Learning Better Embeddings for Rare Words Using Distributional Representations

Irina Sergienya and **Hinrich Schütze**
Center for Information and Language Processing
University of Munich, Germany
sergienya@cis.lmu.de

Abstract

There are two main types of word representations: low-dimensional embeddings and high-dimensional distributional vectors, in which each dimension corresponds to a context word. In this paper, we initialize an embedding-learning model with distributional vectors. Evaluation on word similarity shows that this initialization significantly increases the quality of embeddings for rare words.

1 Introduction

Standard neural network (NN) architectures for inducing embeddings have an input layer that represents each word as a *one-hot vector* (e.g., Turian et al. (2010), Collobert et al. (2011), Mikolov et al. (2013)). There is no usable information available in this input-layer representation except for the identity of the word. We call this standard initialization method *one-hot initialization*.

Distributional representations (e.g., Schütze (1992), Lund and Burgess (1996), Sahlgren (2008), Turney and Pantel (2010), Baroni and Lenci (2010)) represent a word as a high-dimensional vector in which each dimension corresponds to a context word. They have been successfully used for a wide variety of tasks in natural language processing such as phrase similarity (Mitchell and Lapata, 2010) and sentiment analysis (Turney and Littman, 2003).

In this paper, we investigate *distributional initialization*: the use of distributional vectors as representations of words at the input layer of NN architectures for embedding learning to improve the embeddings of rare words. It is difficult for one-hot initialization to learn good embeddings from only a few examples. In contrast, distributional initialization provides an additional source of information – the global distribution of the word in the corpus – that improves embeddings learned for rare words. We will demonstrate this type of improvement in the experiments reported below.

In summary, we introduce the idea of distributional initialization for embedding learning, an alternative to one-hot initialization that combines distributed representations (or embeddings) with distributional representations (or high-dimensional vectors). We show that distributional initialization significantly improves the quality of embeddings learned for rare words.

We will first describe our methods in Section 2 and the experimental setup in Section 3. Section 4 presents and discusses experimental results. We summarize related work in Section 5 and finish with conclusion in Section 6 and discussion of future work in Section 7.

2 Method

Weighting. We use two different weighting schemes for distributional vectors. Let $v_1, \ldots, v_n$ be the vocabulary of context words. In BINARY weighting, entry $1 \leq i \leq n$ in the distributional vector of target word w is set to 1 iff v_i and w cooccur at a distance of at most ten words in the corpus and to 0 otherwise.

In PPMI weighting, entry $1 \leq i \leq n$ in the distributional vector of target word w is set to the PPMI (positive pointwise mutual information, introduced by Niwa and Nitta (1994)) of w and v_i. We divide PPMI values by their maximum to ensure they are in $[0, 1]$ because we will combine one-hot vectors (whose values are 0/1) with PPMI weights and it is important that they are on the same scale.

We use two different **distributional initializations**, shown in Figure 1: *separate* (left) and *mixed* (right). Combinations of these two initializations with both BINARY and PPMI weighting will be investigated in the experiments.

Recall that n is the dimensionality of the distri-

Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 280–285,
Lisbon, Portugal, 17-21 September 2015.

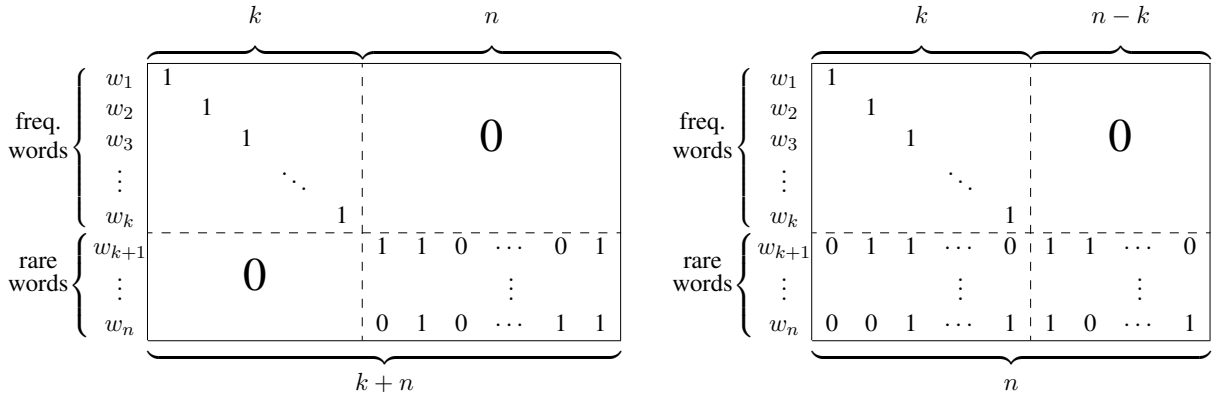

Figure 1: One-hot vectors of frequent words and distributional vectors of rare words are separate in *separate initialization* (left) and overlap in *mixed initialization* (right). This example is for BINARY weighting.

butional vectors. Let k be the number of words with frequency $> \theta$, where the frequency threshold θ is a parameter.

