Probabilistic Modeling of Joint-context in Distributional Similarity

Oren Melamud[§], **Ido Dagan**[§], **Jacob Goldberger**[♢], **Idan Szpektor**[†], **Deniz Yuret**[‡]
§ Computer Science Department, Bar-Ilan University
♢ Faculty of Engineering, Bar-Ilan University
† Yahoo! Research Israel
‡ Koç University
{melamuo,dagan,goldbej}@{cs,cs,eng}.biu.ac.il
idan@yahoo-inc.com, dyuret@ku.edu.tr

Abstract

Most traditional distributional similarity models fail to capture syntagmatic patterns that group together multiple word features within the same joint context. In this work we introduce a novel generic distributional similarity scheme under which the power of probabilistic models can be leveraged to effectively model joint contexts. Based on this scheme, we implement a concrete model which utilizes probabilistic n-gram language models. Our evaluations suggest that this model is particularly well-suited for measuring similarity for verbs, which are known to exhibit richer syntagmatic patterns, while maintaining comparable or better performance with respect to competitive baselines for nouns. Following this, we propose our scheme as a framework for future semantic similarity models leveraging the substantial body of work that exists in probabilistic language modeling.

1 Introduction

The Distributional Hypothesis is commonly phrased as "words which are similar in meaning occur in similar contexts" (Rubenstein and Goodenough, 1965). Distributional similarity models following this hypothesis vary in two major aspects, namely the representation of the context and the respective computational model. Probably the most prominent class of distributional similarity models represents context as a vector of word features and computes similarity using feature vector arithmetics (Lund and Burgess, 1996; Turney et al., 2010). To construct the feature vectors, the context of each target word token[1], which is commonly a word window around it, is first broken into a set of individual independent words. Then the weights of the entries in the word feature vector capture the degree of association between the target word type and each of the individual word features, *independently* of one another.

Despite its popularity, it was suggested that the word feature vector approach misses valuable information, which is embedded in the co-location and inter-relations of words (e.g. word order) within the same context (Ruiz-Casado et al., 2005). Following this motivation, Ruiz-Casado et al. (2005) proposed an alternative *composite-feature* model, later adopted in (Agirre et al., 2009). This model adopts a richer context representation by considering entire word window contexts as features, while keeping the same computational vector-based model. Although showing interesting potential, this approach suffers from a very high-dimensional feature space resulting in data sparseness problems. Therefore, it requires exceptionally large learning corpora to consider large windows effectively.

A parallel line of work adopted richer context representations as well, with a different computational model. These works utilized neural networks to learn low dimensional continuous vector representations for word types, which were found useful for measuring semantic similarity (Collobert and Weston, 2008; Mikolov et al., 2013). These vectors are trained by optimizing the prediction of target words given their observed contexts (or variants of this objective). Most of these models consider each observed context as a joint set of context words within a word window.

In this work we follow the motivation in the previous works above to exploit richer *joint-context* representations for modeling distributional similarity. Under this approach the set of features in the context of each target word token is considered to jointly reflect on the meaning of the target word type. To further facilitate this type of mod-

[1] We use *word type* to denote an entry in the vocabulary, and *word token* for a particular occurrence of a word type.

eling we propose a novel probabilistic computational scheme for distributional similarity, which leverages the power of probabilistic models and addresses the data sparseness challenge associated with large joint-contexts. Our scheme is based on the following probabilistic corollary to the distributional hypothesis:

"words are similar in meaning if they are *likely* to occur in the same contexts" (1)

To realize this corollary, our distributional similarity scheme assigns high similarity scores to word pairs a and b, for which a is likely in the contexts that are observed for b and vice versa. The scheme is generic in the sense that various underlying probabilistic models can be used to provide the estimates for the likelihood of a target word given a context. This allows concrete semantic similarity models based on this scheme to leverage the capabilities of probabilistic models, such as established language models, which typically address the modeling of joint-contexts.

We hypothesize that an underlying model that could capture syntagmatic patterns in large word contexts, yet is flexible enough to deal with data sparseness, is desired. It is generally accepted that the semantics of verbs in particular are correlated with their syntagmatic properties (Levin, 1993; Hanks, 2013). This provides grounds to expect that such model has the potential to excel for verbs. To capture syntagmatic patterns, we choose in this work standard n-gram language models as the basis for a concrete model implementing our scheme. This choice is inspired by recent work on learning syntactic categories (Yatbaz et al., 2012), which successfully utilized such language models to represent word window contexts of target words. However, we note that other richer types of language models, such as class-based (Brown et al., 1992) or hybrid (Tan et al., 2012), can be seamlessly integrated into our scheme.

