that the DM graph is used both to find the concepts to enter in the prototype (the set of nouns that are connected to a verb by the relevant edge) and to compute similarity. Thus, the method is fully unsupervised.

We test on the two datasets of human judgments about the plausibility of nouns as arguments (either subjects or objects) of verbs used in Padó et al. (2007), one (McRae) consisting of 100 noun-verb pairs rated by 36 subjects, the second (Padó) with 211 pairs rated by 20 subjects. For each verb in these datasets, we built its prototypical subject/object argument vector by summing the normalized vectors of the 50 nouns with the highest weight on the appropriate dependency link to the verb (e.g., the top 50 nouns connected to *kill* by an *obj* link). The cosine distance of a noun to a prototype is taken as the model "plausibility judgment" about the noun occurring as the relevant verb argument. Since we are interested in generalization, if the target noun is in the prototype set we subtract its vector from the prototype before calculating the cosine. For our comparison models, there is no way to determine which nouns would form the prototype, and thus we train them using the same top noun lists we employ for DM. Following Padó and colleagues, performance is measured by the Spearman ρ correlation coefficient between the average human ratings and the model predictions. Table 4 reports percentage coverage and correlations for our models as well as those in Padó et al. (2007) (ParCos is the best among their purely corpus-based systems).

model	McRae		Padó	
	coverage	ρ	*coverage*	ρ
Padó	56	41	97	51
DM	**96**	**28**	**98**	**50**
ParCos	91	21	98	48
myDV	96	21	98	39
myHAL	96	12	98	29
myPlain	96	12	98	27
Resnik	94	3	98	24

Table 4: Correlation with verb-argument plausibility judgments

DM does very well on this task: its performance on the Padó dataset is comparable to that of the Padó system, that relies on FrameNet. DM has nearly identical performance to the latter on the Padó dataset. On the McRae data, DM has a lower correlation, but much higher coverage. Since we are using a larger corpus than Padó et al. (2007), who train on the BNC, a fairer comparison might be the one with our alternative models, that are all outperformed by DM by a large margin.

4.2 The CCxL semantic space

Another view of the DM graph is exemplified in Table 5, where concept pairs are represented in terms of the edge labels (links) connecting them. Importantly, this matrix contains the same information that was used to build the CxLC space of Table 1, with a different arrangement of what goes in the rows and in the columns, but the same weights in the cells – compare, for example, the *soldier+gun*-by-*with* cell in Table 5 to the *soldier*-by-*with+gun* cell in Table 1.

		in	at	with	use
teacher	school	11894.4	7020.1	28.9	0.0
teacher	handbook	2.5	0.0	3.2	10.1
soldier	gun	2.8	10.3	105.9	41.0

Table 5: A fragment of the CCxL space

We use this space to measure "relational" similarity (Turney, 2006) of concept pairs, e.g., finding that the relation between teachers and handbooks is more similar to the one between soldiers and guns, than to the one between teachers and schools. We also extend relational similarity to prototypes. Given some example pairs instantiating a relation, we can harvest new pairs linked by the same relation by computing the average CCxL vector of the examples, and finding the nearest neighbours to this average. In the case at hand, the link profile of pairs such as *soldier+gun* and *teacher+handbook* could be used to build an "instrument relation" prototype.

We test the CCxL semantic space on *recognizing SAT analogies* (relational similarity between pairs) and *semantic relation classification* (relational similarity to prototypes).

4.2.1 Recognizing SAT analogies

We used the set of 374 multiple-choice questions from the SAT college entrance exam. Each question includes one target pair, usually called