In *separate* initialization, the input representation for a word is the concatenation of a k-dimensional vector and an n-dimensional vector. For a word with frequency $> \theta$, the k-dimensional vector is a one-hot vector and the n-dimensional vector is zero. For a word with frequency $\leq \theta$, the k-dimensional vector is zero and the n-dimensional vector is its distributional vector.

In *mixed* initialization, the input representation for a word is an n-dimensional vector: a one-hot vector for a word with frequency $> \theta$ and a distributional vector for a word with frequency $\leq \theta$.

In summary, separate initialization uses separate representation spaces for frequent words (one-hot space) and rare words (distributional space). Mixed initialization uses the same representation space for all words; and rare words share weights with the frequent words that they cooccur with.

3 Experimental setup

We use ukWaC+WaCkypedia (Baroni et al., 2009), a corpus of 2.4 billion tokens and 6 million word types. Based on (Turian et al., 2010), we preprocess the corpus by removing sentences that are less than 90% lowercase; lowercasing; replacing URLs, email addresses and digits by special tokens; tokenization (Schmid, 2000); replacing words with frequency 1 with <unk>; and adding end-of-sentence tokens. After preprocessing, the size n of the context word vocabulary is 2.7 million.

We evaluate on six word similarity judgment data sets (number of pairs in parentheses): **RG** (Rubenstein and Goodenough (1965), 65) **MC** (Miller and Charles (1991), 30), **MEN**[1] (Bruni et al. (2012), 3000), **W**ord**S**im353[2] (Finkelstein et al. (2001), 353), Stanford **R**are **W**ord[3] (Luong et al. (2013), 2034) and **S**im**L**ex-999[4] (Hill et al. (2014), 999). We exclude from the evaluation the 16 pairs in RW that contain a word that does not occur in our corpus.

Our goal in this paper is to investigate the effect of using distributional initialization vs. one-hot initialization on the quality of embeddings of rare words.

However, except for RW, the six data sets contain only a single word with frequency $\leq$100, all other words are more frequent.

To address this issue, we artificially make all words in the six data sets rare. We do this by keeping only θ randomly chosen occurrences in the corpus (for words with frequency $>\theta$) and replacing all other occurrences with a different token (e.g., "fire" is replaced with "*fire*"). This procedure – *corpus downsampling* – ensures that all words in the six data sets are rare in the corpus and that our setup directly evaluates the impact of distributional initialization on rare words.

Note that we use θ for two different purposes: (i) θ is the frequency threshold that determines which words are classified as rare and which as frequent in Figure 1 – changing θ corresponds to moving the horizontal dashed line in separate and mixed initialization up and down; (ii) θ is the parameter that determines how many occurrences of a word are left in the corpus when we remove oc-

[1] clic.cimec.unitn.it/~elia.bruni/MEN
[2] alfonseca.org/eng/research/wordsim353.html
[3] www-nlp.stanford.edu/~lmthang/morphoNLM/
[4] cl.cam.ac.uk/~fh295/simlex.html

			A	B	C	D	E	F	G	H	I	J	K	L
			RG		MC		MEN		WS		RW		SL	
		θ	mixed	sep	mixed	sep	mixed	sep	mixed	sep	mixed	sep	mixed	sep
1	BINARY	10	***56.54**	**47.06**	**35.96**	**32.10**	***43.76**	***45.56**	34.21	***40.93**	***24.81**	**20.85**	***18.30**	***13.76**
2		20	***59.08**	**45.31**	***46.66**	**35.22**	**52.05**	***52.38**	41.44	**47.53**	***29.48**	**26.93**	***20.85**	***16.86**
3		50	***63.20**	51.07	***52.35**	37.45	58.21	53.80	43.14	44.88	31.32	29.16	***24.19**	***22.45**
4		100	**68.33**	52.50	**61.70**	35.94	61.69	55.23	48.25	44.89	33.29	30.22	***26.74**	**24.66**
5	PPMI	10	***56.87**	***51.94**	***37.31**	***46.52**	***48.05**	***50.49**	**38.41**	***47.54**	***25.53**	**23.12**	***19.70**	***15.59**
6		20	***59.08**	***50.32**	***47.51**	***45.17**	***54.88**	***56.42**	43.31	***53.19**	***29.78**	***28.51**	***21.84**	***19.23**
7		50	***64.90**	***64.36**	***55.27**	***56.75**	**60.51**	**61.04**	45.76	**55.55**	32.05	30.25	***25.11**	***21.60**
8		100	**71.08**	58.37	**68.14**	52.33	63.05	60.74	48.66	55.49	33.25	30.49	***27.13**	22.60
9	one-hot	10	38.93		16.67		40.70		35.17		20.69		8.97	
10		20	42.17		25.21		50.21		43.74		26.58		13.62	
11		50	56.01		42.35		60.22		54.10		32.16		20.01	
12		100	67.47		61.33		65.14		59.87		35.19		24.06	