Our evaluations suggest that our model is indeed particularly advantageous for measuring semantic similarity for verbs, while maintaining comparable or better performance with respect to competitive baselines for nouns.

2 Background

In this section we provide additional details regarding previous works that we later use as baselines in our evaluations.

To implement the composite-feature approach, Ruiz-Casado et al. (2005) used a Web search engine to compare entire window contexts of target word types. For example, a single feature that could be retrieved this way for the target word *like* is "Children ___ cookies and milk". They showed good results on detecting synonyms in the 80 multiple-choice questions TOEFL test. Agirre et al. (2009) constructed composite-feature vectors using an exceptionally large 1.6 Teraword learning corpus. They found that this approach outperforms the traditional independent feature vector approach on a subset of the WordSim353 testset (Finkelstein et al., 2001), which is designed to test the more restricted relation of semantic similarity (to be distinguished from looser semantic relatedness). We are not aware of additional works following this approach, of using entire word windows as features.

Neural networks have been used to train language models that are based on low dimensional continuous vector representations for word types, also called *word embeddings* (Bengio et al., 2003; Mikolov et al., 2010). Although originally designed to improve language models, later works have shown that such word embeddings are useful in various other NLP tasks, including measuring semantic similarity with vector arithmetics (Collobert and Weston, 2008; Mikolov et al., 2013). Specifically, the recent work by Mikolov et al. (2013) introduced the *CBOW* and *Skip-gram* models, achieving state-of-the-art results in detecting semantic analogies. The CBOW model is trained to predict a target word given the set of context words in a word window around it, where this context is considered *jointly* as a bag-of-words. The Skip-gram model is trained to predict each of the context words independently given the target word.

3 Probabilistic Distributional Similarity

3.1 Motivation

In this section we briefly demonstrate the benefits of considering joint-contexts of words. As an illustrative example, we note that the target words *like* and *surround* may share many individual word features such as "school" and "campus" in the sentences "Mary's son *likes* the school campus" and "The forest *surrounds* the school campus". This potentially implies that individual features may not be sufficient to accurately reflect the difference

between such words. Alternatively, we could use the following composite features to model the context of these words, "Mary's son ___ the school campus" and "The forest ___ the school campus". This would discriminate better between *like* and *surround*. However, in this case sentences such as "Mary's son *likes* the school campus" and "John's son *loves* the school campus" will not provide any evidence to the similarity between *like* and *love*, since "Mary's son ___ the school campus" is a different feature than "John's son ___ the school campus".

In the remainder of this section we propose a modeling scheme and then a concrete model, which can predict that *like* and *love* are likely to occur in each other's joint-contexts, whereas *like* and *surround* are not, and then assign similarity scores accordingly.

3.2 The probabilistic similarity scheme

We now present a computational scheme that realizes our proposed corollary (1) to the distributional hypothesis and facilitates robust probabilistic modeling of joint contexts. First, we slightly rephrase this corollary as follows: "words a and b are similar in meaning if word b is likely in the contexts of a and vice versa". We denote the probability of an occurrence of a target word b given a joint-context c by $p(b|c)$. For example, p(love|"Mary's son _ the school campus") is the probability of the word *love* to be the filler of the 'place-holder' in the given joint-context "Mary's son _ the school campus". Similarly, we denote $p(c|a)$ as the probability of a joint-context c given a word a, which fills its place-holder. We now propose $p^{sim}(b|a)$ to reflect how likely b is in the joint-contexts of a. We define this measure as:

$$p^{sim}(b|a) = \sum_{c} p(c|a) \cdot p(b|c) \qquad (2)$$

where c goes over all possible joint-contexts in the language.

To implement this measure we need to find an efficient estimate for $p^{sim}(b|a)$. The most straight forward strategy is to compute simple corpus count ratio estimates for $p(b|c)$ and $p(c|a)$, denoted $p_{\#}(b|c) = \frac{count(b,c)}{count(*,c)}$ and $p_{\#}(c|a) = \frac{count(a,c)}{count(a,*)}$. However, when considering large joint-contexts for c, this approach becomes similar to the composite-feature approach since it is based on co-occurrence counts of target words with large joint-contexts. Therefore, we expect in this case to encounter the data sparseness problems mentioned in Section 1, where semantically similar word type pairs that share only few or no identical joint-contexts yield very low $p^{sim}(b|a)$ estimates.

