Addressing Low-Resource Scenarios with Character-aware Embeddings

Sean Papay and **Sebastian Padó** and **Ngoc Thang Vu**
Institut für Maschinelle Sprachverarbeitung
Universität Stuttgart, Germany
{sean.papay,pado,thangvu}@ims.uni-stuttgart.de

Abstract

Most modern approaches to computing word embeddings assume the availability of text corpora with billions of words. In this paper, we explore a setup where only corpora with millions of words are available, and many words in any new text are out of vocabulary. This setup is both of practical interest – modeling the situation for specific domains and low-resource languages – and of psycholinguistic interest, since it corresponds much more closely to the actual experiences and challenges of human language learning and use. We evaluate skip-gram word embeddings and two types of character-based embeddings on word relatedness prediction. On large corpora, performance of both model types is equal for frequent words, but character awareness already helps for infrequent words. Consistently, on small corpora, the character-based models perform overall better than skip-grams. The concatenation of different embeddings performs best on small corpora and robustly on large corpora.

1 Introduction

State-of-the-art word embedding models are routinely trained on very large corpora. For example, Mikolov et al. (2013a) train word2vec on a corpus of 6 billion tokens, and Pennington et al. (2014) report the best GloVe results on 42 billion tokens.

From a language technology perspective, it is perfectly reasonable to use large corpora where available. However, even with large corpora, embeddings struggle to accurately model the meaning of infrequent words (Luong et al., 2013). Moreover, for the vast majority of languages, substantially less data is available. For example, there are only 4 languages with Wikipedias larger than 1 billion words,[1] and 25 languages with more than 100 million words. Similarly, specialized domains even in very high-resource languages are bound to have much less data available.

From a psycholinguistic point of view, current models miss the crucial ability of the human language faculty to generalize from little data. By seventh grade, students have only heard about 50 million spoken words, and read about 3.8 million tokens of text, acquiring a vocabulary of 40,000–100,000 words (Landauer and Dumais, 1997). This also means that any new text likely contains out-of-vocabulary words which students interpret by generalizing from existing knowledge – an ability that plain word embedding models lack.

There are some studies that have focused on modeling infrequent and unseen words by capturing information at the subword and character levels. Luong et al. (2013) break words into morphemes, and use recursive neural networks to compose word meanings from morpheme meanings. Similarly, Bojanowski et al. (2017) represent words as bags of character n-grams, allowing morphology to inform word embeddings without requiring morphological analysis. However, both models are still typically applied to large corpora of training data, with the smallest English corpora used comprising about 1 billion tokens.

Our study investigates how embedding models fare when applied to much smaller corpora, containing only millions of words. Few studies, except Sahlgren and Lenci (2016), have considered this setup in detail. We evaluate one word-based and two character-based embedding models on word relatedness tasks for English and German. We find that that the character-based models mimics human learning more closely, with both better results on small datasets and better performance on rare words. At the same time, a fused representation that takes both word and character level into account yields the best results for small corpora.

[1] https://en.wikipedia.org/wiki/List_of_Wikipedias#Detailed_list (as of 9 Jan 2018)

Proceedings of the Second Workshop on Subword/Character LEvel Models, pages 32–37
New Orleans, Louisiana, June 6, 2018.

WL	
Window size	5
Negative samples	15
Word embedding dim.	300
Minimum word count as inclusion as target	5 for full, 1 for small corpora
Starting learning rate	0.025
Training epochs	5 for full, 15 for 100 MiB, 75 for 10 MiB corpora
FT	
Word embedding dim.	300
Training epochs	5
Learning rate	0.05
Minimum n-gram length	3
Maximum n-gram length	6
Negative samples	5
CL	
Word length	16 characters
Character embedding dim.	15
Convolution filter widths	$\langle 1, 2, 3, 4, 5, 6, 7 \rangle$
Convolution filter units	$\langle 200, 200, 200, 200, 250, 300, 350 \rangle$
Word embedding dim.	300
Minimum word count for inclusion as context	5
Batch size	100
Learning rate	0.05
Training epochs	as above
CAT	
Word embedding dim.	300 + 300 + 300 = 900

Table 1: Hyperparameters (dim. = dimensionality). For WL and FT, software defaults were used for all hyperparameters unless otherwise specified.

2 Models

We examine four models for generating our word embeddings. All of the use a skip-gram objective function but differ in the granularity of linguistic input that they model: the first model works at the word level (WL); the second model, fastText (FT), works at the character n-gram level; the third model is character-based (CL); the forth model is a fusion of the first three (CAT). The hyperparameters of the models are shown in Table 1.

