Towards Bilingual Word Embedding Models for Engineering

Evaluating Semantic Linking Capabilities of Engineering-Specific Word Embeddings Across Languages

Tim Schopf*
Technical University of Munich, Department of Informatics
tim.schopf@tum.de

Peter Weinberger*
Technical University of Munich, Department of Informatics
peter.weinberger@tum.de

Thomas Kinkeldei
ROKIN GmbH
thomas.kinkeldei@rokin.tech

Florian Matthes
Technical University of Munich, Department of Informatics
matthes@tum.de

ABSTRACT

Word embeddings represent the semantic meanings of words in high-dimensional vector space. Because of this capability, word embeddings could be used in a wide range of Natural Language Processing (NLP) tasks. While domain-specific monolingual word embeddings are common in literature, domain-specific bilingual word embeddings are uncommon. In general, large text corpora are required for training high quality word embeddings. Furthermore, training domain-specific word embeddings necessitates the use of source texts from the relevant domain. To train bilingual domain-specific word embeddings, the domain-specific texts must also be available in two different languages. In this paper, we use a large dataset of engineering-related articles in German and English to train bilingual engineering-specific word embedding models using different approaches. We will evaluate our trained models, identify the most promising approach, and demonstrate that the best performing one is very capable of representing semantic relationships between engineering-specific words and mapping languages in a shared vector space. Moreover, we show that the additional use of an engineering-specific learning dictionary can improve the quality of bilingual engineering-specific word embeddings.

CCS CONCEPTS

• **Information systems** → Information retrieval; Retrieval models and ranking; Similarity measures; Information retrieval; Evaluation of retrieval results; Relevance assessment; Information retrieval; Retrieval tasks and goals; Recommender systems; • **Computing methodologies** → Artificial intelligence; Natural language processing; Language resources.

*Authors contributed equally to this work.

MSIE 2022, April 28–30, 2022, Chiang Mai, Thailand

ACM ISBN 978-1-4503-9581-6/22/04…$15.00
https://doi.org/10.1145/3535782.3535835

KEYWORDS

Bilingual Word Embeddings, Engineering, Natural Language Processing

ACM Reference Format:
Tim Schopf, Peter Weinberger, Thomas Kinkeldei, and Florian Matthes. 2022. Towards Bilingual Word Embedding Models for Engineering: Evaluating Semantic Linking Capabilities of Engineering-Specific Word Embeddings Across Languages. In *2022 4th International Conference on Management Science and Industrial Engineering (MSIE) (MSIE 2022), April 28–30, 2022, Chiang Mai, Thailand*. ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3535782.3535835

1 INTRODUCTION

Digital platforms changed our lives significantly by helping us to access needed information fast and easily. However, there is no suitable tool available for engineers looking for technologies to solve engineering-specific problems. In the engineering domain, the same tools that were used 30 years ago are still used today. This typically includes exhibitions, specialized journals, or search engines. To find new technologies, most search engines require keywords as input. However, users who search for engineering technologies to solve their problems, frequently do not know what keywords to use at first. Therefore, they e.g. first use search keywords that either describe the problem or describe an already known technology that has similar characteristics as the desired one. Following that, users review the search results and modify their keywords based on the newly gathered information for a new search. This iterative procedure is repeated until a suitable technology is discovered. To assist engineers during their research, engineering-specific word embedding models can suggest related words to previously defined search keywords. The suggested words could refer to previously unknown technologies that contain the required features, significantly speeding up the search process.

Furthermore, for non-native English speakers, engineering-specific information is sometimes not available in their native language, but only in English or vice versa. As a result, engineers must search for new technologies in multiple languages, complicating the process further. Bilingual engineering-specific word embedding models have the potential to assist in the discovery of new technologies by suggesting related search keywords for previously defined keywords in multiple languages.

In this paper, we compare and contrast various methods for training bilingual engineering-specific word embedding models and determine which are the most promising ones. This enables us to assist engineers in their search for new technologies by suggesting related search terms to previously defined keywords in two languages at the same time.

2 RELATED WORK

Word embeddings are vector representations of words that capture their semantic information and are used as input for a variety of different NLP downstream tasks such as text classification [1] or document retrieval [2]. Word embeddings can be trained using a variety of methods, including Word2Vec [3], GloVe [4], fastText [5] and ELMo [6]. Although most word embeddings are typically trained on generic text corpora, there have been some attempts to train domain-specific word embeddings as well. Efstathiou et al. [7] published a word embedding model for the software engineering domain trained on over 15GB of textual data from Stack Overflow[1] posts. Chalkidis and Kampas [8] trained a legal word embedding model called Law2Vec trained on a large number of legal corpora from various public legal sources. Wang et al. [9] empirically evaluated word embeddings trained on biomedical publications and showed that the semantic similarity captured by the domain-specific word embeddings is closer to human experts' judgments than domain unspecific ones. When compared to a domain unspecific model, the domain-specific word embedding model for engineering trained by Braun et. al. [10] demonstrated an improved performance on engineering related NLP tasks.