To address the data sparseness challenge and adopt more advanced context modeling, we aim to use a more robust underlying probabilistic model θ for our scheme and denote the probabilities estimated by this model by $p_\theta(b|c)$ and $p_\theta(c|a)$. We note that contrary to the count ratio model, given a robust model θ, such as a language model, $p_\theta(b|c)$ and $p_\theta(c|a)$ can be positive even if the target words b and a were not observed with the joint-context c in the learning corpus.

While using $p_\theta(b|c)$ and $p_\theta(c|a)$ to estimate the value of $p^{sim}(b|a)$ addresses the sparseness challenge, it introduces a computational challenge. This is because estimating $p^{sim}(b|a)$ would require computing the sum over all of the joint-contexts in the learning corpus regardless of whether they were actually observed with either word type a or b. For that reason we choose a middle ground approach, estimating $p(b|c)$ with θ, while using a count ratio estimate for $p(c|a)$, as follows. We denote the collection of all joint-contexts observed for the target word a in the learning corpus by C_a, where $|C_a| = count(a, *)$. For example, C_{like} = {c_1="Mary's son ___ the school campus", c_2="John's daughter ___ to read poetry",...}. We note that this collection is a multi-set, where the same joint-context can appear more than once.

We now approximate $p^{sim}(b|a)$ from Equation (2) as follows:

$$\hat{p}^{sim}_\theta(b|a) = \sum_{c} p_{\#}(c|a) \cdot p_\theta(b|c) = \frac{1}{|C_a|} \cdot \sum_{c \in C_a} p_\theta(b|c) \qquad (3)$$

We note that this formulation still addresses sparseness of data by using a robust model, such as a language model, to estimate $p_\theta(b|c)$. At the same time it requires our model to sum only over the joint-contexts in the collection C_a, since contexts not observed for a yield $p_{\#}(c|a) = 0$. Even so, since the size of these context collections grows linearly with the corpus size, considering all observed contexts may still present a scalability challenge. Nevertheless, we expect our approximation $\hat{p}^{sim}_\theta(b|a)$ to converge with a reasonable sample

size from a's joint-contexts. Therefore, in order to bound computational complexity, we limit the size of the context collections used to train our model to a maximum of N by randomly sampling N entries from larger collections. In all our experiments we use $N = 10,000$. Higher values of N yielded negligible performances differences. Overall we see that our model estimates $\hat{p}_\theta^{sim}(b|a)$ as the average probability predicted for b in (a large sample of) the contexts observed for a.

Finally, we define our similarity measure for target word types a and b:

$$sim_\theta(a,b) = \sqrt{\hat{p}_\theta^{sim}(b|a) \cdot \hat{p}_\theta^{sim}(a|b)} \quad (4)$$

As intended, this similarity measure promotes word pairs in which both b is likely in the contexts of a and vice versa. Next, we describe a model which implements this scheme with an n-gram language model as a concrete choice for θ.

3.3 Probabilistic similarity using language models

In this work we focus on the word window context representation, which is the most common. We define a word window of order k around a target word as a window with up to k words to each side of the target word, not crossing sentence boundaries. The word window does not include the target word itself, but rather a 'place-holder' for it.

Since word windows are sequences of words, probabilistic language models are a natural choice of a model θ for estimating $p_\theta(b|c)$. Language models assign likelihood estimates to sequences of words using approximation strategies. In this work we choose n-gram language models, aiming to capture syntagmatic properties of the word contexts, which are sensitive to word order. To approximate the probability of long sequences of words, n-gram language models compute the product of the estimated probability of each word in the sequence conditioned on at most the $n-1$ words preceding it. Furthermore, they use 'discounting' methods to improve the estimates of conditional probabilities when learning data is sparse. Specifically, in this work we use the Kneser-Ney n-gram model (Kneser and Ney, 1995).

We compute $p_\theta(b|c)$ as follows:

$$p_\theta(b|c) = \frac{p_\theta(b,c)}{p_\theta(c)} \quad (5)$$

where $p_\theta(b,c)$ is the probability of the word sequence comprising the word window c, in which the word b fills the place-holder. For instance, for c = "I drive my ___ to work every" and b = *car*, $p_\theta(b,c)$ is the estimated language model probability of "I drive my car to work every". $p_\theta(c)$ is the marginal probability of $p_\theta(*,c)$ over all possible words in the vocabulary. [2]

4 Experimental Settings

Although sometimes used interchangeably, it is common to distinguish between semantic *similarity* and semantic *relatedness* (Budanitsky and Hirst, 2001; Agirre et al., 2009). Semantic similarity is used to describe 'likeness' relations, such as the relations between synonyms, hypernym-hyponyms, and co-hyponyms. Semantic relatedness refers to a broader range of relations including also meronymy and various other associative relations as in 'pencil-paper' or 'penguin-Antarctica'. In this work we focus on semantic similarity and evaluate all compared methods on several semantic similarity tasks.

