What's in Your Embedding, And How It Predicts Task Performance

Anna Rogers, Shashwath Hosur Ananthakrishna, Anna Rumshisky
Department of Computer Science,
University of Massachusetts Lowell, Lowell, MA, USA
{arogers,sha,arum}@cs.uml.edu

Abstract

Attempts to find a single technique for general-purpose intrinsic evaluation of word embeddings have so far not been successful. We present a new approach based on scaled-up qualitative analysis of word vector neighborhoods that quantifies interpretable characteristics of a given model (e.g. its preference for synonyms or shared morphological forms as nearest neighbors). We analyze 21 such factors and show how they correlate with performance on 14 extrinsic and intrinsic task datasets (and also explain the lack of correlation between some of them). Our approach enables multi-faceted evaluation, parameter search, and generally – a more principled, hypothesis-driven approach to development of distributional semantic representations.

1 Introduction

Dense lexical embeddings are the most common distributional semantic representations in both industrial and academic natural language processing (NLP) systems (Bengio et al., 2003; Collobert and Weston, 2008; Mikolov et al., 2013a; Pennington et al., 2014; Ruppert et al., 2015). They are used in task-specific neural network models, solving such tasks as named entity recognition (Guo et al., 2014), semantic role labeling (Chen et al., 2014), syntactic parsing (Chen and Manning, 2014), and more.

Each year dozens of new models are proposed, each of them with multiple hyper-parameters that may dramatically influence performance (Lapesa and Evert, 2014; Kiela and Clark, 2014; Levy et al., 2015; Lai et al., 2016; Melis et al., 2017). Equally important are the source corpus, its domain, and the type of context (Padó and Lapata, 2007; Levy and Goldberg, 2014a; Li et al., 2017; Lapesa and Evert, 2017). This amounts to an exponential explosion of options in the quest for the best model for a given task.

Ideally, there would be a single intrinsic metric for identifying "good" embeddings – and there are many proposals for such a metric (including word relatedness and analogies). However, none of them have been shown to predict performance on a wide range of tasks, and there is evidence to the contrary (Chiu et al., 2016).

We hypothesize that *different extrinsic tasks may rely on different aspects of word representations*. In that case, the only way to reliably predict what an embedding can do is to know what aspects of language it captures, and what aspects of language are relevant for different tasks.

To that end, we propose *Linguistic Diagnostics* (LD), a new approach to automated qualitative analysis of vector neighborhoods. To the best of our knowledge, this is the first large-scale attempt to identify and quantify the factors that make word embeddings successful with different tasks. We evaluate 60 models (the popular GloVe and Word2Vec with varying vector sizes and 4 types of context), identifying 21 factors that, to varying extent, correlate with the models' performance on 14 extrinsic and intrinsic task datasets. LD scores can be used not only for evaluation, but also for model development and optimization.

LD is implemented in LDT (Linguistic Diagnostics Toolkit), an open-source Python library[1] that offers a wide range of analysis options with corpus-based statistics, psychological association norms, and dictionaries. LDT provides broad lexical coverage thanks to a combination of the English WordNet, Wiktionary, and BabelNet, and is potentially extensible to many languages.

[1] http://ldtoolkit.space

2 Related Work

Perhaps the most popular kind of intrinsic evaluation of word embeddings are the **semantic relatedness** tests (Finkelstein et al., 2002; Bruni et al., 2014; Luong et al., 2013; Radinsky et al., 2011). They rely on the idea that the distance between word vectors should correlate with human judgements of how related the two words are (e.g., *cat* should be closer to *tiger* than to *hammer*). A more sophisticated version of this task is the **semantic similarity** (Agirre et al., 2009; Hill et al., 2015), which basically restricts relatedness to synonymy and co-hyponymy.

This evaluation paradigm has come under fire for methodological reasons (Faruqui et al., 2016; Batchkarov et al., 2016), in particular, due to the unreliability of the "middle" judgments: while *cat* should be closer to *tiger* than to *hammer*, it is not clear whether it should be closer to *lion* or to *tiger* (Gladkova and Drozd, 2016). Furthermore, only 1 out of 10 datasets was a good predictor of performance on sequence labeling tasks (Chiu et al., 2016). The proposal for evaluation via coherence of semantic space (Schnabel et al., 2015) inherits all the problems with relatedness (Gladkova and Drozd, 2016).

There are multiple proposals for "subconscious intrinsic evaluation" (Bakarov, 2018) based on **correlations with psycholinguistic data** such as N400 effect (Van Petten, 2014; Ettinger et al., 2016), fMRI scans (Devereux et al., 2010; Søgaard, 2016), eye-tracking (Klerke et al., 2015; Søgaard, 2016), and semantic priming data (Lund et al., 1995; Lund and Burgess, 1996; Jones et al., 2006; Lapesa and Evert, 2013; Ettinger and Linzen, 2016; Auguste et al., 2017). However, there are no large-scale studies that would show the utility of these methods in predicting downstream task performance. It is also possible that any psychological measure would share the subjectivity problem of relatedness judgments.

The idea behind the **word analogy task** (Mikolov et al., 2013b) is that the "best" word embedding is the one that encodes linguistic relations in the most regular way: simple vector offset should be sufficient to capture semantic shifts such as $France : Paris$ to $Japan : Tokyo$. However, this view of linguistic relations (and analogical reasoning) is oversimplified, and performance on word analogies has also been shown to depend on cosine similarity between source word vectors (Rogers et al., 2017; Linzen, 2016; Levy and Goldberg, 2014b). Furthermore, the original vector offset method is underestimating the amount of semantic information captured by the embedding (Drozd et al., 2016). Last but not the least, analogies also fail to yield results consistent with downstream task performance (Ghannay et al., 2016).

One more line of research could be called **linguistically motivated evaluation**. The idea is that a "good" embedding would be somehow similar to a representation that could be constructed from a gold-standard linguistic resource (Tsvetkov et al., 2015; Tsvetkov et al., 2016; Acs and Kornai, 2016).

Crucially, all these approaches make the same core assumption: that there is *one* feature of a representation that would make it the "best" (the highest correlation with human judgements, the most regular vector offsets, the closest approximation of a linguistic resource, etc.) However, language is a multi-faceted phenomenon, and different NLP tasks may rely on its different aspects – which would doom any one-metric-to-rule-them-all approach. This is the starting point for our solution.

3 LDT: the methodology

Consider two published modifications of the word2vec model, both trained on Wikipedia: the dependency-based embeddings (DEPS) (Levy and Goldberg, 2014a) and FastText (Bojanowski et al., 2017).