Bilingual word embeddings are semantic word representations associated across two languages embedded in the same vector space [11]. Transvec [12] learns a linear mapping between language vector spaces using ridge regression and distributed representation of words. By using linear algebra and computing a singular value decomposition, VecMap [13] [14] [15] [16] tries to find an orthogonal projection to map monolingual word embeddings onto each other. VecMap can create bilingual word embeddings in supervised, semi-supervised as well as in unsupervised learning settings. After using a Generative Adversarial Network (GAN) with the objectives of the discriminator and generation of a mapping matrix, MUSE [17] produces an approximate projection of monolingual vector spaces. MUSE mappings can be trained supervised or unsupervised. For bilingual word embedding learning, BiLex [18] uses publicly available lexical definitions. Most pervious approaches required pre-defined seed lexica, but this eliminates that requirement.

Evaluations of word embeddings face the problem of being based on very subjective approaches, making it difficult to evaluate word embeddings objectively [19]. The performance of word embeddings is determined not only by their ability to reflect contextual similarity but also by how well they perform when used as input for specific downstream tasks. In general, word embedding evaluations can be classified as either extrinsic or intrinsic approaches [20]. Extrinsic approaches evaluate word embeddings on their ability to be used as input for downstream machine learning tasks. Intrinsic approaches evaluate word embeddings independently of specific natural language processing downstream tasks. They rely on human judgment of their ability to represent word relations instead. The outcomes of extrinsic evaluations are heavily reliant on the defined downstream tasks. However, the defined downstream tasks aren't always applicable to the intended use case. Additionally, labeled data is required to perform at least semi-supervised downstream tasks. In contrast, intrinsic evaluation approaches rely solely on human judgment, which is highly subjective and vulnerable to bias based on the evaluator. Therefore, extrinsic as well as intrinsic evaluation approaches, in general, are discussed controversially in literature [19].

[1]https://stackoverflow.com/

3 DATA

The used document dataset consists of 3,132,016 engineering-related articles in German and English. It is distributed slightly unevenly with 1,895,025 English and 1,236,991 German documents. We collect articles from more than 100 specialized web journals in the domain of mechanical and electrical engineering from 2017 to 2021. The data sources chosen, such as "engineering.com" or "elektronik.de" are widely used and well-known in the engineering domain. The document dataset includes topics about additive manufacturing, virtual reality, robotics, automation, production process technologies, and more. Each document consists of title, text body, source URL, and publication date. When extracting all distinct words for each language, we receive an English vocabulary size of 338,661 and a German vocabulary size of 530,070.

Multiple algorithms that map monolingual word embeddings from two different languages into one shared embedding space use supervised or semi-supervised approaches to create bilingual word embeddings. As a result, we require a learning dictionary with word translations from German to English and vice versa in addition to our document dataset. Furthermore, the learning dictionary has to focus on engineering-related words to create an ideal mapping for our domain of interest. As a result, we make use of the technical online dictionary eLengua[2], which contains engineering-specific word pairs in both German and English. We download the entire dictionary and use BeautifulSoup[3] and regular expressions to process it. For simplicity, we only use single words in our embedding model. Therefore, we reject translations that contain compound words in either German or English. Furthermore, we create separate word pairs if words have multiple translations. Thereby, we transform for example the translation ("klebrigkeit"–"stickiness"/"tackiness") to the two word pairs ("klebrigkeit"–"stickiness") and ("klebrigkeit"–"tackiness"). We apply this method bidirectionally to include all word pair variations and include as much knowledge as possible. Our final learning dictionary consists of 655 engineering-specific word pairs in German and English.