Following previous works (Lin, 1998; Riedl and Biemann, 2013) we use Wordnet to construct large scale gold standards for semantic similarity evaluations. We perform the evaluations separately for nouns and verbs to test our hypothesis that our model is particularly well-suited for verbs. To further evaluate our results on verbs we use the verb similarity test-set released by (Yang and Powers, 2006), which contains pairs of verbs associated with semantic similarity scores based on human judgements.

4.1 Compared methods

We compare our model with a traditional feature vector model, the composite-feature model (Agirre et al., 2009), and the recent state-of-the-art word embedding models, CBOW and Skip-gram (Mikolov et al., 2013), all trained on the same learning corpus and evaluated on equal grounds.

We denote the traditional feature vector baseline by IFV^{W-k}, where *IFV* stands for *"Independent-Feature Vector"* and *k* is the order of the context word window considered. Similarly, we

[2] Computing $p_\theta(c)$ by summing over all possible place-holder filler words, as we did in this work, is computationally intensive. However, this can be done more efficiently by implementing customized versions of (at least some) n-gram language models with little computational overhead, e.g. by counting the learning corpus occurrences of n-gram templates, in which one of the elements matches any word.

denote the composite-feature vector baseline by CFV^{W-k}, where *CFV* stands for *"Composite-Feature Vector"*. This baseline constructs traditional-like feature vectors, but considers entire word windows around target word tokens as single features. In both of these baselines we use Cosine as the vector similarity measure, and positive pointwise mutual information (PPMI) for the feature vector weights. PPMI is a well-known variant of pointwise mutual information (Church and Hanks, 1990), and the combination of Cosine with PPMI was shown to perform particularly well in (Bullinaria and Levy, 2007).

We denote Mikolov's CBOW and Skip-gram baseline models by $CBOW^{W-k}$ and $SKIP^{W-k}$ respectively, where *k* denotes again the order of the window used to train these models. We used Mikolov's word2vec utility[3] with standard parameters (600 dimensions, negative sampling 15) to learn the word embeddings, and Cosine as the vector similarity measure between them.

As the underlying probabilistic language model for our method we use the Berkeley implementation[4] (Pauls and Klein, 2011) of the Kneser-Ney n-gram model with the default discount parameters. We denote our model PDS^{W-k}, where *PDS* stands for *"Probabilistic Distributional Similarity"*, and *k* is the order of the context word window. In order to avoid giving our model an unfair advantage of tuning the order of the language model n as an additional parameter, we use a fixed $n = k + 1$. This means that the conditional probabilities that our n-gram model learns consider a scope of up to half the size of the window, which is the distance in words between the target word and either end of the window. We note that this is the smallest reasonable value for n, as smaller values effectively mean that there will be context words within the window that are more than n words away from the target word, and therefore will not be considered by our model.

As learning corpus we used the first CD of the freely available Reuters RCV1 dataset (Rose et al., 2002). This learning corpus contains approximately 100M words, which is comparable in size to the British National Corpus (BNC) (Aston, 1997). We first applied part-of-speech tagging and lemmatization to all words. Then we represented each word w in the corpus as the pair $[pos(w), lemma(w)]$, where $pos(w)$ is a coarse-grained part-of-speech category and $lemma(w)$ is the lemmatized form of w. Finally, we converted every pair $[pos(w), lemma(w)]$ that occurs less than 100 times in the learning corpus to the pair $[pos(w), ?]$, which represents all rare words of the same part-of-speech tag. Ignoring rare words is a common practice used in order to clean up the corpus and reduce the vocabulary size (Gorman and Curran, 2006; Collobert and Weston, 2008).

The above procedure resulted in a word vocabulary of 27K words. From this vocabulary we constructed a *target verb set* with over 2.5K verbs by selecting all verbs that exist in Wordnet (Fellbaum, 2010). We repeated this procedure to create a *target noun set* with over 9K nouns. We used our learning corpus for all compared methods and had them assign a semantic similarity score for every pair of verbs and every pair of nouns in these target sets. These scores were later used in all of our evaluations.