Table 1 lists the first 7 nearest neighbors of *color* (as measured by cosine similarity). Both models output the British spelling of the target word (*colour*). However, DEPS also includes derivatives and synonyms, while FastText favors misspellings and compounds, as could be expected of a subword-level model.

Rank	Deps		FastText	
1	colour	0.93	$color	0.75
2	colors	0.72	color…	0.69
3	coloration	0.69	colour	0.69
4	colouration	0.68	color#ff	0.69
5	colours	0.68	color#d	0.68
6	hue	0.66	@color	0.67
7	hues	0.65	barcolor	0.67

Table 1: Top 7 neighbors of *color* in dependency-based and FastText embeddings.

Which of these models is "better"? Without the context of some application, the question is meaningless. There is no theoretical reason why plural forms of nouns would make better/worse neighbors than their synonyms or misspellings.

This is a more meaningful question: *what are the properties of embedding X that could predict its performance on tasks Y and Z?* For example, question answering would likely benefit from synonymy more than morphology induction. Consider that relatedness tests were found to poorly correlate with performance on sequence labeling tasks, but SimLex (Hill et al., 2015) performed better (Chiu et al., 2016). This could be due to its focus on a particular type of semantic relations (synonymy, co-hyponymy), which turned out to be relevant for the labeling tasks.

Our solution is based on "linguistic diagnostic" tests, achieved by large-scale automatic annotation of linguistic, psychological and distributional relations between words vectors and their neighbors. The resulting data can then be used to find what features are useful for what extrinsic tasks. This work is inspired by the BLESS categorization dataset (Baroni and Lenci, 2011) and by evaluation via a set of representative extrinsic tasks (Nayak et al., 2016).

LD analysis starts with sampling the corpus vocabulary, as will be described in Section 4.2. For each word, top n neighbor vectors are extracted from each embedding. Each neighbor undergoes spelling normalization and is paired with the source word for analysis of possible morphological, semantic, distributional and psychological relations between them, as shown in Figure 1. The annotated data is analyzed to produce direct or statistically derived measures of the degree to which a given embedding is characterized by a given factor (e.g. how many synonyms or morphologically related words are neighbors of a given word vector). The exact set of LD factors considered in this study will be described in Section 5.1.

Since linguistic relations (especially semantic relations such as hypernymy and antonymy) cannot yet be classified accurately by purely distributional means[2], LDT relies on the largest freely available lexicographic resources: WordNet (Fellbaum, 1998) and Wiktionary[3], with the option of BabelNet (Navigli and Ponzetto, 2012). Currently, only English is supported, but (thanks to Wiktionary and BabelNet) LDT can be extended to other languages.

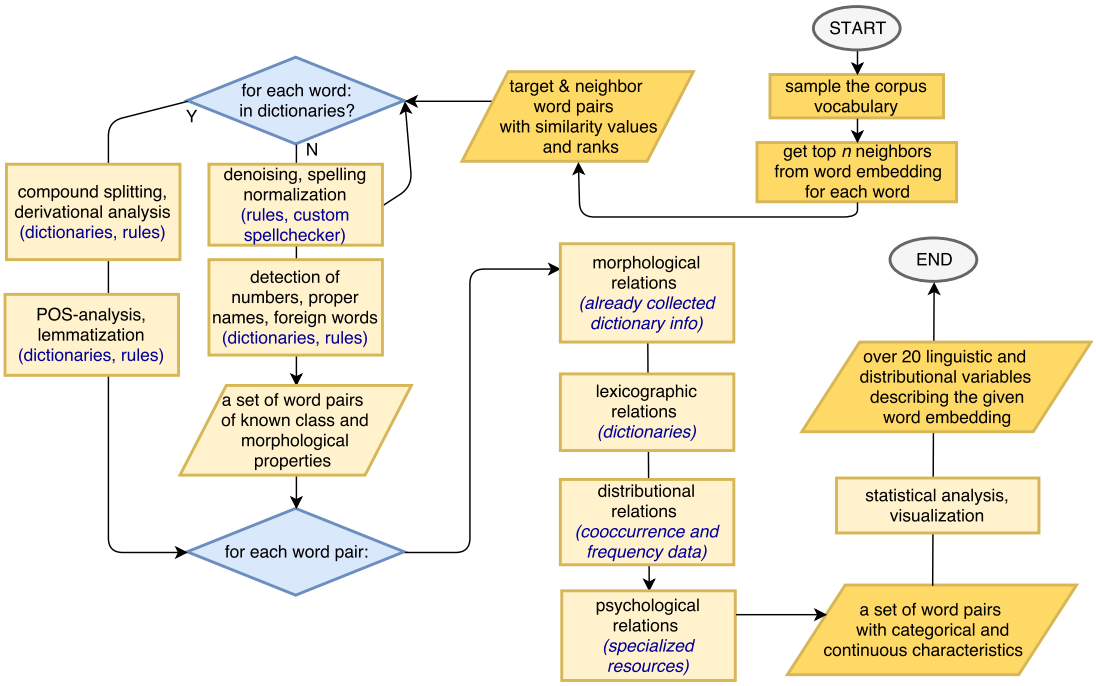


Figure 1: LDT analysis pipeline

[2]For example, the best performing system in the recent CogALex-V shared task (Santus et al., 2016) achieved only 45% accuracy in classifying only 5 semantic relations.

[3]https://en.wiktionary.org

4 Experiment set-up

4.1 Word embeddings

This work explores 3 popular word embedding algorithms: GloVe (Pennington et al., 2014), CBOW, and Skip-Gram (SG) (Mikolov et al., 2013a). The pre-trained vectors we used were published[4] by Li et al. (2017) who experimented with 4 different types of contexts on sequence labeling tasks. Additionally, they provided models with different vector sizes (25, 50, 100, 250, 500). In total, there were 60 models.

Table 2 shows that there are two types of context (linear and dependency-based), and two context representations: bound and unbound. The linear unbound context is the classic bag-of-words context (window size 2). The linear bound context is the same, except that words to the left and to the right are counted separately (Levy and Goldberg, 2014b; Ling et al., 2015). In the "bound" DEPS context (Levy and Goldberg, 2014a), the corpus is syntactically parsed, and only the words that are connected with the target word by some dependency relation are taken into account. Li et al. (2017) extended this idea into the "unbound" DEPS context, where the labels of syntactic roles are ignored.

Context Representation \ Context Type	Linear (Bag-of-Words)	DEPS
unbound	every, non-trivial, has, at	every, non-trivial, has
bound	every/-2, non-trivial/-1, has/+1, at/+2	every/+det, non-trivial+amod, has+nsubj

Table 2: Bound and unbound linear/dependency contexts for the word *program* in the sentence "*Every non-trivial program has at least one bug*". Adapted from (Li et al., 2017).

