Evaluation of Word Embeddings for the Social Sciences

Ricardo Schiffers
RWTH Aachen
Aachen, Germany
rschiffers@posteo.de

Dagmar Kern and **Daniel Hienert**
GESIS - Leibniz Institute for the Social Sciences
Cologne, Germany
firstname.lastname@gesis.org

Abstract

Word embeddings are an essential instrument in many NLP tasks. Most available resources are trained on general language from Web corpora or Wikipedia dumps. However, word embeddings for domain-specific language are rare, in particular for the social science domain. Therefore, in this work, we describe the creation and evaluation of word embedding models based on 37,604 open-access social science research papers. In the evaluation, we compare domain-specific and general language models for (i) language coverage, (ii) diversity, and (iii) semantic relationships. We found that the created domain-specific model, even with a relatively small vocabulary size, covers a large part of social science concepts, their neighborhoods are diverse in comparison to more general models Across all relation types, we found a more extensive coverage of semantic relationships.

1 Introduction

Word embedding models learn word representations from large sets of text so that similar words have a similar representation. Models can be used to find semantically related words, for example for applications such as natural language understanding. Technically, word embeddings are distributed representations of words in a vector space (Bengio et al., 2003) so that related words are nearby in the space and can be found with distance measures such as the cosine similarity (Mikolov et al., 2013).

In general, word embedding models are trained on large and general language text collections, e.g., on Web corpora or on Wikipedia dumps. However, there are some initiatives to create and evaluate word embeddings for specific domains on a smaller scale, for example, for computer science (Roy et al., 2017; Ferrari et al., 2017), finance (Theil et al., 2018), patents (Risch and Krestel, 2019), oil & gas industry (Nooralahzadeh et al., 2018; da Silva Magalhães Gomes et al., 2021), and especially in the biomedical domain (Jiang et al., 2015; Chiu et al., 2016; Zhao et al., 2018; Chen et al., 2018; Moradi et al., 2020).

Word embeddings capture "precise syntactic and semantic word relationships" (Mikolov et al., 2013). However, general and domain-specific models can differ much in terms of included specialized vocabulary and semantic relationships (Nooralahzadeh et al., 2018; Chen et al., 2018; da Silva Magalhães Gomes et al., 2021). Intrinsic evaluation methods are used to test models for these relationships (Schnabel et al., 2015; Gladkova and Drozd, 2016). In this work, we focus on creating and evaluating word embeddings for the social science domain and comparing them to general language models.

Word embeddings are used in the social sciences domain for a number of NLP tasks. Matsui and Ferrara (2022) provide an overview of word embeddings techniques and applications in the social sciences based on a literature review. Word embeddings are used, for example, for the extraction of trends of biases or culture from data (Caliskan et al., 2017), using vectors to define working variables that embody the concept or research questions (Toubia et al., 2021), or use reference words and their semantic neighborhoods to analyze gender terms and its relation to specific occupations (Garg et al., 2018). Other applications are the processing of scientific documents, for example the extraction of acknowledged entities from full texts (Smirnova and Mayr, 2022). For the retrieval of specialized information, word embeddings can be used for query expansion (Roy et al., 2022). All these applications depend on meaningful word embeddings. The more precise the specialized language is available in the vector space and the better related terms are arranged in the vector space, the better the applications work.

Proceedings of the 6th LaTeCH-CLfL Workshop, pages 1–6
October 16, 2022. ©2022 International Conference on Computational Linguistics

2 Generation of Social Science Word Embeddings

2.1 Corpora and Pre-processing

The Social Science Open Access Repository (SSOAR)[1] is a document server that makes scientific articles from the social sciences freely available. At the time of this work, it contained 58,883 documents from which 37,604 were directly machine-readable. The rest was included via links and was not directly accessible for us. Most of the texts are written in German, followed by English-language ones. The publication years were mainly in the 2000s (*min*=1923, *max*=2020, *M*=2006, *SD*=9.36).