Table 1: Spearman correlation coefficients $\times 100$ between human and embedding-based similarity judgments, averaged over 5 runs. Distributional initialization correlations that are higher (resp. significantly higher) than corresponding one-hot correlations are set in **bold** (resp. marked *).

currences to ensure that words from the evaluation data sets are rare in the corpus.

We covary these two parameters in the experiments below; e.g., we apply distributional initialization with $\theta = 20$ to a corpus constructed to have $\theta = 20$ occurrences of words from similarity data sets. We do this to ensure that all evaluation words are rare words for the purpose of distributional initialization and so we can exploit all pairs in the evaluation data sets for evaluating the efficacy of our method for rare words.

We modified word2vec[5] (Mikolov et al., 2013) to accommodate distributional initialization; to support distributional vectors at the input layer, we changed the implementation of activation functions and backpropagation. We use the skipgram model, hierarchical softmax, set the size of the context window to 10 (10 words to the left and 10 to the right), min-count to 1 (train on all tokens), embedding size to 100, sampling rate to 10^{-3} and train models for one epoch.

For four values of the frequency threshold, $\theta \in \{10, 20, 50, 100\}$,[6] we train word2vec models with one-hot initialization and with the four combinations of weighting (BINARY, PPMI) and distributional initialization (mixed, separate), a total of $4 \times (1 + 2 \times 2) = 20$ models. For each training run, we perform corpus downsampling and initialize the parameters of the models randomly. To get a reliable assessment of performance, we train 5 instances of each model and report averages of the 5 runs. One model takes $\sim$3 hours to train on 23 CPU cores, 2.30GHz.

4 Experimental results and discussion

Table 1 shows experimental results, averaged over 5 runs. The evaluation measure is Spearman correlation $\times 100$ between human and machine-generated pair similarity judgments.

Frequency threshold θ. The main result is that for $\theta \in \{10, 20\}$ *distributional initialization is better than one-hot initialization* (see bold numbers): compare lines 1&5 with line 9; and lines 2&6 with line 10. This is true for both mixed and separate initialization, with the exception of WS, for which mixed (column G) is better in only 1 (line 5) of 4 cases.

Looking only at results for $\theta \in \{10, 20\}$, 18 of 24 improvements are significant[7] for mixed initialization and 16 of 24 improvements are significant for separate initialization (lines 1&5 vs 9 and lines

[5] code.google.com/p/word2vec

[6] A reviewer asks whether the value of θ should depend on the size of the training corpus. Our intuition is that it is independent of corpus size. If a certain amount of information – corresponding to a certain number of contexts – is required to learn a meaningful representation of a word, then it should not matter whether that given number of contexts occurs in a small corpus or in a large corpus. However, if the contexts themselves contain many rare words (which is more likely in a small corpus), then corpus size could be an important variable to take into account.

[7] Two-sample t-test, two-tailed, assuming equal variance, $p < .05$

2&6 vs 10).

For $\theta \in \{50, 100\}$, mixed initialization does well for RG, MC and SL, but the gap between mixed and one-hot initializations is generally smaller for these larger values of θ; e.g., the difference is larger than 9 for $\theta = 10$ (A1&A5 vs A/B9, C1&C5 vs C/D9, K1&K5 vs K/L9) and less than 9 for $\theta = 100$ (A4&A8 vs A/B12, C4&C8 vs C/D12, K4&K8 vs K/L12) for these three data sets.

Recall that each value of θ effectively results in a different training corpus – a training corpus in which the number of occurrences of the words in the evaluation data sets has been reduced to $\leq \theta$ (cf. Section 3).

Our results indicate that distributional initialization is beneficial for very rare words – those that occur no more than 20 times in the corpus. Our results for medium rare words – those that occur between 50 and 100 times – are less clear: either there are no improvements or improvements are small.