All embeddings were trained on English Wikipedia (August 2013 dump), with a minimum frequency of 100. After dependency parsing by Stanford CoreNLP (Manning et al., 2014), the corpus was lowercased. Negative sampling was set to 5 for SG and 2 for CBOW, no "dirty" sub-sampling. Distribution smoothing was set to 0.75. SG was trained for 2 epochs, CBOW - for 5, and GloVe - for 30.

4.2 Vocabulary Filtering and Sampling

Fair evaluation must take into account the amount of information that was available during training. It is possible to run LDT on any embeddings, but it yields the most information when the source corpus is available, and it is possible to estimate raw frequencies and cooccurence counts.

The source Wikipedia dump from which the embeddings were produced contains 14,404,885 token types. Only 273,229 of these occur over 100 times, but because of 4 context representations, the vocabulary of the different models is not the same (the DEPS vocabularies are particularly large, up to 5 times as many words). Since LDT methodology is based on the content of vector neighborhoods, to level the playing field for all models we filtered[5] the vocabulary down to 269,860 that were present in all models.

In this study, we focus on the general vocabulary and exclude proper nouns. We use LDT to draw a balanced sample of WordNet lemmas for four parts of speech (nouns, verbs, adjectives, adverbs) in 4 logarithmic frequency bins in the source corpus: 100, 1,000, 10,000, 100,000 (lower boundary inclusive). Following Baroni and Lenci (2011), we control for the polysemy of the words in the sample. For each part of speech at most 30 monosemous and polysemous words were drawn. Polysemy was defined as a word having over 2 meanings in WordNet[6]. The structure of the resulting sample is shown in Table 3.

Note that we also exclude the words belonging to several parts of speech (e.g. *a dog* (noun), *to dog* (verb)) to preserve the morphological class variable. This discards a lot of high-frequency vocabulary, which is why the higher frequency bins for verbs and adjectives were not populated fully. The total number of words in the sample is 908.

[4] `http://vecto.space/data/embeddings/en`

[5] Preliminary experiments showed that the filtering was beneficial to the performance on some tasks, and detrimental to others. The scope of this paper does not permit full investigation of the matter, but the effect was consistent across embeddings.

[6] This measure is not perfect, as numbers of senses in WordNet do not necessarily correspond to the number of senses in which a given word is used in Wikipedia, but it does provide a useful estimate.

4.3 Running LDT

We extract the 1,000 nearest neighbors for each word in the above sample. While most words will not have 1,000 meaningful relations, high-frequency words might have more than that. For example, a SG model with bound DEPS context has *rather* as the neighbor of *quite* at rank 920.

Frequency bin	Nouns	Verbs	Adj.	Adv.
100∼1,000	30 / 30	30 / 30	30 / 30	30 / 30
1,000∼10,000	30 / 30	30 / 30	30 / 30	30 / 30
10,000∼100,000	30 / 30	21 / 30	30 / 30	30 / 30
100,000 >	30 / 30	2 / 29	22 / 24	30 / 30

Table 3: The number of words sampled in each frequency bin per POS (monosemous / polysemous).

In total, 908,000 word pairs were processed for each of 60 embeddings. We limited the used resources to Wiktionary and WordNet, since BabelNet's maximum usage quota for research purposes (50,000 queries per day) would not be sufficient for this large-scale experiment.

The dictionaries covered 76,946 (28.51%) of all the neighbor words; another 124,511 (46.14%) were detected as proper nouns (as could be expected of a Wikipedia corpus). Thus, only 25.35% of the total vocabulary was not covered by LDT.

4.4 Extrinsic tasks

Each of 60 embeddings was evaluated on 8 extrinsic tasks[7]. The selection is similar to what Nayak et al. (2016) proposed as a representative suite of tasks for evaluation purposes. We also follow the recommendation of Nayak et al. (2016) in selecting simpler models for evaluation: more complex models often yield better accuracy, but they could smooth out the performance of different word embeddings and also raise the question of whether the gains are due to the model or the embeddings.

The morphological and syntactic information is targeted by two sequence labeling tasks: **POS-tagging** and **chunking**. We use the CoNLL 2003 shared task dataset (Tjong Kim Sang and De Meulder, 2003), following the method by Li et al. (2017). The model is a softmax classifier on the window-based concatenation of word embeddings of every training example (window size 3, 20 training epochs).

Semantic information at the word level is targeted by one more CoNLL 2003 shared task: **named entity recognition (NER)**, evaluated in the same way as POS-tagging and chunking. We also consider the task of **multi-way classification of semantic relations (Relation class.)** between pairs of nominals in the SemEval 2010 task 8 dataset (Hendrickx et al., 2010). The model we use is similar to the model by Zeng et al. (2014): a CNN equipped with word and distance embeddings.

Next, we have 3 tasks relying on how the word embeddings encode semantic information, and to what degree individual word vectors can be combined into an accurate sentence representation. The **sentence-level sentiment polarity classification (Sentiment (sent.))** task is tested with the MR dataset of short movie reviews (Pang and Lee, 2005). Binary classification is performed by a simplified version of the model proposed by Kim (2014).

We also add the **document-level polarity classification (Sentiment (text))** with the Stanford IMDB movie review dataset (Maas et al., 2011). Polarity is harder to estimate at the document than at the sentence level, because sentiment is more likely to be mixed. The task is performed with a single layer LSTM with 100 hidden units.

The **classification of subjectivity and objectivity (Subjectivity class.)** is tested on Rotten Tomato user review snippets vs official movie plot summaries (Pang and Lee, 2004). We follow the method by Li et al. (2017), employing a simple logistic regression model for the binary classification task. The input sentences are represented as a sum of their constituent word vectors.

Finally, the **natural language inference** task is represented with the SNLI dataset (Bowman et al., 2015). Similarly to the original proposal, we use two separate LSTMs to get a representation of the premise and the hypothesis using the last hidden state. The two hidden representations are merged and fed into a 50-unit dense layer, over which 3-class classification with softmax is performed.

[7] https://github.com/shashwath94/Extrinsic-Evaluation-tasks

4.5 Intrinsic tasks

Section 2 mentioned the reported lack of correlation between the performance of word embedding models on relatedness and sequence labeling tasks (Chiu et al., 2016). Ghannay et al. (2016) also report that the best-performing embedding on sequence labeling and mention detection tasks is not necessarily the embedding that performs the best on analogy and relatedness datasets. However, these studies have a limited selection of word embeddings (amounting to 9 and 5 data points, correspondingly). Crucially, they also focus on the same sequence labeling CoNLL tasks.