Extracting raw text from PDF files has proven to be an error-prone process, but is, on the other side, a crucial part for the creation of word embeddings. A pre-evaluation with standard python parsers showed massive problems, e.g., with word separation. We evaluated five different PDF parsers (PyPDF2, PyPDF4, PDFMiner, PDFBox, Tika) with a random sample of 238 documents taken from different publication years and with documents producing a lot of errors. PDFBox[2] showed the best results. Since it has been meanwhile integrated in Apache Tika,[3] we chose this library for further processing.

From the extracted raw texts, we built language-specific corpus files by applying a number of cleaning steps. All texts were cleaned from the cover pages, which are included in every SSOAR document. All hyphens and line breaks were removed, camel cases were separated using regular expressions. We used a line-based identification of the language based on word embeddings for language identification,[4] which helps to maximise content for a specific language. Subsequently, numbers in numeric form are converted to word numbers, multiple spaces are merged, and all characters are written in lower case. We use sentence-wise deduplication based on a hash value and sort out duplicate sentences. Finally, all texts were tokenized. As a result, we got two corpus files for the German and English languages. Table 1 gives an overview of the count of tokens, vocabulary size, number of raw data files, and the file sizes.

[1] https://www.gesis.org/ssoar/home
[2] https://github.com/apache/pdfbox
[3] https://github.com/apache/tika
[4] https://fasttext.cc/docs/en/language-identification.html

	ssoar.de.txt	**ssoar.en.txt**
Tokens	152,341,432	92,123,735
Vocabulary	1,678,657	367,574
Files	25,227	23,045
MB	1,076.31	540.49

Table 1: German and English corpus data files from n=37,604 SSOAR documents.

2.2 Training of Word Embeddings

We rely on the fastText model (Bojanowski et al., 2017) to train word embeddings. It uses a character-based model based on the word-based skipgram model (Mikolov et al., 2013). The representation of a word is the sum of its n-grams with a default size between three and six characters. German-language word embeddings benefit from using such a model due to the frequent occurrence of compounds which can be captured with longer character sequences (Bojanowski et al. 2017).

We used the fastText Python module for implementation. During the training, word embeddings with different dimensions $(100, 150, 200, 300, 500)$ were created since the dimensionality of the models is a crucial parameter for the evaluation applied here. We used default values for the other hyperparameters: For the number of iterations of the data set, we apply five epochs, a learning rate of 0.05, five negative examples and a context window of five by using the skip-gram model. The resulting word embeddings are open-source and can be downloaded.[5]

2.3 Reference Knowledge Resources

In this evaluation, we aim to understand the impact of domain-specific language on the availability of specialized terms and semantic relations in the models. We use the thesaurus for the social sciences (TheSoz, Zapilko et al. 2013) as a reference knowledge resource. It contains 36,320 keywords with 5,986 descriptors, including the relations *broader*, *narrower*, *related*, *altLabel*, and 30,334 non-descriptors that are either used synonymously or represent more general terms related to a descriptor. Figure 1 shows the descriptor *social inequality* with its related concepts.[6]

Additionally, as reference models and as a baseline, we use the German word embeddings models *wiki.de*[7] offered by FastText and the fasttext model

[5] https://zenodo.org/record/5645048
[6] http://lod.gesis.org/thesoz/concept_10038124
[7] https://fasttext.cc/docs/en/pretrained-vectors.html

Figure 1: The descriptor *social inequality* in its environment of broader, narrower, and related terms

deepset.de[8] from Deepset. Both are trained on the German Wikipedia with the skip-gram-model and with 300 dimensions. Remaining with the example of "social inequality", this concept has its own Wikipedia page[9]. Links to other concepts result from the full text, the links in the text or the Wikipedia category system.

3 Evaluation

In what follows, we evaluate word embeddings trained on social science language versus those trained on general language. We want to understand the effects on domain-specific language coverage, diversity, and semantic relationships. The evaluation is performed with the German models, since a larger part of the source texts is written in German. These models are called *ssoar.de* in the remainder of this paper.