Thus, our recommendation is to use $\theta = 20$.

Scalability. The time complexity of the basic version of word2vec is $O(ECWD \log V)$ (Mikolov et al., 2013) where E is the number of epochs, C is the corpus size, W is the context window size, D is the number of dimensions of the embedding space, and V is the vocabulary size. Distributional initialization adds a term I, the average number of entries in the distributional vectors, so that time complexity increases to $O(IECWD \log V)$. For rare words, I is small, so that there is no big difference in efficiency between one-hot initialization and distributional initialization of word2vec. However, for frequent words I would be large, so that distributional initialization may not be scalable in that case. So even if our experiments had shown that distributional initialization helps for both rare and frequent words, scalability would be an argument for only using it for rare words.

Binary vs. PPMI. PPMI weighting is almost always better than BINARY, with three exceptions (I8, L7, L8) where the difference between the two is small and not significant. The probable explanation is that the PPMI weights in $[0, 1]$ convey detailed, graded information about the strength of association between two words, taking into account their base frequencies. In contrast, the BINARY weights in $\{0, 1\}$ only indicate if there was any instance of cooccurrence at all – without considering frequency of cooccurrence and without normalizing for base frequencies.

Mixed vs. Separate. Mixed initialization is less variable and more predictable than separate initialization: performance for mixed initialization always goes up as θ increases, e.g., 56.54 $\rightarrow$ 59.08 $\rightarrow$ 63.20 $\rightarrow$ 68.33 (column A, lines 1–4). In contrast, separate initialization performance often decreases, e.g., from 47.06 to 45.31 (column B, lines 1–2) when θ is increased. Since more information (more occurrences of the words that similarity judgments are computed for) should generally not have a negative effect on performance, the only explanation is that separate is more variable than mixed and that this variability sometimes results in decreased performance. Figure 1 explains this difference between the two initializations: in mixed initialization (right panel), rare words are tied to frequent words, so their representations are smoothed by representations learned for frequent words. In separate initialization (left panel), no such links to frequent words exist, resulting in higher variability.

Because of its lower variability, our experiments suggest that mixed initialiation is a better choice than separate initialization.

One-hot vs. Distributional initialization. Our experiments show that distributional representation is helpful for rare words. It is difficult for one-hot initialization to learn good embeddings for such words, based on only a small number of contexts in the corpus. In such cases, distributional initialization makes the learning task easier since in addition to the *contexts of the rare word*, the learner now also has access to the *global distribution of the rare word* and can take advantage of weight sharing with other words that have similar distributional representations to smooth embeddings systematically.

Thus, distributional initialization is a form of smoothing: the embedding of a rare word is tied to the embeddings of other words via the links shown in Figure 1: the 1s in the lower "rare words" part of the illustrations for separate and mixed initialization. As is true for smoothing in general, parameter estimates for frequent events benefit less from smoothing or can even deteriorate. In contrast, smoothing is essential for rare events. Where the boundary lies between rare and frequent events depends on the specifics of the problem and the

smoothing method used and is usually an empirical question. Our results indicate that that boundary lies somewhere between 20 and 50 in our setting.[8]

Variance of results. Table 1 shows averages of five runs. The variance of results was quite high for low-performing models. For higher performing models – those with values ≥ 40 – the ratio of standard deviation divided by mean ranged from .005 to .29. The median was .044. While the variance from run to run is quite high for low-performing models and for a few high-performing models, the significance test takes this into account, so that the relatively high variability does not undermine our results.

In summary, we have shown that distributional initialization improves the quality of word embeddings for rare words. Our recommendation is to use mixed initialization with PPMI weighting and the value $\theta = 20$ of the frequency threshold.

5 Related work

An alternative to using distributional information for initialization is to use syntactic and semantic information for initialization. Approaches along these lines include Botha and Blunsom (2014) who represent a word as a sum of embedding vectors of its morphemes. Cui et al. (2014) use a weighted average of vectors of morphologically similar words. Bian et al. (2014) extend a word's vector with vectors of entity categories and POS tags. This line of work also is partially motivated by improving the embeddings of rare words. Distributional information on the one hand and syntactic/semantic information on the other hand are likely to be complementary, so that a combination of our approach with this prior work is promising.

Le et al. (2010) propose three schemes to address word embedding initialization. *Reinitialization* and *iterative reinitialization* use vectors from prediction space to initialize the context space during training. This approach is both more complex and less efficient than ours. *One-vector initialization* initializes all word embeddings with the same random vector to keep rare words close to each other. This approach is also less efficient than ours since the initial embedding is much denser than in our approach.