4 TRAINING

4.1 Word Embeddings

We choose Word2Vec for monolingual word embedding training, as the algorithm creates word embeddings based on their surrounding context. As a result, word embeddings are treated as related if they appear in a similar context. This behavior is consistent with the use

[2]https://elengua.de/blog/technisches-woerterbuch-deutsch-englisch/
[3]https://www.crummy.com/software/BeautifulSoup/

case described in Section 1, as we want to suggest related technology keywords to previously defined search keywords. Furthermore, we do no need word embeddings that also consider the word order within sentences because our use case is strictly keyword-based. In addition, we use the full vocabularies from our German and English document datasets for word embedding training. This enables us to include less common words, which is useful when looking for words that are related to rare technologies. Compared to the common vocabulary size ranging from 10,000 to 50,000 [21], ours is therefore considerably larger. We add additional prefixes to each word before word embedding training so that we can later determine their language origin. We use the prefix "de_" for German words and "en_" for English words. As training algorithm, we use gensim's[4] Word2Vec implementation, changing the embedding dimensionality to 300, the window size to 10, and the number of epochs to 500. For both languages, we train a separate word embedding model on the respective document corpus. Using 32 CPU cores in parallel results in a training duration of approximately 5 days for each word embedding model.

4.2 Bilingual Mappings

To map bilingual word embeddings in a shared vector space, we evaluate three different approaches during our experiments. Using the combined German and English document corpora, we first train a naïve bilingual Word2Vec model. The pretrained monolingual word embeddings from Section 4.1 are mapped into a shared bilingual vector space using MUSE in our second approach. Using the same monolingual word embeddings as for MUSE, we also assess VecMap's bilingual mapping ability. We focus on the evaluation of MUSE and VecMap's bilingual mapping ability for the engineering domain because they produced the most promising results in previous experiments.

4.2.1 Naïve Bilingual Model. We train the naïve bilingual model similarly to the monolingual word embedding models. As a result, we combine the German and English document corpora and train a Word2Vec model on 500 epochs with a window size of 10 and an embedding dimensionality of 300. The resulting vocabulary size now is even larger than that used for the monolingual embedding models. The merged document dataset consists of 62,772,826 distinct sentences.

4.2.2 MUSE. For the supervised approach, we use our constructed engineering learning dictionary and evaluate the resulting mappings after 5 and 20 refinement iterations to examine how additional refinements influence the results. The effect of additional refinements fades significantly after the first iteration according to the MUSE authors. As a result, we do not anticipate a significant performance difference between the supervised MUSE mappings of 5 and 20 refinements. For the unsupervised mapping approach, we evaluate the resulting mappings that were trained with the default parameters of 5 adversarial training epochs and 5 refinements. For all approaches, the unsupervised as well as the supervised ones, we train the algorithm to map vector spaces from English as source embedding space into German as the target embedding space with an embedding dimensionality of 300.

[4]https://radimrehurek.com/gensim/

```
fencing Gymnastik 2.17
volcano Magma 2.21
lizard Krokodil 3.13
episode Kapitel 3.17
Times Guardian 3.17
Latin Deutsch 2.92
Clinton Obama 3.04
Jupiter Merkur 3.21
evolution Darwin 2.25
essay Hausaufgaben 2.33
```

Figure 1: First 10 word pairs of the word similarity evaluation dataset (English to German) of . The columns consist of English words, German words, similarity scores and are tab separated.

After each refinement iteration, MUSE internally evaluates the current mapping performance on third party similarity scores. For example, one of the evaluation scores is based on [22] that manually annotated similarity scores of bilingual word pairs on a scale from 0 (totally dissimilar) to 4 (very similar). Figure 1 illustrates an example of this word similarity dataset.

For evaluation, MUSE computes Pearson correlation coefficients between the manually generated dataset scores and the scores created by its own mapping. As the evaluation data contains no language-specific prefixes, we remove them before MUSE training and reintroduce them afterward. Another significant issue with using this third-party evaluation data is that they only contain correctly spelled words from a wide range of fields, with no abbreviations or technical jargon. This contradicts our source texts from the engineering domain and results in an information loss during the internal evaluation steps.

4.2.3 VecMap. For VecMap training, we use the default parameters recommended by the authors to map our word embeddings with the supervised, semi-supervised and unsupervised approaches. Furthermore to our monolingual word embeddings, we use the constructed engineering learning-specific dictionary for supervised and semi-supervised VecMap training. Similar to MUSE training, we train the supervised, semi-supervised and unsupervised algorithms to map vector spaces from English as source embedding space into German as target embedding space.

5 EVALUATION

Our use case of interest does not have a clear natural language processing downstream task. As a result, we use intrinsic evaluation approaches to evaluate the resulting bilingual word embeddings. In our evaluation, we want to examine two different aspects of the resulting bilingual word embeddings. To begin, we want to assess their ability to represent the semantic relationship between words. As a result, embeddings of semantically similar words should be close to each other in vector space. Second, we want to evaluate the quality of bilingual vector space mappings between word embeddings originating from two different languages. Thereby, translations of words should be close to each other in the shared vector space. Hence, we employ two different intrinsic approaches to assess both aspects.