2.1 Word-level Skip-gram Model

As a character-agnostic model, we use a standard, word-level skip-gram model (WL, Mikolov et al. 2013a), with negative sampling loss. All in-vocabulary words are assigned an embedding; out-of-vocabulary words are assigned the vector average of all in-vocabulary embeddings. We use the word2vec software for our WL model[2].

[2] https://code.google.com/archive/p/word2vec/

2.2 fastText

The fastText (FT) model was introduced in Bojanowski et al. (2017). This model is based upon the word-level skip-gram model. However, while WL explicitly stores vectors for each word in the vocabulary, FT learns vector representations for character n-grams which appear within words. The embedding for an individual word is then identified with the sum of that word's n-gram vectors. As unseen words are still composed of familiar n-grams, this model is capable of assigning embedding vectors to words not seen in the training data. We used the fastText software package for this model[3].

2.3 Character-aware Skip-gram Model

Our character-aware skip-gram model (CL) models word meaning by learning representations for individual characters. It consists of two components: the *embedding subnet* generates embeddings for individual words using a convolutional neural network (CNN). The *NCE loss layer* uses a skip-gram loss function to score the embeddings. This architecture allows character-level sequence information to inform word embeddings, while using a loss function similar to those of more traditional embedding architectures.

Embedding Subnet. The *embedding subnet* is a CNN that takes as input a character sequence (representing a word), and outputs an embedding vector for that word. The architecture of this network was adapted from Kim et al. (2016), with modifications to use a skip-gram loss function and to produce low-dimensional embedding vectors.

Figure 1 provides a schematic overview of the embedding subnet. First, input words are normalized to a length of 16 characters, truncating longer words and appending null characters to the end of shorter ones. Each character in this input sequence is then assigned a character-embedding vector. The values of these character embeddings are trainable parameters of the model. The resulting sequence of character embeddings is then used as the input for a convolution layer, with a sigmoid activation function. A max-pooling layer is applied to all outputs of the convolution layer. The results of this pooling are all concatenated to form a single vector, which is then passed through two successive highway networks (Srivastava et al., 2015). Since highway networks preserve dimensionality,

[3] https://github.com/facebookresearch/fastText

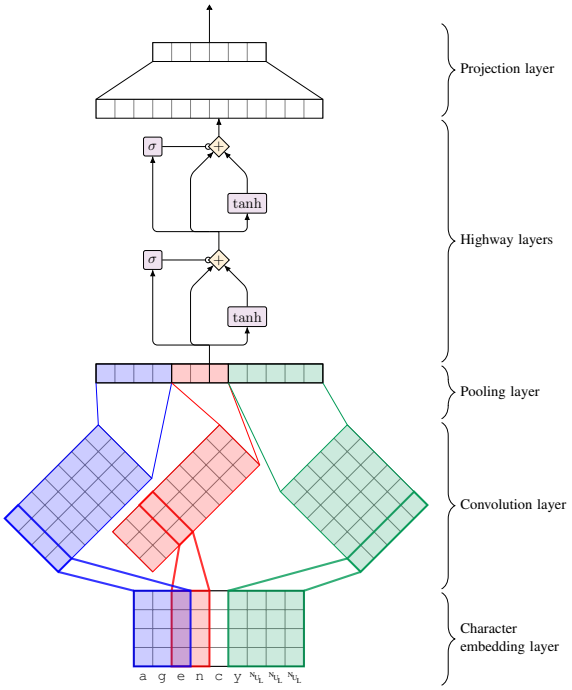

Figure 1: Embedding subnet

the output of the second highway network depends directly on the number of convolution filters used. As we would like a word embedding of relatively small dimensionality, we use a linear projection layer to yield our final embedding vector.

NCE Loss. In order for the embedding subnet to produce semantically-meaningful embeddings, we use it in a skip-gram model (Mikolov et al., 2013a) using noise-contrastive estimation (NCE) loss (Gutmann and Hyvärinen, 2012; Mnih and Teh, 2012). This is rather similar to the NEG models of Mikolov et al. (2013b), but with "input" vectors coming from our embedding subnet, and with NCE in the place of negative sampling. As in Mikolov et al. (2013b), we still depend on a fixed vocabulary of context words for the "output" vectors. All model parameters are optimized to minimize NCE loss using stochastic gradient descent.