We explore the problem with our set of 60 embeddings, and a wider selection of extrinsic tasks. The intrinsic task datasets are WordSim353 (Finkelstein et al., 2002), together with its split into similarity and relatedness sections (Agirre et al., 2009), RareWords (Luong et al., 2013), MTurk (Radinsky et al., 2011), MEN (Bruni et al., 2014), and also the SimLex999 (Hill et al., 2015) similarity dataset. For the analogy task we use BATS dataset (Gladkova et al., 2016), which is currently the largest analogy dataset for English. We report separate scores for inflectional and derivational morphology, lexicographic and encyclopedic semantics, and the average of all categories.

The evaluation on similarity and relatedness datasets is performed as Spearman's correlation with the human judgement scores. The evaluation on analogies is performed with the state-of-the-art LRCos method (Drozd et al., 2016).

5 Results

5.1 Correlation analysis

In this study we experimented with 21 morphological, lexicographic, psychological, and distributional factors of word vector neighborhoods. For better readability, they are presented in Figure 2 together with their correlations with each other and the performance on 14 extrinsic and intrinsic task datasets (based on the data from 60 GloVe and Word2Vec embeddings described above).

Binary relations (e.g. synonymy is either detected or not) were quantified as a simple count of all cases of that relation in all target:neighbor pairs for each embedding. Directed lexicographic relations (hypernymy, hyponymy, meronymy) are counted when the target word is e.g. a hypernym of the neighbor. Continuous variables are broken down into bins, the size of which is chosen empirically: e.g. instead of frequency of the target word we count the number of low-frequency or high-frequency neighbors.

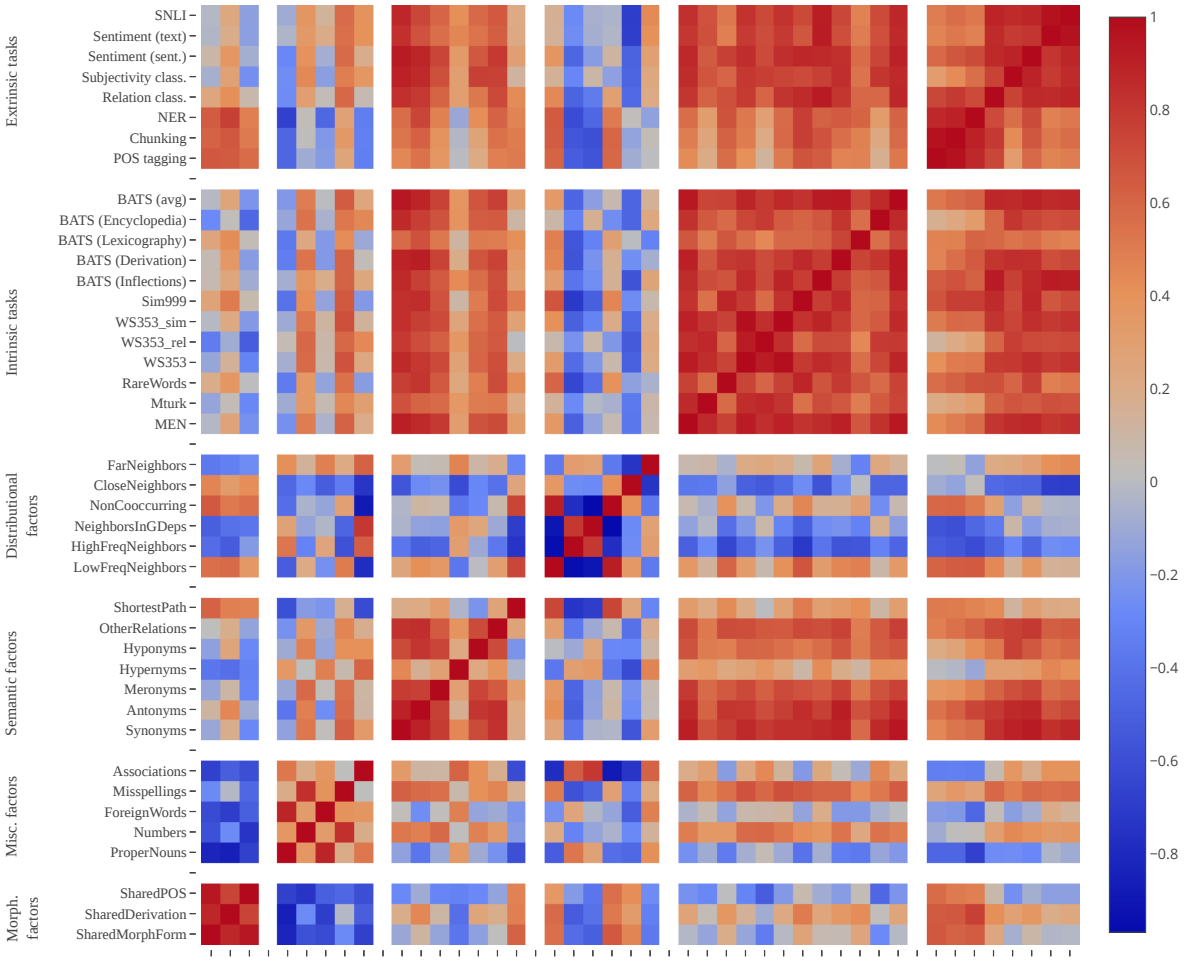
We experimented with the scores from 1,000, 5000, and 100 top neighbors of the sample words. The overall correlation patterns were similar, suggesting that it will be sufficient for future work to limit the selection to the top 100 neighbors. We thus report the results for the top 100 neighbors in this section.

The largest immediately observable pattern is the high correlation between *almost all* intrinsic tasks. The correlations are lower (but still over 0.5) for the lexicography and encyclopedic section of BATS; but the performance on these categories is generally rather low (Drozd et al., 2016) and could be unreliable. On the other hand, the high correlation between analogy and all relatedness and similarity datasets confirms the conclusion of (Rogers et al., 2017) that accuracy on analogy depends on the similarity between the source word pairs (even for LRCos method).

As for the extrinsic tasks, the immediate observation is the low correlation between all the intrinsic and 3 sequence labeling tasks. These are the same tasks that were reported by Chiu et al. (2016) and Ghannay et al. (2016) as the tasks for which higher performance does not correspond to higher performance on intrinsic tasks. However, all the non-sequence-labeling tasks in our sample *do* correlate with the intrinsic datasets.