3.1 Coverage

To evaluate the coverage of the models' language with respect to the social science domain, keywords x_i from the TheSoz are iteratively compared with words in the vocabularies V (see Nooralahzadeh et al. 2018 for a similar method). Ratio string similarity from the Levenshtein Python C extension module[10] was used for the calculation of the word's similarity. We applied different thresholds $s = 0.9$, $s = 0.95$ and $s = 1$ to find identical but also very similar terms. In the case of compound descriptors, the result is only valid if all terms of a TheSoz entry are included in the vocabulary of the model. Since all ssoar.de models with different dimenions are based on the same text corpus, the vocabulary is identical, and we use the smallest model with $dimension = 100$. We used formula (1) to compute the coverage c for all $n = 36,320$ keywords.

$$c = \frac{\sum_{i=1}^{n} x_i \in V}{n} \tag{1}$$

	ssoar.de	wiki.de	deepset.de
Vocab size	403,452	2,275,233	1,319,232
s=0.9	87.95	**92.22**	63.31
s=0.95	84.51	**90.08**	60.54
s=1.0	82.63	**88.80**	59.36

Table 2: Coverage of TheSoz keywords in the vocabulary of different models (n=36,320 keywords)

Table 2 shows the results. The model wiki.de shows a coverage in 88%-92%, ssoar.de in 82%-88% and deepset.de only in 59%-63%. Thus, wiki.de shows the best results, but also has a vocabulary size five times larger. The other way around, deepset is three times larger in vocabulary size but shows worse results. This suggests that similarly good results for covering domain-specific language can be obtained with a small model trained on specialized texts compared to larger general language models.

3.2 Diversity

To determine the diversity d of a model relative to other models, we compare the neighbors related to a TheSoz keyword. The procedure for determining diversity is described in Formula 2. For this purpose, the nearest neighbors of the keywords x_i of two models (A and B) are compared. If the intersection between the neighbors A_{x_i} and B_{x_i} corresponds to the empty set, the diversity between the compared models increases with a return value $1 = true$ and $0 = false$. Here, the number of neighbors returned by the models is limited by the $top\text{-}k$ entries. To obtain the relative diversity, the result is then divided by the number of keywords n to be tested. This ensures comparability between the different results.

$$d = \frac{\sum_{i=1}^{n} A_{x_i} \cap B_{x_i} = \emptyset}{n} \tag{2}$$

[8] https://deepset.ai/german-word-embeddings
[9] https://de.wikipedia.org/wiki/Soziale_Ungleichheit
[10] https://github.com/ztane/python-Levenshtein

top-k	Model	ssoar.100	ssoar.150	ssoar.200	ssoar.300	ssoar.500	wiki	deepset
10	ssoar.100	-	0.23	0.19	0.47	1.19	21.59	**44.30**
	ssoar.150	0.23	-	0.10	0.17	0.44	19.52	42.52
	ssoar.200	0.19	0.10	-	0.08	0.17	18.81	42.29
	ssoar.300	0.47	0.17	0.08	-	0.07	17.94	41.84
	ssoar.500	1.19	0.44	0.17	0.07	-	17.56	41.63
	wiki	21.59	19.52	18.81	17.94	17.56	-	25.81
	deepset	**44.30**	42.52	42.29	41.84	41.63	25.81	-
50	ssoar.100	-	0.05	0.05	0.05	0.06	8.12	**20.20**
	ssoar.150	0.05	-	0.05	0.05	0.06	7.16	19.07
	ssoar.200	0.05	0.05	-	0.05	0.05	6.71	18.85
	ssoar.300	0.05	0.05	0.05	-	0.05	6.31	18.53
	ssoar.500	0.06	0.06	0.05	0.05	-	6.06	18.28
	wiki	8.12	7.16	6.71	6.31	6.06	-	5.86
	deepset	**20.20**	19.07	18.85	18.53	18.28	5.86	-
200	ssoar.100	-	0.05	0.04	0.05	0.05	3.82	**7.70**
	ssoar.150	0.05	-	0.05	0.05	0.05	3.40	7.50
	ssoar.200	0.04	0.05	-	0.05	0.04	3.06	7.31
	ssoar.300	0.05	0.05	0.05	-	0.05	2.92	7.06
	ssoar.500	0.05	0.05	0.04	0.05	-	2.79	7.00
	wiki	3.82	3.40	3.06	2.92	2.79	-	1.20
	deepset	**7.70**	7.50	7.31	7.06	7.00	1.20	-