2.4 Fusion model

CAT assigns each word the concatenation of its CL, WL, and FT embeddings, i.e., performs late fusion (Bruni et al., 2014). As the individual models produce vectors of different average magnitudes, we rescale the embeddings produced by individual models prior to this concatenation, such that, after rescaling, each constituent model has the same average vector magnitude, when averaged over all words present in the training data. Experiments with joint training of the models did not yield superior results.

Corpus	Articles	Tokens	Vocab	Size
en-full	4,280,642	1,699M	8,745K	8.69 GiB
en-100M	48,430	19,211K	468K	100 MiB
en-10M	4,833	1,924K	106K	10.0 MiB
de-full	1,539,077	587M	6,323K	3.50 GiB
de-100M	43,078	16,507K	720K	100 MiB
de-10M	4,362	1,645K	153K	9.99 MiB

Table 2: Statistics for training corpora.

Benchmark	Mean Freq.	Morphemes per word	Portion OOV
WS353	44.7K	1.27	0/437
RW	1.91K	1.53	19/2951
WS353-de	8.36K	1.30	2/455
GUR350	3.11K	1.46	13/469

Table 3: Statistics for evaluation benchmarks (computed on full-size corpora). Frequencies averaged geometrically (on lemmas with non-zero frequency); morphemes/word averaged algebraically.

3 Experimental Setup

We evaluate our models for two languages, English and German. These are clearly not low-resource languages, but the availability of corpora and evaluation datasets makes them suitable for experiments.

Training data. All models were trained on the standard Wikipedia corpora for English and German preprocessed by Al-Rfou et al. (2013).[4]

In addition, we sampled two (sub)corpora with 10 and 100 million characters to evaluate the models' effectiveness on limited training data. To generate a subcorpus of a particular size, articles were sampled uniformly at random (without replacement) from that language's Wikipedia until the target size was reached. Table 2 presents all corpora and subcorpora used, and their sizes. Our 100M corpora account for around 1% (for English) and 3% (for German) of the full Wikipedia corpora, and 10M corpora for .1% and .3%, respectively. We picked these sizes because they cover the "typical" Wikipedia sizes for low-resource languages.

Evaluation benchmarks. We evaluate our models on a standard task in lexical semantics, predicting human word relatedness ratings. Compared to relation prediction, this task has the advantage of

[4] https://sites.google.com/site/rmyeid/projects/polyglot

Test	Benchmark	size	WL	FT	CL	CAT
all items	[en] WS353	353	.73	**.74**	.51	.72
	[en] RW	2034	.44	**.50**	.31	.46
	[de] WS353	350	**.64**	**.64**	.47	**.64**
	[de] GUR350	350	.61	**.72**	.52	.65
IV items	[en] WS353	353	**.73**	**.73**	.51	.72
	[en] RW	1977	.45	**.50**	.32	.47
	[de] WS353	348	**.64**	**.64**	.47	**.64**
	[de] GUR350	324	.64	**.72**	.55	.67
OOV items	[en] WS353	0	–	–	–	–
	[en] RW	57	-.24	**.64**	-.04	.15
	[de] WS353	2	**1.00**	**1.00**	-1.00	**1.00**
	[de] GUR350	26	-.11	**.61**	.00	.25

Table 4: Spearman correlations of embedding similarity and human judgments (full training sets: en 9G, de 3.5G). Top: full benchmarks; middle: in-vocabulary items; bottom: out-of-vocabulary items. Best model for each benchmark and training bolded.

Test	Benchmark	size	WL	FT	CL	CAT
all items	[en] WS353	353	.65	.66	.42	**.67**
	[en] RW	2034	.22	**.43**	.29	.40
	[de] WS353	350	.49	.50	.30	**.52**
	[de] GUR350	350	.32	**.56**	.51	.47
IV items	[en] WS353	353	.65	.66	.42	**.67**
	[en] RW	1413	.30	**.45**	.33	.42
	[de] WS353	347	.48	.49	.29	**.52**
	[de] GUR350	290	.42	**.55**	.52	.51
OOV items	[en] WS353	0	–	–	–	–
	[en] RW	621	.15	**.41**	.21	.35
	[de] WS353	3	.50	**1.00**	.50	.50
	[de] GUR350	60	.14	**.62**	.41	.41

Table 5: Spearman correlations of embedding similarity and human judgments (100M training sets).