This is a crucial finding; it shows that the traditional intrinsic tasks are after all useful for predicting performance on *some* downstream tasks. Their disadvantage is that they offer no explanation about why this is the case, and what could be expected of other extrinsic tasks not in our sample.

This is where LD methodology comes in. Figure 2 shows 4 groups of factors that we analyzed, together with their correlations with both extrinsic and intrinsic tasks. An immediate observation is that a large amount of neighbors that are in some lexicographic semantic relation with the target word is a good predictor of performance on both traditional intrinsic datasets and all the extrinsic tasks *except for the sequence labeling*. On the other hand, these particular tasks correlate highly with the three



Distributional factors
LowFreqNeighbors, HighFreqNeighbors: the frequency of the neighbor in the source Wikipedia corpus is under/above 10,000
NeighborsInGDeps: whether the two words co-occur in the Google dependency ngrams
NonCooccurring: the number of word pairs that do not co-occur in the source Wikipedia corpus (bag-of-words window size 2).
CloseNeighbors: the number of top 100 neighbors with cosine distance to the target word over 0.8.
FarNeighbors: the number of top n neighbors with cosine distance to the target word less than 0.7.

Semantic factors
Synonyms, Antonyms, Hypernyms, Hyponyms, Meronyms: the corresponding lexicographic relations established by the dictionaries.
OtherRelations: holonymy, troponymy, coordinate terms, and "otherwise related" in Wiktionary.
ShortestPath: the minimum path between synsets of two words in the WordNet ontology

Miscellaneous factors
ProperNouns: the neighbor is a proper noun.
Numbers: the neighbor is a numeral, or contains a number.
ForeignWords: the neighbor is not found in English, but found in German, French or Spanish spellechecker dictionaries.
Misspellings: the neighbor is not found in dictionaries and contains an unusual combination of letters and punctuation or numbers.
Associations: the two words constitute an associative pair (in either direction), according to EAT[8] (Wilson et al.,) and USF-FAN[9] (Nelson et al., 2004).

Morphological factors
SharedMorphForm: the two words share their morphological form (in this case, both are lemmas).
SharedDerivation: the two words share affix(es) or stem(s), or are both compounds (based on Wiktionary and custom LDT tools).
SharedPOS: the two words have the same part of speech (any overlap counts).

Figure 2: Pairwise Spearman's correlations of extrinsic and intrinsic tasks between themselves and LDT scores for top 100 neighbors. An interactive version of this chart, as well as numerical data and data for top 1000 neighbors can be found at `http://ldtoolkit.space/analysis`.

morphological factors we considered: the neighbors sharing morphological form, derivational pattern and/or part-of-speech of the target word. This finding confirms our original hypothesis: *different tasks rely on different information, making a single-number intrinsic evaluation unfeasible.*

At the same time, the border between morphology and semantics is not a stone wall. The derivational morphology factor does have weak positive correlations with all the intrinsic and extrinsic tasks, since shared derivation does indicate at least partially shared semantics. The performance on sequence labeling tasks also does correlate with the scores on lexicographic semantic relations. We attribute this to the fact that dictionaries usually store relations between words of the same part of speech, so these scores implicitly contain the *SharedPOS* factor.

The semantic factors that appear to be the least useful across all tasks are the *ShortestPath* and hypernymy. The latter is surprising in the light of such tasks as SNLI that seems to clearly rely on it.

The psychological associations turn out to be only weakly useful in the semantic extrinsic tasks (presumably to the same degree to which they correlate with relatedness tests, and relatedness tests correlate with extrinsic tasks). This is in line with Gladkova and Drozd (2016)'s suggestion that human relatedness scores depend on the psychological factors such as speed of association, rather than pure semantics.

It could be expected that, in the sample of general English vocabulary, the neighbors that are proper nouns or foreign words would be detrimental to any task. However, we observe a positive correlation with the amount of neighbors that contain numbers (presumably due to the meaningful hyponymy that they could indicate, such as model numbers, addresses etc.). A large number of misspelled neighbors is also apparently good for all tasks: since all the models in this study are word-level, this could indicate their ability to mitigate the lack of subword information.

Among the distributional factors, the clearest positive effect is observed for the models that have the highest number of low-frequency vocabulary (under 10,000 occurrences in the corpus) in the word vector neighborhoods. Since most word types fall in this range, this indicates that a "good" model should be able to populate vector neighborhoods with related words, even if they are not particularly frequent. The *NonCooccurring* factor is apparently useful for sequence labeling and some intrinsic tasks to find more *latent* relations between words that do not actually co-occur in the corpus, i.e. deduce relations on the basis of the "second-hand" rather than direct similarity between distributional patterns of words. Finally, the scores on the *FarNeighbors* factor suggest that high-level semantic tasks benefit from more neighbors that are less than 0.7 similar to the target word. This could be interpreted as follows: if a neighborhood is packed with words that are all quite similar, many of them will end up being within 0.00000001 from each other, making the margin of error very small for the models that use these representations in tasks.

One more important observation from this experiment is that all the extrinsic and intrinsic tasks have high correlations with more than one LD factor, which illustrates the point about tasks being complex *ensembles* of various linguistic features. However, it is only by breaking them down into smaller, controllable factors that we can explain and improve on them.

Note also that all the factors we considered correlate considerably with each other within their subclasses: the morphological features have mostly *negative* correlations with the lexicographic ones, while the sequence labeling tasks only weakly correlate with the high-level semantic tasks. This raises the question of what it would take for a representation to do both equally well.

5.2 Profiling embeddings with LD

As a brief demonstration of explanatory power of LD methodology, let us consider CBOW, SG, and GloVe models and their performance on the 8 above tasks in one condition: linear bag-of-words context, 500-dimensional vectors (the LD scores for top 1000 neighbors are reported). Table 4 lists some of the LD factors identified in Section 5.1 together with performance on our 8 extrinsic tasks.

We see that the 3 models are very close in most of factors; yet it is SG that is always slightly ahead in semantic, morphological and distributional LD factors and actual performance. CBOW is consistently slightly behind SG on these accounts, and slightly ahead on the scores for misspellings, foreign words, numbers and proper nouns (apparently at the cost of the meaningful relations).

The key difference between GloVe and Word2Vec appears to be the *LowFreqNeighbors* (amount of

low-frequency words as neighbors) and *NonCooccurring* words (words that end up as neighbors in spite of not co-occurring in the source corpus). This suggests that the success of SG is due to its ability to bring together related words even if they were rare, and/or did not co-occur in the corpus. This apparently outweighs even GloVe's significant advantage in sparser vector neighborhoods.