the stem (*ostrich-bird*) , and 5 other pairs (*lion-cat*, *goose-flock*, *ewe-sheep*, *cub-bear*, *primate-monkey*). The task is to choose the pair most analogous to the stem. Each SAT pair can be represented by the corresponding row vector in the CCxL matrix, and we select the pair with the highest cosine to the stem. In Table 6 we report our results, together with the state-of-the-art from the ACL wiki[5] and the scores of Turney (2008) (PairClass) and from Amaç Herdağdelen's PairSpace system, that was trained on ukWaC. The Attr cells summarize the performance of the 6 models on the wiki table that are based on "attributional similarity" only (Turney, 2006). For the other systems, see the references on the wiki. Since our coverage is very low (44% of the stems), in order to make a meaningful comparison with the other models, we calculated a corrected score (DM−). Having full access to the results of the ukWaC-trained, similarly performing PairSpace system, we calculated the adjusted score by assuming that the DM-to-PairSpace error ratio (estimated on the items we cover) is constant on the whole dataset, and thus the DM hit count on the unseen items is approximated by multiplying the PairSpace hit count on the same items by the error ratio (DM+ is DM's accuracy on the covered test items only).

model	*% correct*	*model*	*% correct*
LRA	56.1	KnowBest	43.0
PERT	53.3	**DM−**	**42.3**
PairClass	52.1	LSA	42.0
VSM	47.1	AttrMax	35.0
DM+	**45.3**	AttrAvg	31.0
PairSpace	44.9	AttrMin	27.3
k-means	44.0	Random	20.0

Table 6: Accuracy with SAT analogies

DM does not excel in this task, but its corrected performance is well above chance and that of all the attributional models, and comparable to that of a WordNet-based system (KnowBest) and a system that uses manually crafted information about analogy domains (LSA). All systems with performance above DM+ (and *k*-means) use corpora that are orders of magnitude larger than ukWaC.

4.2.2 Classifying semantic relations

We also tested the CCxL space on the 7 semantic relations between nominals adopted in Task 4 of SEMEVAL 2007 (Girju et al., 2007): Cause-Effect, Instrument-Agency, Product-Producer, Origin-Entity, Theme-Tool, Part-Whole, Content-Container. For each relation, the dataset includes 140 training examples and about 80 test cases. Each example consists of a small context retrieved from the Web, containing word pairs connected by a certain pattern (e..g., "* contains *"). The retrieved contexts were manually classified by the SEMEVAL organizers as positive (e.g., *wrist-arm*) or negative (e.g., *effectiveness-magnesium*) instances of a certain relation (e.g., Part-Whole). About 50% training and test cases are positive instances. For each relation, we built "hit" and "miss" prototype vectors, by averaging across the vectors of the positive and negative training pairs attested in our CCxL model (we use only the word pairs, not the surrounding contexts). A test pair is classified as a hit for a certain relation if it is closer to the hit prototype vector for that relation than to the corresponding miss prototype. We used the SEMEVAL 2007 evaluation method, i.e., precision, recall, F-measure and accuracy, macroaveraged over all relations, as reported in Table 7. The DM+ scores ignore the 32% pairs not in our CCxL space; the DM− scores assume random performance on such pairs. These scores give the range within which our performance will lie once we introduce techniques to deal with unseen pairs. We also report results of the SEMEVAL systems that did not use the organizer-provided WordNet sense labels nor information about the query used to retrieve the examples, as well as performance of several trivial classifiers, also from the SEMEVAL task description.

model	*precision*	*recall*	*F*	*accuracy*
UCD-FC	66.1	66.7	64.8	66.0
UCB	62.7	63.0	62.7	65.4
ILK	60.5	69.5	63.8	63.5
DM+	**60.3**	**62.6**	**61.1**	**63.3**
UMELB-B	61.5	55.7	57.8	62.7
SemeEval avg	59.2	58.7	58.0	61.1
DM−	**56.7**	**58.2**	**57.1**	**59.0**
UTH	56.1	57.1	55.9	58.8
majority	81.3	42.9	30.8	57.0
probmatch	48.5	48.5	48.5	51.7
UC3M	48.2	40.3	43.1	49.9
alltrue	48.5	100.0	64.8	48.5

Table 7: SEMEVAL relation classification

The DM accuracy is higher than the three SEMEVAL baselines (majority, probmatch and alltrue), DM+ is above the average performance of

[5] http://www.aclweb.org/aclwiki/index.php?title=SAT_Analogy_Questions

the comparable SEMEVAL models. Differently from DM, the models that outperform it use features extracted from the training contexts and/or specific additional resources: an annotated compound database for UCD-FC, machine learning algorithms to train the relation classifiers (ILK, UCD-FC), Web counts (UCB), etc. The less than optimal performance by DM is thus counterbalanced by its higher "parsimony" and generality.