4.2 Wordnet evaluation

There is a shortage of large scale test-sets for semantic similarity. Popular test-sets such as WordSim353 and the TOEFL synonyms test contain only 353 and 80 test items respectively, and therefore make it difficult to obtain statistically significant results. To automatically construct larger-scale test-sets for semantic similarity, we sampled large target word subsets from our corpus and used Wordnet as a gold standard for their semantically similar words, following related previous evaluations (Lin, 1998; Riedl and Biemann, 2013). We constructed two test-sets for our primary evaluation, one for verb similarity and another for noun similarity.

To perform the verb similarity evaluation, we randomly sampled 1,000 verbs from the target verb set, where the probability of each verb to be sampled is set to be proportional to its frequency in the learning corpus. Next, for each sampled verb a we constructed a Wordnet-based gold standard set of semantically similar words. In this set each verb a' is annotated as a 'synonym' of a if at least one of the senses of a' is a synonym of any of the senses of a. In addition, each verb a' is annotated as a 'semantic neighbor' of a if at least one of the senses of a' is a synonym, co-hyponym, or a direct hypernym/hyponym of any of the senses of a. We note that by definition all verbs annotated as

[3] http://code.google.com/p/word2vec

[4] http://code.google.com/p/berkeleylm/

synonyms of a are annotated as semantic neighbors as well. Next, per each verb a and an evaluated method, we generated a ranked list of all other verbs, which was induced according to the similarity scores of this method.

Finally, we evaluated the compared methods on two tasks, 'synonym detection' and 'semantic neighbor detection'. In the synonym detection task we evaluated the methods' ability to retrieve as much verbs annotated in our gold standard as 'synonyms', in the top-n entries of their ranked lists. Similarly, we evaluated all methods on the 'semantic neighbors' task. The synonym detection task is designed to evaluate the ability of the compared methods to identify a more restrictive interpretation of semantic similarity, while the semantic neighbor detection task does the same for a somewhat broader interpretation.

We repeated the above procedure for sampling 1,000 target nouns, constructing the noun Wordnet-based gold standards and evaluating on the two semantic similarity tasks.

4.3 VerbSim evaluation

The publicly available VerbSim test-set contains 130 verb pairs, each annotated with an average of 6 human judgements of semantic similarity (Yang and Powers, 2006). We extracted a 107 pairs subset of this dataset for which all verbs are in our learning corpus. We followed works such as (Yang and Powers, 2007; Agirre et al., 2009) and compared the Spearman correlations between the verb-pair similarity scores assigned by the compared methods and the manually annotated scores in this dataset.

5 Results

For each method and verb a in our 1,000 tested verbs, we used the Wordnet gold standard to compute the precision at top-1, top-5 and top-10 of the ranked list generated by this method for a. We then computed mean precision values averaged over all verbs for each of the compared methods, denoted as P@1, P@5 and P@10. The detailed report of P@10 results is omitted for brevity, as they behave very similarly to P@5. We varied the context window order used by all methods to test its effect on the results. We measured the same metrics for nouns.

The results of our Wordnet-based 1,000 verbs evaluation are presented in the upper part of Figure 1. The results show significant improvement of our method over all baselines, with a margin between 2 to 3 points on the synonyms detection task and 5 to 7 points on the semantic neighbors detection task. Our best performing configurations are PDS^{W-3} and PDS^{W-4}, outperforming all other baselines on both tasks and in all precision categories. This difference is statistically significant at $p < 0.001$ using a paired t-test in all cases except for the P@1 in the synonyms detection task. Within the baselines, the composite feature vector (CFV) performs somewhat better than the independent feature vector (IFV) baseline, and both methods perform best around window order of two, with gradual decline for larger windows. The word embedding baselines, CBOW and SKIP, perform comparably to the feature vector baselines and to one another, with best performance achieved around window order of four.

When gradually increasing the context window order within the range of up to 4 words, our PDS model shows improvement. This is in contrast to the feature vector baselines, whose performance declines for context window orders larger than 2. This suggests that our approach is able to take advantage of larger contexts in comparison to standard feature vector models. The decline in performance for the independent feature vector baseline (IFV) may be related to the fact that independent features farther away from the target word are generally more loosely related to it. This seems consistent with previous works, where narrow windows of the order of two words performed well (Bullinaria and Levy, 2007; Agirre et al., 2009; Bruni et al., 2012) and in particular so when evaluating semantic similarity rather than relatedness. On the other hand, the decline in performance for the composite feature vector baseline (CFV) may be attributed to the data sparseness phenomenon associated with larger windows. The performance of the word embedding baselines (CBOW and SKIP) starts declining very mildly only for window orders larger than 4. This might be attributed to the fact that these models assign lower weights to context words the farther away they are from the center of the window.