It is interesting that comparable or even superior scores on "morphological" factors did not give GloVe an advantage in POS-tagging and chunking tasks. Apparently specialized information is necessary but not sufficient for top performance, and it is successful ensembles of features that matter.

5.3 LD for parameter search

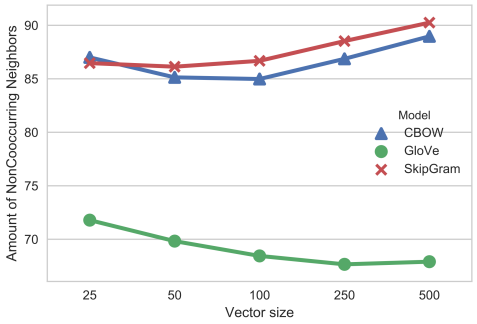
LD factors are equally useful for studying the effect of hyperparameters as well as underlying algorithms. As a brief demonstration, consider the behavior of the *NonCooccurring* factor discussed above when varying the size of SG, GloVe, and CBOW vectors (linear unbound context, top 1000 neighbors).

The larger representations are often assumed to be more informative, but Figure 3a shows that this is not the case for GloVe. The questions of why the compression effect is the smallest for the smallest vectors, and what other factors are at play here, merit a separate investigation. As in the case discussed in 5.2, Skip-Gram is consistently slightly ahead of CBOW, except for the lowest dimensionality.

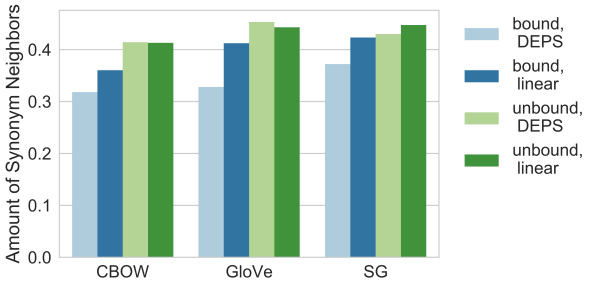
As a final example, let us take a quick look at the idea that the dependency-based contexts pack more synonyms than linear contexts. This does not seem to be the case in Fig. 3b: the positive effect is rather due to unbound vs bound representation than to dependency-based or linear context. Thus, if the goal is to maximize the number of synonyms, the effort of parsing is not justified. This result is consistent with the finding that dependency-based vector space models do not outperform the optimized window-based models on the TOEFL synonym task (Lapesa and Evert, 2017).

	CBOW	GloVe	SG
LD factors			
SharedMorphForm	51.819	52.061	**52.9**
SharedPOS	30.061	**35.507**	31.706
SharedDerivation	4.468	3.938	**5.084**
Synonyms	0.413	0.443	**0.447**
Antonyms	0.128	0.133	**0.144**
Hyponyms	0.035	0.035	**0.038**
OtherRelations	0.013	0.013	0.013
Misspellings	**13.546**	9.914	12.809
ForeignWords	**2.147**	1.976	1.793
ProperNouns	**30.442**	27.278	27.864
Numbers	**4.313**	3.147	3.64
LowFreqNeighbors	94.778	66.51	**96.109**
HighFreqNeighbors	3.421	**15.697**	2.513
NonCooccurring	88.97	67.904	**90.252**
CloseNeighbors	**3.102**	0.16	2.278
FarNeighbors	25.209	**49.934**	21.41
Downstream tasks			
POS-tagging	87.660	83.800	**87.860**
Chunking	77.530	66.100	**78.230**
NER	75.210	69.620	**75.720**
Relation class.	74.780	71.050	**74.800**
Subjectivity class.	89.800	89.160	**89.920**
Polarity (sent.)	75.900	74.600	**76.860**
Sentiment (text)	82.220	82.240	**82.730**
SNLI	69.290	69.510	**69.740**

Table 4: CBOW, GloVe and SG properties and performance



(a) Vector dimensionality effect on *NonCooccurring* factor.



(b) Amount of synonyms in models with different context types.

6 Discussion and Future Work

We have presented the LD methodology for quantitative/qualitative exploration of word embeddings. As proof of concept, our analysis of GloVe and Word2Vec showed that LD can effectively identify the linguistic and distributional factors that make word embeddings more or less successful on the downstream and traditional intrinsic tasks. We are hoping that this work will contribute to the NLP community efforts in the following directions:

- *comparison of word embedding algorithms* (e.g. different modifications of the Word2Vec);
- *hyperparameter effects* on encoding of different linguistic relations;
- a more informed, *hypothesis-driven design of new distributional representations*;
- *informed choice of word embeddings for various downstream tasks*;
- the degree to which different relations are useful for different tasks and to which they can be combined in a *generalized representation without sacrificing too much accuracy on specialist tasks*.
- interaction between *preference for different linguistic relations and performance on different tasks*.

LD methodology itself can be expanded by expanding LDT to other languages and by formulating the criteria for comparing representations of proper nouns. For example, the co-hyponymy relation between names of composers would be covered with the current implementation, but giving a higher score to a model that places *violin* closer to *Bach* than to *Beatles* would require evaluating frame-semantic, or at least topical relations, going beyond the traditional dictionaries.

A major caveat is the instability of word embeddings: different runs of the same model may yield word vector neighborhoods with significantly different lexical content. Some models are more stable than others (in particular, GloVe was found to be more stable than word2vec) Wendlandt et al. (2018), but most models published in the recent years do not explore their stability. This fact does not disqualify evaluations based on vector neighborhoods (not only LD, but also the traditional relatedness and analogy tasks), but it does highlight the absolute necessity of large-scale studies to reach any definitive conclusions about the relative (de)merits of different models. At the moment, most work on word embeddings still report experiments with less than ten models, each trained just once.

Important directions for future research also include going beyond simple word-level word embeddings. Some of the questions to investigate include the balance between semantic and morphological information in subword-level models (Bojanowski et al., 2017) and their ensembles with word-level models (Yang et al., 2017). It would also be interesting to expand LD to sense-aware embeddings (Melamud et al., 2016), particularly for contextualized representations (Peters et al., 2018).

7 Conclusion

We presented LD, a methodology for quantitative/qualitative intrinsic evaluation of word embeddings implemented in an open-source Python library. Moving away from unrealistic single-number evaluations, LD identifies precisely what kinds of information a given word embedding encodes in its vector neighborhoods. Unlike traditional intrinsic tasks, LD can also be used to *explain* the correlation between performance on different tasks, or the lack thereof.