Table 3: Diversity between models (n=36,320 TheSoz keywords)

Table 3 shows the results of the evaluation method described for all models, including the reference models and for the $top\text{-}k$ $10, 50, 200$ neighbors. When looking at the results, it is noticeable that the diversity between the ssoar models and the reference model deepset.de consistently shows the greatest differences. When compared with the wiki.de model, the diversity to the ssoar.de model is still high but shows roughly only half of the values before. For the smallest $top\text{-}k$ ($k = 10$), the diversity between the two reference models is even higher (25.81) than it is when comparing the ssoar.de models with the wiki.de model (21.59). As expected, the diversity decreases with increasing $top\text{-}k$ entries.

In addition, it can be seen that the use of fewer dimensions in the training of the ssoar.de models has a positive effect on diversity. The ssoar.de model with the dimensionality 100 (ssoar.100) has the greatest diversity across all $top\text{-}k$.

3.3 Relations

To measure relational coverage r for the social science domain, we used an evaluation method inspired by the intrinsic evaluation of comparing semantic relations of established knowledge resources (Nooralahzadeh et al., 2018; Chen et al., 2018; da Silva Magalhães Gomes et al., 2021). A data set of TheSoz was used to determine the coverage of the relations. Related concepts were assigned to the descriptors using different relations: the relation *broader* describes superordinate concepts to the descriptor (Hypernyms). With *narrower* terms, concepts subordinate to the descriptor are distinguished (Hyponyms), and *related* refers to related terms. The relation *altLabel* describes concepts that can be used alternatively to the descriptor. These relations are based on the standard Simple Knowledge Organization System (SKOS, cf. Zapilko et al. 2013).

Equation 3 shows the basis for calculating the relational coverage of a model. The test is whether the concept $k(x_i)$ associated with the descriptor x_i is contained in the neighborhood set N_{x_i} with the return value $1 = true$ and $0 = false$. The number of neighbors returned by the models is limited to $top\text{-}k$. Finally, dividing the sum of found concepts and the total set of used descriptors n yields the domain-specific coverage of the model. The described procedure is performed per available relation type.

$$r = \frac{\sum_{i=1}^{n} k(x_i) \in N_{x_i}}{n} \tag{3}$$

For the evaluation, concepts consisting of multiple words were not considered since the neighborhood query returns only single words. In addition, only descriptors that are annotated in German language were applied. Accordingly, a total of $14,998$ out of $35,473$ descriptor-concept pairs were used, which in turn were subdivided by type of relations. The coverage of the relations was determined with neighborhoods for different $top\text{-}k$ entries.

The results in Table 4 show that the ssoar.de models perform better than the models used for comparison across all $top\text{-}k$. Only for the *broader* relation with $top\text{-}k = 10$, the deepset.de model performs better than the ssoar.de models. The comparison with the reference models indicates a real specialization with respect to the social science domain. Deepset.de achieves better results for *broader* relations for $top\text{-}k = 10$, but the superiority fades away at larger neighborhoods. The results are better than those of the reference models above a $top\text{-}k$ of 50 across all relation types.

Comparing the ssoar.de models to each other, it is noticeable that word embeddings trained on smaller dimensions perform better at smaller $top\text{-}k$ than models with more dimensions. For larger neighborhoods, more dimensions tend to be pre-