The results of our Wordnet-based 1,000 nouns evaluation are presented in the lower part of Figure 1. These results are partly consistent with the results achieved for verbs, but with a couple of notable differences. First, though our model still

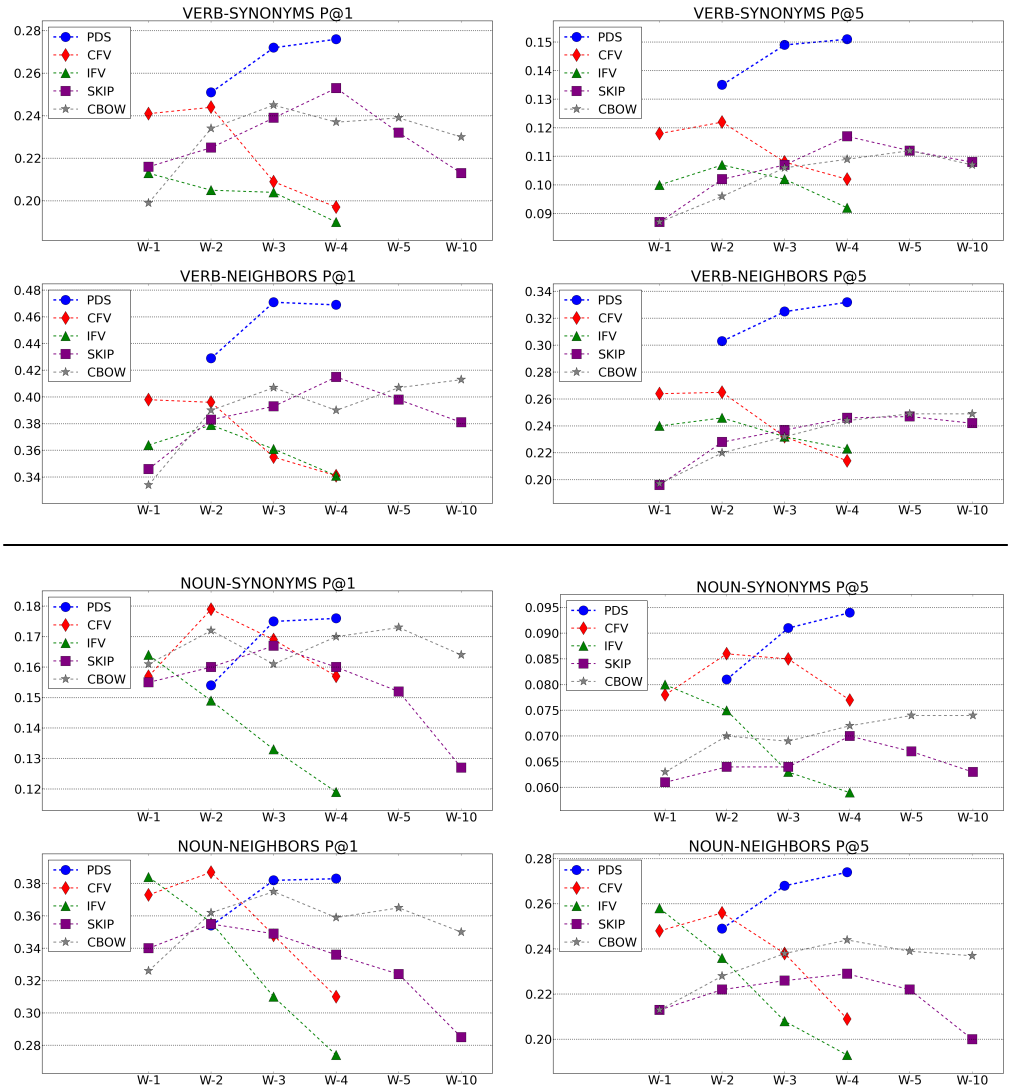

Figure 1: Mean precision scores as a function of window order, obtained against the Wordnet-based gold standard, on both the verb and noun test-sets with both the synonyms and semantic neighbor detection tasks. "P@n" stands for precision in the top-n words of the ranked lists. Note that the Y-axis scale varies between graphs.

outperforms or performs comparably to all other baselines, in this case the advantage of our model over the feature vector baselines is much more moderate and not statistically significant. Second, the word embedding baselines generally perform worst (with CBOW performing a little better than SKIP), and our model outperforms them in both P@5 and P@10 with a margin of around 2 points for the synonyms detection task and 3-4 points for the neighbor detection task, with statistical significance at $p < 0.001$.