The effectiveness of LD was shown in a large-scale experiment with 60 GloVe and Word2Vec models on 14 intrinsic and extrinsic task datasets. We have identified 21 morphological, semantic and distributional factors that are useful for predicting and interpreting the performance patterns of word embeddings. In addition to providing practical guidelines for choosing the best embeddings for a given task, LD opens new possibilities for more informed, hypothesis-driven development of distributional representations.

Acknowledgements

This work was supported in part by an NSF CAREER award to Anna Rumshisky (IIS-1652742).

References

Judit Acs and András Kornai. 2016. Evaluating embeddings on dictionary-based similarity. In *Proceedings of The 1st Workshop on Evaluating Vector Space Representations for NLP*, pages 78–82, Berlin, Germany. Association for Computational Linguistics.

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 Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, NAACL '09, pages 19–27, Stroudsburg, PA, USA. Association for Computational Linguistics.

Jeremy Auguste, Arnaud Rey, and Benoit Favre. 2017. Evaluation of word embeddings against cognitive processes: Primed reaction times in lexical decision and naming tasks. In *Proceedings of the 2nd Workshop on Evaluating Vector Space Representations for NLP*, pages 21–26.

Amir Bakarov. 2018. A survey of word embeddings evaluation methods. *arXiv:1801.09536 [cs]*, January.

Marco Baroni and Alessandro Lenci. 2011. How we BLESSed distributional semantic evaluation. In *Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics*, GEMS '11, pages 1–10. Association for Computational Linguistics.

Miroslav Batchkarov, Thomas Kober, Jeremy Reffin, Julie Weeds, and David Weir. 2016. A critique of word similarity as a method for evaluating distributional semantic models. In *Proceedings of The 1st Workshop on Evaluating Vector Space Representations for NLP*, Berlin, Germany. Association for Computational Linguistics.

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

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5(0):135–146, June.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics.

Elia Bruni, Nam-Khanh Tran, and Marco Baroni. 2014. Multimodal Distributional Semantics. *JAIR*, 49(1-47).

Danqi Chen and Christopher D. Manning. 2014. A fast and accurate dependency parser using neural networks. In *Proceedings of the Empiricial Methods in Natural Language Processing (EMNLP) 2014*, pages 740–750.

Yun-Nung Chen, William Yang Wang, and Alexander I. Rudnicky. 2014. Leveraging frame semantics and distributional semantics for unsupervised semantic slot induction in spoken dialogue systems. In *Spoken Language Technology Workshop (SLT), 2014 IEEE*, pages 584–589. IEEE.

Billy Chiu, Anna Korhonen, and Sampo Pyysalo. 2016. Intrinsic evaluation of word vectors fails to predict extrinsic performance. In *ACL 2016*, pages 1–6, Berlin, Germany. Association for Computational Linguistics.

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.

Barry Devereux, Colin Kelly, and Anna Korhonen. 2010. Using fMRI activation to conceptual stimuli to evaluate methods for extracting conceptual representations from corpora. In *Proceedings of the NAACL HLT 2010 First Workshop on Computational Neurolinguistics*, pages 70–78. Association for Computational Linguistics.

Aleksandr Drozd, Anna Gladkova, and Satoshi Matsuoka. 2016. Word embeddings, analogies, and machine learning: Beyond king - man + woman = queen. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3519–3530, Osaka, Japan, December 11-17.

Allyson Ettinger and Tal Linzen. 2016. Evaluating vector space models using human semantic priming results. In *Proceedings of The 1st Workshop on Evaluating Vector Space Representations for NLP*, pages 72–77, Berlin, Germany. Association for Computational Linguistics.

Allyson Ettinger, Naomi H. Feldman, Philip Resnik, and Colin Phillips. 2016. Modeling N400 amplitude using vector space models of word representation. In *Proceedings of the 38th Annual Conference of the Cognitive Science Society*, pages 1445–1450.

Manaal Faruqui, Yulia Tsvetkov, Pushpendre Rastogi, and Chris Dyer. 2016. Problems with evaluation of word embeddings using word similarity tasks. *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 30–35.

Christiane Fellbaum, editor. 1998. *Wordnet: An Electronic Lexical Database*. Language, speech, and communication series. MIT Press Cambridge.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2002. Placing search in context: The concept revisited. In *ACM Transactions on Information Systems*, volume 20(1), pages 116–131. ACM.

Sahar Ghannay, Benoit Favre, Yannick Esteve, and Nathalie Camelin. 2016. Word embedding evaluation and combination. pages 300–305, Portorož, Slovenia, May 23, 2016 – May 28, 2016. Association for Computational Linguistics.

Anna Gladkova and Aleksandr Drozd. 2016. Intrinsic evaluations of word embeddings: What can we do better? In *Proceedings of The 1st Workshop on Evaluating Vector Space Representations for NLP*, pages 36–42, Berlin, Germany. Association for Computational Linguistics. [doi:10.18653/v1/W16-2507].

Anna Gladkova, Aleksandr Drozd, and Satoshi Matsuoka. 2016. Analogy-based detection of morphological and semantic relations with word embeddings: What works and what doesn't. In *Proceedings of the NAACL-HLT SRW*, pages 47–54, San Diego, California, June 12-17, 2016. ACL.

Jiang Guo, Wanxiang Che, Haifeng Wang, and Ting Liu. 2014. Revisiting embedding features for simple semi-supervised learning. In *Proceedings of the Empiricial Methods in Natural Language Processing (EMNLP) 2014*, pages 110–120.

Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó. Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. 2010. Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, SemEval '10, pages 33–38, Stroudsburg, PA, USA. Association for Computational Linguistics.

Felix Hill, Roi Reichart, and Anna Korhonen. 2015. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4):665–695.

Michael N. Jones, Walter Kintsch, and Douglas J.K. Mewhort. 2006. High-dimensional semantic space accounts of priming. *Journal of Memory and Language*, 55(4):534–552, November.

Douwe Kiela and Stephen Clark. 2014. A systematic study of semantic vector space model parameters. In *Proceedings of the 2nd Workshop on Continuous Vector Space Models and Their Compositionality (CVSC) at EACL*, pages 21–30.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

Sigrid Klerke, Héctor Martínez Alonso, and Anders Søgaard. 2015. Looking hard: Eye tracking for detecting grammaticality of automatically compressed sentences. *Proceedings of the 20th Nordic Conference of Computational Linguistics (NODALIDA 2015)*, pages 97–105.

Siwei Lai, Kang Liu, Shizhu He, and Jun Zhao. 2016. How to generate a good word embedding. *IEEE Intelligent Systems*, 31(6):5–14.

Gabriella Lapesa and Stefan Evert. 2013. Evaluating neighbor rank and distance measures as predictors of semantic priming. In *Proceedings of the ACL Workshop on Cognitive Modeling and Computational Linguistics (CMCL 2013)*, pages 66–74.