6 Conclusion

We have introduced distributional initialization of neural network architectures for learning better embeddings for rare words. Experimental results on a word similarity judgment task demonstrate that embeddings of rare words learned with distributional initialization perform better than embeddings learned with traditional one-hot initialization.

7 Future work

Our work is the first exploration of the utility of distributional representations as initialization for embedding learning algorithms like word2vec. There are a number of research questions we would like to investigate in the future.

First, we showed that distributional representation is beneficial for words with very low frequency. It was not beneficial in our experiments for more frequent words. A more extensive analysis of the factors that are responsible for the positive effect of distributional representation is in order.

Second, to simplify our experimental setup and make the number of runs mangeable, we used the parameter θ both for corpus processing (only θ occurrences of a particular word were left in the corpus) and as the separator between rare words that are distributionally initialized and frequent words that are not. It remains to be investigated whether there are interactions between these two properties of our model, e.g., a high rare-frequent separator may work well for words whose corpus frequency is much smaller than the separator.

Third, while we have shown that distributional initialization improves the quality of representations of rare words, we did not investigate whether distributional initialization for rare words has any adverse effect on the quality of representations of frequent words for which one-hot initialization is applied. Since rare and frequent words are linked in the mixed model, this possibility cannot be dismissed and we plan to investigate it in future work.

Acknowledgments. This work was supported by Deutsche Forschungsgemeinschaft (grant DFG SCHU 2246/10-1, FADeBaC).

[8] A reviewer asks: "If a word is rare, its distributional vector should also be sparse and less informative, which does not guarantee to be a good starting point." This is true and it suggests that it may not be possible to learn a very high-quality representation for a rare word. But this it not our goal. Our goal is simply to learn a *better* representation than the one that is learned by standard word2vec. Our explanation for our positive experimental results is that distributional initialization implements a form of smoothing.

References

Marco Baroni and Alessandro Lenci. 2010. Distributional memory: A general framework for corpus-based semantics. *Computational Linguistics*, 36(4):673–721.

Marco Baroni, Silvia Bernardini, Adriano Ferraresi, and Eros Zanchetta. 2009. The WaCky Wide Web: A Collection of Very Large Linguistically Processed Web-Crawled Corpora. *Language Resources and Evaluation*, 43(3):209–226.

Jiang Bian, Bin Gao, and Tie-Yan Liu. 2014. Knowledge-powered deep learning for word embedding. In *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD*, pages 132–148.

Jan A. Botha and Phil Blunsom. 2014. Compositional morphology for word representations and language modelling. In *Proceedings of the 31st International Conference on Machine Learning (ICML)*, Beijing, China, June.

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

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

Qing Cui, Bin Gao, Jiang Bian, Siyu Qiu, and Tie-Yan Liu. 2014. Knet: A general framework for learning word embedding using morphological knowledge. *Preprint pubslished on arXiv* `arXiv:1407.1687 [cs.CL]`.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2001. Placing search in context: The concept revisited. In *WWW*, pages 406–414.

Felix Hill, Roi Reichart, and Anna Korhonen. 2014. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Preprint pubslished on arXiv* `arXiv:1408:3456 [cs.CL]`.

Hai Son Le, Alexandre Allauzen, Guillaume Wisniewski, and François Yvon. 2010. Training continuous space language models: Some practical issues. In *EMNLP*, pages 778–788.

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

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

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

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

Jeff Mitchell and Mirella Lapata. 2010. Composition in distributional models of semantics. *Cognitive Science*, 34(8):1388–1439.

Yoshiki Niwa and Yoshihiko Nitta. 1994. Co-occurrence vectors from corpora vs. distance vectors from dictionaries. In *COLING*, volume 1, pages 304–309.

Herbert Rubenstein and John B. Goodenough. 1965. Contextual correlates of synonymy. *Commun. ACM*, 8(10):627–633, October.

Magnus Sahlgren. 2008. The distributional hypothesis. *Rivista di Linguistica (Italian Journal of Linguistics)*, 20(1):33–53.

Helmut Schmid. 2000. Unsupervised Learning of Period Disambiguation for Tokenisation. Technical report, IMS, University of Stuttgart.

Hinrich Schütze. 1992. Dimensions of Meaning. In *ACM/IEEE Conference on Supercomputing*, pages 787–796.

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

Peter D. Turney and Michael L. Littman. 2003. Measuring praise and criticism: Inference of semantic orientation from association. *ACM TOIS*, 21(4):315–346.

Peter D. Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research (JAIR)*, 37:141–188.