Next, to reconfirm the particular applicability of our model to verb similarity as apparent from the Wordnet evaluation, we performed the VerbSim evaluation and present the results in Table 1. We compared the Spearman correlation obtained for the top-performing window order of each of the evaluated methods in the Wordnet verbs evaluation. We present two sets of results. The 'all scores' results follow the standard evaluation procedure, considering all similarity scores produced by each method. In the 'top-100 scores' results, for each method we converted to zero the scores that it assigned to word pairs, where neither of the words is in the top-100 most similar words of the other. Then we performed the evaluation with these revised scores. This procedure focuses on evaluating the quality of the methods' top-100 ranked word lists. The results show that our method outperforms all baselines by a nice mar-

Method	All scores	top-100 scores
PDS W-4	**0.616**	**0.625**
CFV W-2	0.477	0.497
IFV W-2	0.467	0.546
SKIP W-4	0.469	0.512
CBOW W-5	0.528	0.469

Table 1: Spearman correlation values obtained for the VerbSim evaluation. Each method was evaluated with the optimal window order found in the Wordnet verbs evaluation.

gin of more than 8 points with the score of 0.616 and 0.625 for the 'all scores' and 'top-100 scores' evaluations respectively. Though not statistically significant, due to the small test-set size, these results support the ones from the Wordnet evaluation, suggesting that our model performs better than the baselines on measuring verb similarity.

In summary, our results suggest that in lack of a robust context modeling scheme it is hard for distributional similarity models to effectively leverage larger word window contexts for measuring semantic similarity. It appears that this is somewhat less of a concern when it comes to noun similarity, as the simple feature vector models reach near-optimal performance with small word windows of order 2, but it is an important factor for verb similarity. In his recent book, Hanks (2013) claims that contrary to nouns, computational models that are to capture the meanings of verbs must consider their syntagmatic patterns in text. Our particularly good results on verb similarity suggest that our modeling approach is able to capture such information in larger context windows. We further conjecture that the reason the word embedding baselines did not do as well as our model on verb similarity might be due to their particular choice of joint-context formulation, which is not sensitive to word order. However, these conjectures should be further validated with additional evaluations in future work.

6 Future Directions

In this paper we investigated the potential for improving distributional similarity models by modeling jointly the occurrence of several features under the same context. We evaluated several previous works with different context modeling approaches and suggest that the type of the underlying context modeling may have significant effect on the performance of the semantic model. Furthermore, we introduced a generic probabilistic distributional similarity approach, which can leverage the power of established probabilistic language models to effectively model joint-contexts for the purpose of measuring semantic similarity. Our concrete model utilizing n-gram language models outperforms several competitive baselines on semantic similarity tasks, and appears to be particularly well-suited for verbs. In the remainder of this section we describe some potential future directions that can be pursued.

First, the performance of our generic scheme is largely inherited from the nature of its underlying language model. Therefore, we see much potential in exploring the use of other types of language models, such as class-based (Brown et al., 1992), syntax-based (Pauls and Klein, 2012) or hybrid (Tan et al., 2012). Furthermore, a similar approach to ours could be attempted in word embedding models. For instance, our syntagmatic joint-context modeling approach could be investigated by word embedding models to generate better embeddings for verbs.

Another direction relates to the well known tendency of many words, and particularly verbs, to assume different meanings (or senses) under different contexts. To address this phenomenon context sensitive similarity and inference models have been proposed (Dinu and Lapata, 2010; Melamud et al., 2013). Similarly to many semantic similarity models, our current model aggregates information from all observed contexts of a target word type regardless of its different senses. However, we believe that our approach is well suited to address context sensitive similarity with proper enhancements, as it considers joint-contexts that can more accurately disambiguate the meaning of target words. As an example, it is possible to consider the likelihood of word b to occur in a subset of the contexts observed for word a, which is biased towards a particular sense of a.

Finally, we note that our model is not a classic vector space model and therefore common vector composition approaches (Mitchell and Lapata, 2008) cannot be directly applied to it. Instead, other methods, such as similarity of compositions (Turney, 2012), should be investigated to extend our approach for measuring similarity between phrases.