Gabriella Lapesa and Stefan Evert. 2014. A large scale evaluation of distributional semantic models: parameters, interactions and model selection. *Transactions of the Association for Computational Linguistics*, 2:531–545.

Gabriella Lapesa and Stefan Evert. 2017. Large-scale evaluation of dependency-based DSMs: Are they worth the effort? In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 394–400. Association for Computational Linguistics.

Omer Levy and Yoav Goldberg. 2014a. Dependency-based word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, volume 2, pages 302–308.

Omer Levy and Yoav Goldberg. 2014b. Linguistic regularities in sparse and explicit word representations. *CoNLL-2014*, pages 171–180.

Omer Levy, Yoav Goldberg, and Ido Dagan. 2015. Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3:211–225.

Bofang Li, Tao Liu, Zhe Zhao, Buzhou Tang, Aleksandr Drozd, Anna Rogers, and Xiaoyong Du. 2017. Investigating different syntactic context types and context representations for learning word embeddings. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2411–2421.

Wang Ling, Chris Dyer, Alan W Black, and Isabel Trancoso. 2015. Two/too simple adaptations of word2vec for syntax problems. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1299–1304. Association for Computational Linguistics.

Tal Linzen. 2016. Issues in evaluating semantic spaces using word analogies. In *Proceedings of the First Workshop on Evaluating Vector Space Representations for NLP*, pages 13–18. Association for Computational Linguistics.

Kevin Lund and Curt Burgess. 1996. Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior Research Methods, Instruments, & Computers*, 28(2):203–208.

Kevin Lund, Curt Burgess, and Ruth Ann Atchley. 1995. Semantic and associative priming in high-dimensional semantic space. In *Proceedings of the 17th Annual Conference of the Cognitive Science Society*, volume 17, pages 660–665.

Minh-Thang Luong, Richard Socher, and Christopher D. Manning. 2013. Better word representations with recursive neural networks for morphology. *CoNLL-2013*, 104.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA, June. Association for Computational Linguistics.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60.

Oren Melamud, Jacob Goldberger, and Ido Dagan. 2016. Context2vec: Learning Generic Context Embedding with Bidirectional LSTM. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning (CoNLL)*, pages 51–61, Berlin, Germany, August 7-12, 2016. Association for Computational Linguistics.

Gábor Melis, Chris Dyer, and Phil Blunsom. 2017. On the State of the Art of Evaluation in Neural Language Models. *arXiv:1707.05589 [cs]*, July.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *Proceedings of International Conference on Learning Representations (ICLR)*.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013b. Linguistic regularities in continuous space word representations. In *HLT-NAACL*.

Roberto Navigli and Simone Paolo Ponzetto. 2012. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250, December.

Neha Nayak, Gabor Angeli, and Christopher D. Manning. 2016. Evaluating word embeddings using a representative suite of practical tasks. In *Proceedings of the 1st Workshop on Evaluating Vector Space Representations for NLP*, pages 19–23, Berlin, Germany, August 12, 2016. Association for Computational Linguistics.

Douglas L. Nelson, Cathy L. McEvoy, and Thomas A. Schreiber. 2004. The University of South Florida free association, rhyme, and word fragment norms. *Behavior Research Methods, Instruments, & Computers*, 36(3):402–407, August.

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

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the ACL*.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the ACL*.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, volume 12, pages 1532–1543.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Kira Radinsky, Eugene Agichtein, Evgeniy Gabrilovich, and Shaul Markovitch. 2011. A word at a time: Computing word relatedness using temporal semantic analysis. In *Proceedings of the 20th International Conference on World Wide Web*, WWW '11, pages 337–346, New York, NY, USA. ACM.

Anna Rogers, Aleksandr Drozd, and Bofang Li. 2017. The (too many) problems of analogical reasoning with word vectors. In *Proceedings of STARSEM 2017 (to Appear)*. Association for Computational Linguistics.

Eugen Ruppert, Manuel Kaufmann, Martin Riedl, and Chris Biemann. 2015. JOBIMVIZ: A web-based visualization for graph-based distributional semantic models. In *Proceedings of ACL-IJCNLP 2015 System Demonstrations*, pages 103–108, Beijing, China, July 26-31. ACL.

Enrico Santus, Anna Gladkova, Stefan Evert, and Alessandro Lenci. 2016. The CogALex-V shared task on the corpus-based identification of semantic relations. In *Proceedings of the Workshop on Cognitive Aspects of the Lexicon*, pages 69–79, Osaka, Japan, December 11-17. ACL.

Tobias Schnabel, Igor Labutov, David Mimno, and Thorsten Joachims. 2015. Evaluation methods for unsupervised word embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015), Lisbon, Portugal*, pages 298–307. Association for Computational Linguistics.

Anders Søgaard. 2016. Evaluating word embeddings with fMRI and eye-tracking. In *Proceedings of the 1st Workshop on Evaluating Vector Space Representations for NL*, pages 116–121, Berlin, Germany, August 12, 2016. Association for Computational Linguistics.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003-Volume 4*, pages 142–147. Association for Computational Linguistics.

Yulia Tsvetkov, Manaal Faruqui, Wang Ling, Guillaume Lample, and Chris Dyer. 2015. Evaluation of word vector representations by subspace alignment. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2049–2054, Lisbon, Portugal, 17-21 September 2015. Association for Computational Linguistics.

Yulia Tsvetkov, Manaal Faruqui, and Chris Dyer. 2016. Correlation-based intrinsic evaluation of word vector representations. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 111–115, Berlin, Germany. Association for Computational Linguistics.

Cyma Van Petten. 2014. Examining the N400 semantic context effect item-by-item: Relationship to corpus-based measures of word co-occurrence. *International Journal of Psychophysiology*, 94(3):407–419, December.

Laura Wendlandt, Jonathan K. Kummerfeld, and Rada Mihalcea. 2018. Factors influencing the surprising instability of word embeddings. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2092–2102, New Orleans, Louisiana, June. Association for Computational Linguistics.

Michael Wilson, Georg Kiss, and Christine Armstrong. EAT : The Edinburgh associative corpus [Electronic resource].

Zhilin Yang, Bhuwan Dhingra, Ye Yuan, Junjie Hu, William W Cohen, and Ruslan Salakhutdinov. 2017. Words or characters? Fine-grained gating for reading comprehension. In *Proceedings of ICLR*, pages 1–10.

Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, Jun Zhao, et al. 2014. Relation classification via convolutional deep neural network. In *COLING*, pages 2335–2344.