top-k	Model	bro	nar	rel	alt
10	ssoar.100	8.54	**6.06**	**10.28**	**13.27**
	ssoar.150	9.20	5.67	9.91	12.59
	ssoar.200	9.39	5.49	9.40	12.47
	ssoar.300	9.20	4.78	9.30	11.68
	ssoar.500	8.81	4.34	8.08	10.34
	wiki.de	5.77	3.39	8.05	9.87
	deepset.de	**13.33**	5.17	9.91	8.12
50	ssoar.100	25.03	**21.00**	25.94	27.12
	ssoar.150	26.02	20.46	26.61	27.63
	ssoar.200	27.11	20.76	**26.89**	27.81
	ssoar.300	27.96	20.31	25.43	**28.08**
	ssoar.500	**28.73**	19.16	22.79	26.53
	wiki.de	19.13	14.02	21.34	23.89
	deepset.de	23.57	15.27	19.14	15.02
200	ssoar.100	43.36	40.04	42.34	42.6
	ssoar.150	45.81	41.46	44.17	44.66
	ssoar.200	47.73	41.64	44.57	45.11
	ssoar.300	49.06	**41.67**	**44.88**	45.96
	ssoar.500	**49.72**	41.25	43.25	**46.16**
	wiki.de	36.84	34.01	36.79	40.65
	deepset.de	33.01	24.89	29.49	21.93

Table 4: Relational coverage of all models (n=14,998 descriptor-concept pairs)

ferred. Nooralahzadeh et al. (2018); Chiu et al. (2016) report generally better performance for increasing dimensions, but we found that it also depends on the number of $top\text{-}k$.

4 Conclusion

In this work, we built domain-specific word embeddings for the social sciences and compared them to general language models. First, we checked for coverage of specialized language keywords. Wiki.de performed best with 92%, but ssoar.de followed closely with 88% with only one-fifth of vocabulary size. We then analysed the diversity of models to each other by comparing selected neighbourhoods: domain-specific and general-language models showed the highest diversities. However, diversity decreases with a larger number of returned neighbors. Concerning relational coverage, ssoar.de models performed best in all settings, except for the broader relation with $top\text{-}k = 10$. In summary, the word embeddings produced in this work showed much better results than the general language models when compared to established knowledge resources such as a thesaurus. Domain-specific word embeddings can improve the semantic relatedness metric and applications build upon. This is in line with related works (e.g., Nooralahzadeh et al. 2018; Chen et al. 2018; da Silva Magalhães Gomes et al. 2021) showing for other domains that domain-specific models can better capture semantic relations - even with a small corpus size (e.g., Zhao et al. 2018).

The experiments showed that the underlying texts and their language have a significant impact on the resulting word embeddings. This is even more true for the applications that are based on them. (1) *Coverage* of social science concepts is very different depending on the word embedding model. For example, applications that want to define and extend a working variable depend directly on the concepts contained in the model. (2) The models can be very *diverse* in terms of their semantic neighbors. Applications based on them, for example, query expansion or reference words and their neighborhoods, lead to different results depending on the model. (3) For *relational coverage*, the domain-specific model contains more relations in the sense of a domain thesaurus. This may be important, for example, to keep the precision of search results high in query term expansion or to keep working variables precise in expansion. In summary, the performance of applications is directly dependent on the alignment of word embeddings, their underlying language and their domain.

In future work, we want to compare the effects of specialized language with other embedding models such as Word2Vec, GloVe, or contextual embeddings such as BERT.

References

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural probabilistic language model. *J. Mach. Learn. Res.*, 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*, 5:135–146.

Aylin Caliskan, Joanna J Bryson, and Arvind Narayanan. 2017. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334):183–186.

Zhiwei Chen, Zhe He, Xiuwen Liu, and Jiang Bian. 2018. Evaluating semantic relations in neural word embeddings with biomedical and general domain knowledge bases. *BMC Medical Informatics Decis. Mak.*, 18(S-2):53–68.

Billy Chiu, Gamal K. O. Crichton, Anna Korhonen, and Sampo Pyysalo. 2016. How to train good word embeddings for biomedical NLP. In *Proceedings of the 15th Workshop on Biomedical Natural Language Processing, BioNLP@ACL 2016, Berlin, Germany,*

August 12, 2016, pages 166–174. Association for Computational Linguistics.

Diogo da Silva Magalhães Gomes, Fábio Corrêa Cordeiro, Bernardo Scapini Consoli, Nikolas Laceda Santos, Viviane Pereira Moreira, Renata Vieira, Silvia Moraes, and Alexandre Gonçalves Evsukoff. 2021. Portuguese word embeddings for the oil and gas industry: Development and evaluation. *Comput. Ind.*, 124:103347.

