Intrinsic Evaluation of Word Vectors Fails to Predict Extrinsic Performance

Billy Chiu **Anna Korhonen** **Sampo Pyysalo**
Language Technology Lab
DTAL, University of Cambridge
{hwc25|alk23}@cam.ac.uk, sampo@pyysalo.net

Abstract

The quality of word representations is frequently assessed using correlation with human judgements of word similarity. Here, we question whether such intrinsic evaluation can predict the merits of the representations for downstream tasks. We study the correlation between results on ten word similarity benchmarks and tagger performance on three standard sequence labeling tasks using a variety of word vectors induced from an unannotated corpus of 3.8 billion words, and demonstrate that most intrinsic evaluations are poor predictors of downstream performance. We argue that this issue can be traced in part to a failure to distinguish specific similarity from relatedness in intrinsic evaluation datasets. We make our evaluation tools openly available to facilitate further study.

1 Introduction

The use of vector representations of words is now pervasive in natural language processing, and the importance of their evaluation is increasingly recognized (Collobert and Weston, 2008; Turian et al., 2010; Mikolov et al., 2013a; Faruqui and Dyer, 2014; Chen et al., 2013; Schnabel et al., 2015). Such evaluations can be broadly divided into intrinsic and extrinsic. The most common form of intrinsic evaluation uses word pairs annotated by humans to determine their degree of similarity (for varying definitions of *similarity*). These are then used to directly assess word representations based on how they rank the word pairs. In contrast, in extrinsic evaluation, word representations are used as input to a downstream task such as part-of-speech (POS) tagging or named entity recognition (NER). Here, good models are simply those that provide good performance in the downstream task according to task-specific metrics. Intrinsic evaluations are typically faster and easier to perform and they are often used to estimate the quality of representations before using them in downstream applications. The underlying assumption is that intrinsic evaluations can, to some degree, predict extrinsic performance.

In this study, we demonstrate that this assumption fails to hold for many standard datasets. We generate a set of word representations with varying context window sizes and compare their performance in intrinsic and extrinsic evaluations, showing that these evaluations yield mutually inconsistent results. Among all the benchmarks explored in our study, only SimLex-999 (Hill et al., 2015) is a good predictor of downstream performance. This may be related to the fact that it stands out among other benchmark datasets in distinguishing highly similar concepts (*male, man*) from highly related but dissimilar ones (*computer, keyboard*).

2 Materials and Methods

2.1 Word Vectors

We generate word representations using the *word2vec* implementation of the skip-gram model (Mikolov et al., 2013a), which can be efficiently applied to very large corpora and has been shown to produce highly competitive word representations in many recent evaluations, such as sentence completion, analogy tasks and sentiment analysis. (Mikolov et al., 2013a; Mikolov et al., 2013b; Fernández et al., 2014). We induce embeddings with varying values of the context window size parameter ranging between 1 and 30, holding other hyper-parameters to their defaults.[1]

[1]The default parameters are size=100, sample=0.001, negative=5, min-count=5, and alpha=0.025.

Proceedings of the 1st Workshop on Evaluating Vector Space Representations for NLP, pages 1–6,
Berlin, Germany, August 12, 2016.

Name	#Tokens	Reference
Wikipedia	2,032,091,934	Wikipedia (2016)
WMT14	731,451,760	Bojar et al. (2014)
1B-word-LM	768,648,884	Chelba et al. (2014)

Table 1: Unannotated corpora (sizes before tokenization)

Name	#Pairs	Reference
Wordsim-353	353	Finkelstein et al. (2001)
WS-Rel	252	Agirre et al. (2009)
WS-Sim	203	Agirre et al. (2009)
YP-130	130	Yang and Powers (2006)
MC-30	30	Miller and Charles (1991)
MEN	3000	Bruni et al. (2012)
MTurk-287	287	Radinsky et al. (2011)
MTurk-771	771	Halawi et al. (2012)
Rare Word	2034	Luong et al. (2013)
SimLex-999	999	Hill et al. (2015)

Table 2: Intrinsic evaluation datasets

2.2 Corpora and Pre-processing

To create word vectors, we gather a large corpus of unannotated English text, drawing on publicly available resources identified in word2vec distribution materials. Table 1 lists the text sources and their sizes. We extract raw text from the Wikipedia dump using the Wikipedia Extractor[2]; the other sources are textual. We pre-process all text with the Sentence Splitter and the Treebank Word Tokenizer provided by the NLTK python library (Bird, 2006). In total, there are 3.8 billion tokens (19 million distinct types) in the processed text.