4.3 The CLxC semantic space

A third view of the information in the DM graph is the *concept+link-by-concept* (CLxC) semantic space exemplified by the matrix in Table 8.

		teacher	victim	soldier	policeman
kill	subj_tr	0.0	22.4	1306.9	38.2
kill	obj	9.9	915.4	8948.3	538.1
die	subj_in	109.4	1335.2	4547.5	68.6

Table 8: A fragment of the CLxC space

This view captures patterns of similarity between (surface approximations to) argument slots of predicative words. We can thus use the CLxC space to extract generalizations about the inner structure of lexico-semantic representations of the sort formal semanticists have traditionally being interested in. In the example, the patterns of co-occurrence suggest that objects of killing are rather similar to subjects of dying, hinting at the classic *cause(subj,die(obj))* analysis of killing by Dowty (1977) and many others. Again, no new information has been introduced – the matrix in Table 8 is yet another re-organization of the data in our graph (compare, for example, the *die+subj_in*-by-*teacher* cell of this matrix with the *teacher*-by-*subj_in+die* cell in Table 1).

4.3.1 The causative/inchoative alternation

Syntactic alterations (Levin, 1993) represent a key aspect of the complex constraints that shape the syntax-semantics interface. One of the most important cases of alternation is the *causative/inchoative*, in which the object argument (e.g., *John broke the vase*) can also be realized as an intransitive subject (e.g., *The vase broke*). Verbs differ with respect to the possible syntactic alternations they can participate in, and this variation is strongly dependent on their semantic properties (e.g. semantic roles, event type, etc.). For instance, while *break* can undergo the causative/inchoative alternation, *mince* cannot: cf. *John minced the meat* and **The meat minced*.

We test our CLxC semantic space on the discrimination between transitive verbs undergoing the causative-inchoative alternations and non-alternating ones. We took 232 causative/inchoative verbs and 170 non-alternating transitive verbs from Levin (1993). For each verb v_i, we extracted from the CLxC matrix the row vectors corresponding to its transitive subject ($v_i + subj\_tr$), intransitive subject ($v_i + subj\_in$), and direct object ($v_i + obj$) slots. Given the definition of the causative/inchoative alternation, we predict that with alternating verbs $v_i + subj\_in$ should be similar to $v_i + obj$ (the things that are broken also break), while this should not hold for non-alternating verbs (mincees are very different from mincers).

Our model is completely successful in detecting the distinction. The cosine similarity between transitive subject and object slots is fairly low for both classes, as one would expect (medians of 0.16 for alternating verbs and 0.11 for non-alternating verbs). On the other hand, while for the non-alternating verbs the median cosine similarity between the intransitive subject and object slots is a similarly low 0.09, for the alternating verbs the median similarity between these slots jump up to 0.31. Paired t-tests confirm that the per-verb difference between transitive subject vs. object cosines and intransitive subject vs. object cosines is highly statistically significant for the alternating verbs, but not for the non-alternating ones.

5 Conclusion

We proposed an approach to semantic tasks where statistics are collected only once from the source corpus and stored as a set of weighted *concept+link+concept* tuples (naturally represented as a graph). Different semantic spaces are constructed on demand from this underlying "distributional memory", to tackle different tasks without going back to the corpus. We have shown that a straightforward implementation of this approach leads to excellent performance in various taxonomic similarity tasks, and to performance that, while not outstanding, is at least reasonable on relational similarity. We also obtained good results in a task (detecting the causative/inchoative alternation) that goes beyond classic NLP applications and more in the direction of theoretical semantics.