Acknowledgments

This work was partially supported by the Israeli Ministry of Science and Technology grant 3-8705, the Israel Science Foundation grant 880/12, the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 287923 (EXCITEMENT) and the Scientific and Technical Research Council of Turkey (TÜBİTAK, Grant Number 112E277).

References

Eneko Agirre, Enrique Alfonseca, Keith Hall, Jana Kravalova, Marius Paşca, and Aitor Soroa. 2009. A study on similarity and relatedness using distributional and wordnet-based approaches. In *Proceedings of NAACL*. Association for Computational Linguistics.

Guy Aston. 1997. The BNC Handbook Exploring the British National Corpus with SARA Guy Aston and Lou Burnard.

Yoshua Bengio, Rjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1137–1155.

Peter F. Brown, Peter V. Desouza, Robert L. Mercer, Vincent J. Della Pietra, and Jenifer C. Lai. 1992. Class-based n-gram models of natural language. *Computational linguistics*, 18(4):467–479.

Elia Bruni, Gemma Boleda, Marco Baroni, and Nam-Khanh Tran. 2012. Distributional semantics in technicolor. In *Proceedings of ACL*.

Alexander Budanitsky and Graeme Hirst. 2001. Semantic distance in wordnet: An experimental, application-oriented evaluation of five measures. In *Workshop on WordNet and Other Lexical Resources*.

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.

Kenneth W. Church and Patrick Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16(1):22–29.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167. ACM.

Georgiana Dinu and Mirella Lapata. 2010. Measuring distributional similarity in context. In *Proceedings of EMNLP*.

Christiane Fellbaum. 2010. *WordNet*. Springer.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2001. Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on World Wide Web*. ACM.

James Gorman and James R. Curran. 2006. Scaling distributional similarity to large corpora. In *Proceedings of ACL*.

Patrick Hanks. 2013. *Lexical Analysis: Norms and Exploitations*. Mit Press.

Reinhard Kneser and Hermann Ney. 1995. Improved backing-off for m-gram language modeling. In *International Conference on Acoustics, Speech, and Signal Processing*. IEEE.

Beth Levin. 1993. *English verb classes and alternations: A preliminary investigation*. University of Chicago press.

Dekang Lin. 1998. Automatic retrieval and clustering of similar words. In *Proceedings of COLING-ACL*.

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

Oren Melamud, Jonathan Berant, Ido Dagan, Jacob Goldberger, and Idan Szpektor. 2013. A two level model for context sensitive inference rules. In *Proceedings of ACL*.

Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernockỳ, and Sanjeev Khudanpur. 2010. Recurrent neural network based language model. In *INTERSPEECH*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Jeff Mitchell and Mirella Lapata. 2008. Vector-based models of semantic composition. In *Proceedings of ACL*.

Adam Pauls and Dan Klein. 2011. Faster and Smaller N-Gram Language Models. In *Proceedings of ACL*.

Adam Pauls and Dan Klein. 2012. Large-scale syntactic language modeling with treelets. In *Proceedings of ACL*.

Martin Riedl and Chris Biemann. 2013. Scaling to large^3 data: An efficient and effective method to compute distributional thesauri. In *Proceedings of EMNLP*.

Tony Rose, Mark Stevenson, and Miles Whitehead. 2002. The Reuters Corpus Volume 1-from Yesterday's News to Tomorrow's Language Resources. In *LREC*.

Herbert Rubenstein and John B. Goodenough. 1965. Contextual correlates of synonymy. *Communications of the ACM*.

Maria Ruiz-Casado, Enrique Alfonseca, and Pablo Castells. 2005. Using context-window overlapping in synonym discovery and ontology extension. *Proceedings of RANLP*.

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

Peter D. Turney, Patrick Pantel, et al. 2010. From frequency to meaning: Vector space models of semantics. *Journal of artificial intelligence research*.

Peter D. Turney. 2012. Domain and function: A dual-space model of semantic relations and compositions. *Journal of Artificial Intelligence Researc*, 44(1):533–585, May.

Dongqiang Yang and David M. W. Powers. 2006. Verb similarity on the taxonomy of wordnet. In *the 3rd International WordNet Conference (GWC-06)*.

Dongqiang Yang and David M. W. Powers. 2007. An empirical investigation into grammatically constrained contexts in predicting distributional similarity. In *Australasian Language Technology Workshop 2007*, pages 117–124.

Mehmet Ali Yatbaz, Enis Sert, and Deniz Yuret. 2012. Learning syntactic categories using paradigmatic representations of word context. In *Proceedings of EMNLP*.