2.3 Intrinsic evaluation

We perform intrinsic evaluations on the ten benchmark datasets presented in Table 2. We follow the standard experimental protocol for word similarity tasks: for each given word pair, we compute the cosine similarity of the word vectors in our representation, and then rank the word pairs by these values. We finally compare the ranking of the pairs created in this way with the gold standard human ranking using Spearman's ρ (rank correlation coefficient).

2.4 Downstream Methods

We base our extrinsic evaluation on the seminal work of Collobert et al. (2011) on the use of neural methods for NLP. In brief, we reimplemented the simple *window approach* feedforward neural network architecture proposed by Collobert et al., which takes as input words in a window of size five, followed by the word embedding, a single hidden layer of 300 units and a hard tanh activation leading to an output Softmax layer. Besides the index of each word in the embedding, the only other input is a categorical representation of the capitalization pattern of each word.[3]

We train each model on the training set for 10 epochs using word-level log-likelihood, mini-batches of size 50, and the Adam optimization method with the default parameters suggested by Kingma and Ba (2015). Critically, to emphasize the differences between the different representations, we do *not* fine-tune word vectors by back-propagation, diverging from Collobert et al. and leading to somewhat reduced performance. We use greedy decoding to predict labels for test data.

Name	#Tokens (Train/Test)
PTB	337,195 / 129,892
CoNLL 2000	211,727 / 47,377
CoNLL 2003	203,621 / 46,435

Table 3: Extrinsic evaluation datasets

2.5 Extrinsic evaluation

To evaluate the word representations in downstream tasks, we use them in three standard sequence labeling tasks selected by Collobert et al. (2011): POS tagging of Wall Street Journal sections of Penn Treebank (PTB) (Marcus et al., 1993), chunking of CoNLL'00 shared task data (Tjong Kim Sang and Buchholz, 2000), and NER of CoNLL'03 shared task data (Tjong Kim Sang and De Meulder, 2003). We use the standard train/test splits and evaluation criteria for each dataset, evaluating PTB POS tagging using token-level accuracy and CoNLL'00/03 chunking and NER using chunk/entity-level F-scores as implemented in the `conlleval` evaluation script. Table 3 shows basic statistics for each dataset.

3 Results

Tables 4 and 5 present the results of the intrinsic and extrinsic evaluations, respectively. While the different baselines and the small size of some of the datasets make the intrinsic results challenging to interpret, a clear pattern emerges when holding the result for word vectors of window size 1 as the zero point for each dataset and examining average differences: the intrinsic evaluations show higher

[2] `http://medialab.di.unipi.it/wiki/Wikipedia_Extractor`

[3] For brevity, we refer to Collobert et al. (2011) for further details on this method.

	Window size								
Dataset	1	2	4	5	8	16	20	25	30
WordSim-353	0.6211	0.6524	0.6658	0.6732	0.6839	0.6991	0.6994	**0.7002**	0.6981
MC-30	0.7019	0.7326	0.7903	0.7629	0.7889	0.8114	**0.8323**	0.8003	0.8141
MEN-TR-3K	0.6708	0.6860	0.7010	0.7040	0.7129	0.7222	0.7240	**0.7252**	0.7242
MTurk-287	0.6069	0.6447	0.6403	0.6536	0.6603	0.6580	**0.6625**	0.6513	0.6519
MTurk-771	0.5890	0.6012	**0.6060**	0.6055	0.6047	0.6007	0.5962	0.5931	0.5933
Rare Word	0.3784	0.3893	0.3976	**0.4009**	0.3919	0.3923	0.3938	0.3949	0.3953
YP130	0.3984	0.4089	0.4147	0.3938	0.4025	0.4382	0.4716	0.4754	**0.4819**
SimLex-999	**0.3439**	0.3300	0.3177	0.3144	0.3005	0.2909	0.2873	0.2811	0.2705

Table 4: Intrinsic evaluation results (ρ)

	Window size								
Dataset	1	2	4	5	8	16	20	25	30
CoNLL 2000	**0.9143**	0.9070	0.9058	0.9052	0.8982	0.8821	0.8761	0.8694	0.8604
CoNLL 2003	**0.8522**	0.8473	0.8474	0.8475	0.8474	0.8410	0.8432	0.8399	0.8374
PTB POS	**0.9691**	0.9680	0.9672	0.9674	0.9654	0.9614	0.9592	0.9560	0.9531

Table 5: Extrinsic evaluation results (F-score for CoNLL datasets, accuracy for PTB)

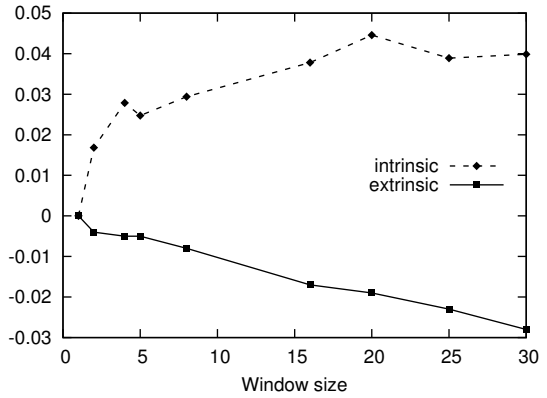

Figure 1: Average difference to performance for window size 1 for intrinsic and extrinsic metrics.