Test	Benchmark	size	WL	FT	CL	CAT
all items	[en] WS353	353	.48	.42	.26	**.53**
	[en] RW	2034	.16	**.32**	.16	.28
	[de] WS353	350	.16	.22	.16	**.29**
	[de] GUR350	350	.11	.34	**.42**	.38
IV items	[en] WS353	333	.52	.44	.29	**.56**
	[en] RW	687	.21	**.34**	.16	.30
	[de] WS353	305	.16	.24	.21	**.28**
	[de] GUR350	218	.29	.40	.32	**.48**
OOV items	[en] WS353	20	**.52**	.05	-.05	.10
	[en] RW	1347	.19	**.30**	.16	.29
	[de] WS353	45	.16	.18	.18	**.24**
	[de] GUR350	132	.08	.25	**.40**	.31

Table 6: Spearman correlations of embedding similarity and human judgments (10M training sets).

being graded instead of categorical (Landauer and Dumais, 1997). We use two relatedness datasets in both languages. For comparison to prior work, and for a rough comparison across languages, we utilize the WordSim353 benchmark (WS353) (Finkelstein et al., 2001) for English and the German version of the Multilingual WordSim353 benchmark (WS353-de) for German (Leviant and Reichart, 2015).

As we are specifically interested in modeling rare words, we also use the Stanford Rare Word Dataset (RW, Luong et al. 2013) for English. It was designed with these goals in mind. While no parallel to this benchmark exists for German, the GUR350 benchmark (Zesch et al., 2007) shows similar properties: As Table 3 shows, the words in GUR350 are less frequent and longer than in WS353-de. Thus, many more words are out of vocabulary in GUR350 even in the full corpus.

4 Results

Tables 4 to 6 show the results for the three different training corpus sizes. We report results for all items, just in-vocabulary items, and just out-of-vocabulary items (i.e., one or both elements of the word pair unseen in training).

Full corpus results. The results by WL on full corpora for all items (top part of Table 4) outperform results reported in the literature[5], indicating that the word-level embeddings are competitive with the state of the art. FT performs as well or even better than WL on the full corpora, indicating that character n-grams can learn well even from large datasets, while the individual character-based CL model cannot profit from this situation. Nevertheless, the fusion model CAT is robust: it performs generally on par with WL.

On full corpora, almost all items in all benchmark datasets are seen; therefore, the separate results on IV and OOV items are not particularly interesting (middle and bottom parts of Table 4).

100M corpora results. The results on the 100M corpora (Table 5) confirm that model performance correlates with training set size: Without exception, the models' performance decreases for smaller corpora. However, this effect is much more pronounced for WL than for the character-based models, FT and CL. For the first time, on these corpora, the fusion model CAT is able to outperform FT, indicating that there is some degree of complementarity between the predictions (and the errors) of

[5]For WordSim, Leviant and Reichart (2015) obtain 0.652 (en) and 0.618 (de) using our "full" corpora. For RW, Sahlgren and Lenci (2016) report 0.285 on 1G words, and for GUR350, Utt and Padó (2014) report 0.42 using 3G words.

the individual models.

On the 100M corpora, the RW and GUR350 datasets both have a significant number of pairs containing OOV words. As expected, WL performed particularly poorly on these pairs. However, it is notable that FT also outperforms WL on every IV benchmark. This shows that the advantage of character-based over word-based models is not restricted to unseen words. The performance gap between FT and WL on IV is small on the two WS353 datasets (with the highest mean item frequency, and the lowest morphological complexity – cf. Table 3) but substantial for RW and GUR350 (which contain low-frequency, morphologically complex words). This indicates that WL struggles in particular with infrequent, complex words.

10M corpora results. Finally, Table 6 shows the results for the 10M corpora. Here, we see a relatively heterogeneous picture regarding the individual models across benchmarks: WL does best on WS353-en; FT does best on RW and WS353-de; CL does best on GUR350. This behavior is consistent with the patterns we found for the 100M corpora, but more marked. Due to the inhomogeneity among models, the fusion model CAT does particularly well, outperforming FT on 3 of 4 benchmarks, often substantially so.

As with the 100M corpora, the character aware models perform much better than WL for OOV pairs. For 10M, however, FT's dominance is not as clear – CL substantially outperforms FT on GUR350. This may indicate that modeling individual character embeddings rather than n-grams is more suitable for the lowest-data setups.