The most pressing issue we plan to address is how to improve performance in the relational sim-

ilarity tasks. Fortunately, some shortcomings of our current model are obvious and easy to fix. The low coverage is in part due to the fact that our set of target concepts does not contain, by design, some words present in the task sets. Moreover, while our framework does not allow ad-hoc optimization of corpus-collection methods for different tasks, the way in which the information in the memory graph is adapted to tasks should of course go beyond the nearly baseline approaches we adopted here. In particular, we need to develop a backoff strategy for unseen pairs in the relational similarity tasks, that, following Turney (2006), could be based on constructing surrogate pairs of taxonomically similar words found in the CxLC space.

Other tasks should also be explored. Here, we viewed our distributional memory in line with how cognitive scientists look at the semantic memory of healthy adults, i.e., as an essentially stable long term knowledge repository. However, much interesting semantic action takes place when underlying knowledge is adapted to context. We plan to explore how contextual effects can be modeled in our framework, focusing in particular on how composition affects word meaning (Erk and Padó, 2008). Similarity could be measured directly on the underlying graph, by relying on graph-based similarity algorithms – an elegant approach that would lead us to an even more unitary view of what distributional semantic memory is and what it does. Alternatively, DM could be represented as a three-mode tensor in the framework of Turney (2007), enabling smoothing operations analogous to singular value decomposition.

Acknowledgments

We thank Ken McRae and Peter Turney for providing data-sets, Amaç Herdağdelen for access to his results, Katrin Erk for making us look at DM as a graph, and the reviewers for helpful comments.

References

J. Curran and M. Moens. 2002. Improvements in automatic thesaurus extraction. *Proceedings of the ACL Workshop on Unsupervised Lexical Acquisition*, 59–66.

D. Dowty. 1977. *Word meaning and Montague Grammar*. Kluwer, Dordrecht.

K. Erk and S. Padó. 2008. A structured vector space model for word meaning in context. *Proceedings of EMNLP 2008*.

S. Evert. 2005. *The statistics of word cooccurrences*. Ph.D. dissertation, Stuttgart University, Stuttgart.

R. Girju, P. Nakov, V. Nastase, S. Szpakowicz, P. Turney and Y. Deniz. 2007. SemEval-2007 task 04: Classification of semantic relations between nominals. *Proceedings of SemEval-2007*, 13–18.

E. Joanis, S. Stevenson and D. James. 2008. A general feature space for automatic verb classification. *Natural Language Engineering*, 14(3): 337–367.

T.K. Landauer and S.T. Dumais. 1997. A solution to Plato's problem: The Latent Semantic Analysis theory of acquisition, induction and representation of knowledge. *Psychological Review*, 104(2): 211–240.

B. Levin. 1993. *English Verb Classes and Alternations. A Preliminary Investigation*. Chicago, University of Chicago Press.

D. Lin. 1998. Automatic retrieval and clustering of similar words. *Proceedings of ACL 1998*, 768–774.

K. Lund and C. Burgess. 1996. Producing high-dimensional semantic spaces from lexical cooccurrence. *Behaviour Research Methods*, 28: 203–208.

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

S. Padó, S. Padó and K. Erk. 2007. Flexible, corpus-based modelling of human plausibility judgements. *Proceedings EMNLP 2007*, 400–409.

P. Pantel and M. Pennacchiotti. 2008. Automatically harvesting and ontologizing semantic relations. In P. Buitelaar and Ph. Cimiano (eds.), *Ontology learning and population*. IOS Press, Amsterdam.

T. Rogers and J. McClelland. 2004. *Semantic cognition: A parallel distributed processing approach*. The MIT Press, Cambridge.

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

M. Sahlgren. 2006. *The Word-space model*. Ph.D. dissertation, Stockholm University, Stockholm.

P. Turney. 2006. Similarity of semantic relations. *Computational Linguistics*, 32(3): 379–416.

P. Turney. 2007. Empirical evaluation of four tensor decomposition algorithms. *IIT Technical Report ERB-1152*, National Research Council of Canada, Ottawa.

P. Turney. 2008. A uniform approach to analogies, synonyms, antonyms and associations. *Proceedings of COLING 2008*, 905–912.