	CoNLL 2000	CoNLL 2003	PTB POS
WordSim-353	-0.90	-0.75	-0.88
MC-30	-0.87	-0.77	-0.90
MEN-TR-3K	-0.98	-0.83	-0.97
MTurk-287	-0.57	-0.29	-0.50
MTurk-771	0.28	0.37	0.27
Rare Word	-0.57	-0.29	-0.50
YP130	-0.82	-0.93	-0.50
SimLex-999	**1.00**	**0.85**	**0.98**

Table 6: Correlation between intrinsic and extrinsic measures (ρ)

overall results with increasing window size, while extrinsic performance drops (Figure 1).

Looking at the individual datasets, the preference for the smallest window size is consistent across all the three tagging tasks (Table 5) but only one out of the eight intrinsic evaluation datasets, Simlex-999, selects this window size, with the majority clearly favoring larger window sizes (Table 4). To further quantify this discrepancy, we ranked the word vectors from highest- to lowest-scoring according to each intrinsic and extrinsic measure and evaluated the correlation of each pair of these rankings using ρ. The results are striking (Table 6): six out of the eight intrinsic measures have *negative* correlations with all the three extrinsic measures, indicating that when selecting among the word vectors for these downstream tasks, it is *better to make a choice at random* than to base it on the ranking provided by any of the six intrinsic evaluations.

4 Discussion

Only two of the intrinsic evaluation datasets showed positive correlation with the extrinsic evaluations: MTurk-287 (ρ 0.27 to 0.37) and SimLex-999 (ρ 0.85 to 1.0). One of the differences between the other datasets and the high-scoring Simlex-999 is that it explicitly differentiates similarity from relatedness and association. For example, in the MEN dataset, the nearly synonymous pair (*stair, staircase*) and the highly associated but non-synonymous pair (*rain, storm*) are both given high ratings. However, as Hill et al. (2015) argue, an evaluation that measures semantic similarity should ideally distinguish these relations and credit a model for differentiating correctly that (*male, man*) are highly synonymous, while (*film, cinema*) are highly associated but dissimilar.

This distinction is known to be relevant to the effect of the window size parameter. A larger window not only reduces sparsity by introducing more contexts for each word, but is also known to affect the tradeoff between capturing *domain* similarity

	Window Size								
Dataset	1	2	4	5	8	16	20	25	30
WS-Rel	0.5430	0.5851	0.6021	0.6112	0.6309	0.6510	0.6551	**0.6568**	0.6514
WS-Sim	0.7465	0.7700	0.7772	0.7807	0.7809	**0.7885**	0.7851	0.7789	0.7776

Table 7: Intrinsic evaluation results for WS-Rel and WS-Sim (ρ)

vs. *functional* similarity: Turney (2012) notes that with larger context windows, representations tend to capture the *topic* or *domain* or a word, while smaller windows tend to emphasize the learning of word function. This is because the role/function of a word is categorized by its proximate syntactic context, while a large window captures words that are less informative for this categorization (Turney, 2012). For example, in the sentence *Australian scientist discovers star with telescope*, the context of the word *discovers* in a window of size 1 includes *scientist* and *star*, while a larger context window will include more words related by topic such as *telescope* (Levy and Goldberg, 2014). The association of large window sizes with greater topicality is discussed also by Hill et al. (2015) and Levy et al. (2015).

This phenomenon provides a possible explanation for the preference for representations created using larger windows exhibited by many of the intrinsic evaluation datasets: as these datasets assign high scores also to word pairs that are highly associated but dissimilar, representations that have similar vectors for all associated words (even if not similar) will score highly when evaluated on the datasets. If there is no need for the representation to make the distinction between similarity and relatedness, a large window has only benefits. On the other hand, the best performance in the extrinsic sequence labeling tasks comes from window size 1. This may be explained by the small window facilitating the learning of word function, which is more important for the POS tagging, chunking, and NER tasks than topic. Similarly, given the emphasis of SimLex-999 on capturing genuine similarity (synonyms), representations that assign similar vectors to words that are related but not similar will score poorly. Thus, we observe a decreasing trend with increasing window size for SimLex-999.