Model choice recommendations. Based on our results, we can formulate the following recommendations: (a) FastText is a good choice for both medium- and large-data situations and is likely to outperform plain word-based models overall, and in particular for low-frequency words; (b) for low-data situations, there is sufficient complementarity among models that model combination, even by simple concatenation, can yield further substantial improvements.

5 Conclusion

This paper argues that it is worthwhile, both from applied and psycholinguistic perspectives, to evaluate embedding models trained on much smaller corpora than generally considered, and have compared a standard word-level skip-gram model against a character n-gram based and a single character-based embedding model.

Even at corpus sizes of billions of words, we find that the character n-gram based model performs at or above the level of the word-level model. This result is in contrast to the findings of Sahlgren and Lenci (2016), who found the best performance for a dimensionality-reduced word embedding model across all corpus sizes. However, all of the models they considered were word-based, indicating that character awareness is what makes the difference. The success of the character n-gram based model can also be interpreted as support for a morpheme-based representation in the mental lexicon (Smolka et al., 2014) in the sense that character n-gram appear to be represent a very informative level of representation for semantic information.

As we move to smaller corpus sizes, we also see more competitive performance for the model based on individual character embeddings. Its forte is to deal with low-data situations, predicting meanings for unfamiliar words by utilizing familiar morphemes and other subword structures, in line with Landauer et al.'s (1997) claim of "vast numbers of weak interrelations that, if properly exploited, can greatly amplify learning by a process of inference". In the future, we will evaluate our character-based model for other languages, and assess other aspects of its psycholinguistic plausibility, such as matching human behavior in performance and acquisition speed (Baroni et al., 2007).

Acknowledgments. Partial funding for this study was provided by Deutsche Forschungsgemeinschaft (project PA 1956/4-1).

References

Rami Al-Rfou, Bryan Perozzi, and Steven Skiena. 2013. Polyglot: Distributed word representations for multilingual NLP. In *Proceedings of CoNLL*. Sofia, Bulgaria, pages 183–192.

Marco Baroni, Alessandro Lenci, and Luca Onnis. 2007. Isa meets lara: An incremental word space model for cognitively plausible simulations of semantic learning. In *Proceedings of the ACL Workshop on Cognitive Aspects of Computational Language Acquisition*. Prague, Czech Republic, pages 49–56.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with

subword information. *Transactions of the Association for Computational Linguistics* 5:135–146.

Elia Bruni, Nam Khanh Tran, and Marco Baroni. 2014. Multimodal distributional semantics. *Journal of Artificial Intelligence Research* 49:1–47.

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*. Hong Kong, China, pages 406–414.

Michael U Gutmann and Aapo Hyvärinen. 2012. Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. *Journal of Machine Learning Research* 13(Feb):307–361.

Yoon Kim, Yacine Jernite, David Sontag, and Alexander M Rush. 2016. Character-aware neural language models. In *Proceedings of AAAI*. Phoenix, AZ, pages 2741–2749.

Thomas K Landauer and Susan T Dumais. 1997. A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review* 104(2):211–240.

Ira Leviant and Roi Reichart. 2015. Separated by an un-common language: Towards judgment language informed vector space modeling. *arXiv preprint arXiv:1508.00106* .

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

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

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*. pages 3111–3119.

Andriy Mnih and Yee Whye Teh. 2012. A fast and simple algorithm for training neural probabilistic language models. *arXiv preprint arXiv:1206.6426* .

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of EMNLP*. Doha, Qatar, pages 1532–1543.

Magnus Sahlgren and Alessandro Lenci. 2016. The effects of data size and frequency range on distributional semantic models. In *Proceedings of EMNLP*. Austin, TX, pages 975–980.

Eva Smolka, Katrin H. Preller, and Carsten Eulitz. 2014. 'verstehen' ('understand') primes 'stehen' ('stand'): Morphological structure overrides semantic compositionality in the lexical representation of german complex verbs. *Journal of Memory and Language* 72:16–36.

Rupesh Kumar Srivastava, Klaus Greff, and Jürgen Schmidhuber. 2015. Highway networks. *arXiv preprint arXiv:1505.00387* .

Jason Utt and Sebastian Padó. 2014. Crosslingual and multilingual construction of syntax-based vector space models. *Transactions of the Association of Computational Linguistics* 2:245–258.

Torsten Zesch, Iryna Gurevych, and Max Mühlhäuser. 2007. Comparing Wikipedia and German Wordnet by evaluating semantic relatedness on multiple datasets. In *Proceedings of NAACL-HLT*. Rochester, NY, pages 205–208.