1, if two words come from the same semantic category (e.g., *man* and *woman*) they appear in similar sentence frames, thus ideally (when all possible sentence formulations exist in the generated sample of the language) they should be found as fully substitutable words. The paradigmatic task evaluates the quality of word vectors generated by a DSM by calculating the cosine similarity of word pairs belonging to same vs. different sematic categories.

$$Accuracy_{par} = Avg\ sim(w_i, w_j) - Avg\ sim(w_k, w_l)$$

where (w_i , w_j) indicates all word pairs coming from same semantic categories, and (w_k , w_l) indicates word pairs belong to different semantic categories. Based on this formulation, the paradigmatic accuracy of a model emphasizing second-order information would be higher than a model favoring first-order information to distribute words in the vector space. The reason is that, in the former model, the cosine similarity between vectors of interchangeable words like *man* and *woman* would converge to 1, or will be at least higher than similarity between other word vectors.[1] Both $Accuracy_{syn}$ and $Accuracy_{par}$ are bounded measures within the range of [-2, 2]; in practice though, they tend to come out within the range of [0, 1].

The above two tasks define the basics of our discriminative approach to investigate which models or parameter settings work best for each type of semantic similarity induction.

2.3 Distributional Methods

In our experiments, we use the implementations of `word2vec` Skip-Gram with Negative Sampling (SGNS) and PMI matrix factorization via Singular Value Decomposition (SVD) by Levy et al. (2015).

The Skip-gram model (SGNS) is one of the two `word2vec` architectures that predicts based on a target word one of its context words at a time. Error of prediction is calculated in the output via softmax and back-propagated to update two weight matrices: the context matrix (*CM*) between the output and the hidden layer $[]_{vd}$, and the word matrix (*WM*) between the input and the hidden layer $[]_{vd}$,where v is the vocabulary size and d is the size of the hidden layer, thus dimensionality of the final word vectors. In the majority of previous work, the word matrix was used as the final output of the model. When context words are sampled from the same vocabulary as that of target words, the final *CM* will have the same dimensionality as *WM*, thus it can also be used as a semantic representation of the words. Averaging both matrices for a final word representation, rather than just the *WM,* is an optional post-processing method indicated by $w+c$.

Singular Value Decomposition (SVD) is a classic representation learning technique for projecting data into a new, and usually, smaller feature space. Other similar techniques in machine learning include eigenvalue decomposition, the basis of Principle Component Analysis. The SVD model in our study is representative of the count-based distributional semantic models. It begins by calculating a $v*v$ matrix of point-wise mutual information between word-context pairs. The matrix is then factorized and reduced to a $v*d$ matrix, where each row will be a word vector in the new semantic space.

2.4 Implementation and Parameter Balancing

In all our experiments, we try to equate the two models by keeping the common parameters constant and iterating over different values of the method-specific parameters to obtain the best performance for each.

Fixed parameters: parameters that we keep constant throughout all experimental conditions are the context window size (set to 2, in order to cover all words within a sentence in the artificial grammar), subsampling & dynamic context (set to off; no frequency-based smoothing or prioritization is applied to co-occurrence counts), rare word removal (set to off, no minimum cut-off is applied to context words). Therefore, in all experimental conditions that result from manipulating other parameters exactly the same word-context population is extracted from a given corpus and fed as input data to the SGNS and SVD models. We also use one iteration (epoch) in SGNS to keep it equated with SVD, and examine the effect of re-occurrences by manipulating the corpus size instead.

Variable parameters: for comparative experiments on small vs. big data, we generate 5

[1] The paradigmatic task can also be defined based on higher-level taxonomic relations. For example, given the grammar in Table 1, we expect models to cluster Verbs and Nouns because each of these higher-level word types share some within-category contextual similarities and between-category differences (e.g., all nouns in the grammar have a verb in context, whereas verbs don't have verbs in their context). In section 3.5 where semantic spaces are visualized we will return to this important point, but for the rest of our experiments model performance is evaluated based on the two basic tasks defined above.