To further assess whether this distinction can explain the results for an intrinsic evaluation dataset for representations using small vs. large context windows, we studied the relatedness (WS-Rel) and similarity (WS-Sim) subsets (Agirre et al., 2009) of the popular WordSim-353 reference dataset (included in the primary evaluation). Table 7 shows the performance of representations with increasing context window size on these subsets. In general, both show higher ρ with an increasing context window size. However, the performance in the relatedness subset increases from 0.54 to 0.65 whereas that in similarity only increases from 0.74 to 0.77. Thus, although the similarity subset did not select a small window size, the lesser preference for a large window compared to the relatedness subset lends some support to the proposed explanation.

5 Conclusion

One of the primary goals of intrinsic evaluation is to provide insight into the quality of a representation before it is used in downstream applications. However, we found that the majority of word similarity datasets fail to predict which representations will be successful in sequence labelling tasks, with only one intrinsic measure, SimLex-999, showing high correlation with extrinsic measures. In concurrent work, we have also observed a similar effect for biomedical domain tasks and word vectors (Chiu et al., 2016). We further considered the differentiation between relatedness (association) and similarity (synonymy) as an explanatory factor, noting that the majority of intrinsic evaluation datasets do not systematically make this distinction.

Our results underline once more the importance of including also extrinsic evaluation when assessing NLP methods and resources. To encourage extrinsic evaluation of vector space representations, we make all of our newly introduced methods available to the community under open licenses from `https://github.com/cambridgeltl/RepEval-2016`.

Acknowledgments

This work has been supported by Medical Research Council grant MR/M013049/1

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-HLT'09*, pages 19–27.

Steven Bird. 2006. NLTK: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, pages 69–72.

Ondrej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, et al. 2014. Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 12–58.

Elia Bruni, Gemma Boleda, Marco Baroni, and Nam-Khanh Tran. 2012. Distributional semantics in technicolor. In *Proceedings of ACL'12*, pages 136–145.

Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. 2014. One billion word benchmark for measuring progress in statistical language modeling. In *Fifteenth Annual Conference of the International Speech Communication Association*.

Yanqing Chen, Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2013. The expressive power of word embeddings. *arXiv preprint arXiv:1301.3226*.

Billy Chiu, Gamal Crichton, Sampo Pyysalo, and Anna Korhonen. 2016. How to train good word embeddings for biomedical NLP. In *Proceedings of BioNLP'16*.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of ICML'08*, pages 160–167.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 12:2493–2537.

Manaal Faruqui and Chris Dyer. 2014. Community evaluation and exchange of word vectors at wordvectors.org. In *Proceedings of ACL'14: System Demonstrations*, June.

Javi Fernández, Yoan Gutiérrez, José M Gómez, and Patricio Martınez-Barco. 2014. Gplsi: Supervised sentiment analysis in twitter using skipgrams. In *Proceedings of SemEval'14*, pages 294–299.

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 WWW'01*, pages 406–414.

Guy Halawi, Gideon Dror, Evgeniy Gabrilovich, and Yehuda Koren. 2012. Large-scale learning of word relatedness with constraints. In *Proceedings of SIGKDD'12*, pages 1406–1414.

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

Diederik Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of ICLR'15*.

Omer Levy and Yoav Goldberg. 2014. Dependency-based word embeddings. In *Proceedings of ACL'14*, pages 302–308.

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.

Thang Luong, Richard Socher, and Christopher Manning. 2013. Better word representations with recursive neural networks for morphology. In *Proceedings of CoNLL*, pages 104–113.

Mitchell P Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. 1993. Building a large annotated corpus of english: The penn treebank. *Computational linguistics*, 19(2):313–330.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *Proceedings of Workshop at ICLR*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Proceedings of NIPS'13*, pages 3111–3119.

George A Miller and Walter G Charles. 1991. Contextual correlates of semantic similarity. *Language and cognitive processes*, 6(1):1–28.

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 WWW'11*, pages 337–346.

Tobias Schnabel, Igor Labutov, David Mimno, and Thorsten Joachims. 2015. Evaluation methods for unsupervised word embeddings. In *Proceedings of EMNLP'15*.

Erik F Tjong Kim Sang and Sabine Buchholz. 2000. Introduction to the conll-2000 shared task: Chunking. In *Proceedings of CoNLL'00*, pages 127–132.

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 CoNLL'03*, pages 142–147.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: a simple and general method for semi-supervised learning. In *Proceedings of ACL'10*, pages 384–394.

Peter D Turney. 2012. Domain and function: A dual-space model of semantic relations and compositions. *Journal of Artificial Intelligence Research*, pages 533–585.

Wikipedia. 2016. Wikipedia, the free encyclopedia. https://dumps.wikimedia.org/enwiki/latest/.

Dongqiang Yang and David MW Powers. 2006. Verb similarity on the taxonomy of wordnet. In *Proceedings of GWC'06*.