Alessio Ferrari, Beatrice Donati, and Stefania Gnesi. 2017. Detecting domain-specific ambiguities: An NLP approach based on wikipedia crawling and word embeddings. In *IEEE 25th International Requirements Engineering Conference Workshops, RE 2017 Workshops, Lisbon, Portugal, September 4-8, 2017*, pages 393–399. IEEE Computer Society.

Nikhil Garg, Londa Schiebinger, Dan Jurafsky, and James Zou. 2018. Word embeddings quantify 100 years of gender and ethnic stereotypes. *Proceedings of the National Academy of Sciences*, 115(16):E3635–E3644.

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, RepEval@ACL 2016, Berlin, Germany, August 2016*, pages 36–42. Association for Computational Linguistics.

Zhenchao Jiang, Lishuang Li, Degen Huang, and Liuke Jin. 2015. Training word embeddings for deep learning in biomedical text mining tasks. In *2015 IEEE International Conference on Bioinformatics and Biomedicine, BIBM 2015, Washington, DC, USA, November 9-12, 2015*, pages 625–628. IEEE Computer Society.

Akira Matsui and Emilio Ferrara. 2022. Word embedding for social sciences: An interdisciplinary survey.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, page 3111–3119, Red Hook, NY, USA. Curran Associates Inc.

Milad Moradi, Maedeh Dashti, and Matthias Samwald. 2020. Summarization of biomedical articles using domain-specific word embeddings and graph ranking. *J. Biomed. Informatics*, 107:103452.

Farhad Nooralahzadeh, Lilja Øvrelid, and Jan Tore Lønning. 2018. Evaluation of domain-specific word embeddings using knowledge resources. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, Miyazaki, Japan. European Languages Resources Association (ELRA).

Julian Risch and Ralf Krestel. 2019. Domain-specific word embeddings for patent classification. *Data Technol. Appl.*, 53(1):108–122.

Arpita Roy, Youngja Park, and Shimei Pan. 2017. Learning domain-specific word embeddings from sparse cybersecurity texts. *CoRR*, abs/1709.07470.

Dwaipayan Roy, Mandar Mitra, Philipp Mayr, and Amritap Chowdhury. 2022. Local or global? a comparative study on applications of embedding models for information retrieval. In *5th Joint International Conference on Data Science Management of Data (9th ACM IKDD CODS and 27th COMAD)*, CODS-COMAD 2022, page 115–119, New York, NY, USA. Association for Computing Machinery.

Tobias Schnabel, Igor Labutov, David M. 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, September 17-21, 2015*, pages 298–307. The Association for Computational Linguistics.

Nina Smirnova and Philipp Mayr. 2022. Evaluation of embedding models for automatic extraction and classification of acknowledged entities in scientific documents. In *Proceedings of the EEKE workshop at JCDL*.

Christoph Kilian Theil, Sanja Stajner, and Heiner Stuckenschmidt. 2018. Word embeddings-based uncertainty detection in financial disclosures. In *Proceedings of the First Workshop on Economics and Natural Language Processing, ECONLP@ACL 2018, Melbourne, Australia, July 20, 2018*, pages 32–37. Association for Computational Linguistics.

Olivier Toubia, Jonah Berger, and Jehoshua Eliashberg. 2021. How quantifying the shape of stories predicts their success. *Proceedings of the National Academy of Sciences*, 118(26):e2011695118.

Benjamin Zapilko, Johann Schaible, Philipp Mayr, and Brigitte Mathiak. 2013. TheSoz: A SKOS representation of the thesaurus for the social sciences. *Semantic Web journal (SWJ)*, 4(3):257–263.

Mengnan Zhao, Aaron J. Masino, and Christopher C. Yang. 2018. A framework for developing and evaluating word embeddings of drug-named entity. In *Proceedings of the BioNLP 2018 workshop, Melbourne, Australia, July 19, 2018*, pages 156–160. Association for Computational Linguistics.