Table 1: Results of the coherence evaluation. Highlighted rows indicate the Top-2 performing models, averaged over all neighborhoods.

Mapping	3-NN	5-NN	10-NN
Naïve Bilingual Model	**1.00**	**1.00**	**0.97**
MUSE $_{\text{supervised; 5 refinements}}$	0.97	1.00	0.93
MUSE $_{\text{supervised; 20 refinements}}$	1.00	1.00	0.87
MUSE $_{\text{unsupervised}}$	1.00	0.97	0.93
VecMap $_{\text{supervised}}$	1.00	0.93	0.83
VecMap $_{\textbf{semi-supervised}}$	**1.00**	**1.00**	**0.93**
VecMap $_{\text{unsupervised}}$	0.97	1.00	0.90

5.1 Coherence

Coherence is an intrinsic evaluation approach to determine the quality of semantic word representations [19]. This approach is based on the idea that high-quality word embedding models should have coherent neighborhoods of semantically similar words. As a result, inserting intruder words that are unrelated to the coherent neighborhoods should be detected easily by a human. For our evaluation, we sample a set $W1_q$ of 30 query words from the vocabulary and get their k-nearest neighbors according to cosine similarity. The query words are representative for the engineering domain and selected by a domain expert. Thereby, the original language of query words is chosen randomly and it is ensured that none of the query words already occurs in the learning dictionary. This prevents mapping algorithms from already knowing these words from training, resulting in biased evaluation results. Following that, we insert an intruder word into each query word's k-nearest neighbors. For evaluation, a domain expert attempts to detect the intruder word among the k-nearest neighbors for each query word, and a coherence score

$$\frac{1}{|W1_q|}\sum_{i=1}^{|W1_q|} d_i \text{ with } d \begin{cases} 1, & \text{if intruder word is detected} \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

for each of our bilingual word embedding models is calculated. As intruder words we randomly select words from both our monolingual trained word embedding vocabularies that have a word count ±1000 of the average word count in the 40,000 most frequent words. For each bilingual word embedding model, we compute the coherence for the neighborhood of k=3, 5, and 10. Table 1 shows the results of this evaluation.

Overall the performance of all models is very similar for this evaluation approach. We can see that all of the trained bilingual word embedding models yield high-quality semantic word representations with coherent neighborhoods. For the best performing models, the evaluator can even correctly detect all intruder words up to the neighborhood of five.

5.2 Comparative Intrinsic Evaluation

Originally, comparative intrinsic evaluation assesses the ability of word embedding models to represent semantic relationships between words [19]. However, we adapt the original approach to evaluate the ability of bilingual word embedding models to map languages in a shared vector space. For this evaluation, we sample a set $W2_q \cap W1_q = \emptyset$ of 20 query words from the vocabulary and get their k-nearest neighbors according to cosine similarity. The query words are chosen by a domain expert to be representative of the engineering domain. As a result, the original language of query words is chosen at random, and none of the query words appear in the learning dictionary. Following that, we determine whether the query words' direct translations are contained in their k-nearest neighbors and compute a comparative intrinsic score

$$\frac{1}{|W2_q|}\sum_{i=1}^{|W2_q|} t_i \text{ with } t \begin{cases} 1, & \text{if translation is included} \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

for each of our bilingual word embedding models. We compute the comparative intrinsic evaluation score for the neighborhood of k=1, 3, 5, and 10 for each bilingual word embedding model. Table 2 shows the results of this evaluation.

We can see that the performance of models highly varies for this evaluation approach. While the naïve bilingual model scored best during the coherence evaluation, it is now the worst performing one. This demonstrates that while the naïve bilingual word embedding approach can generate coherent word embedding neighborhoods, it is incapable of mapping two languages' word embeddings into a common vector space. In contrast, the semi-supervised VecMap model, which already performed well in the coherence evaluation, easily outperforms the other models in the comparative intrinsic evaluation. Supervised MUSE models perform well, but semi-supervised and unsupervised VecMap models outperform them.

5.3 Investigating Bilingual Word Embedding Mappings

Based on our coherence and comparative intrinsic evaluations we observe that the semi-supervised VecMap approach yields the best bilingual word embedding model for engineering. The baseline approach of training a naïve bilingual word embedding model, on the other hand, is completely unsuitable for mapping two engineering-specific word embedding models into a common vector space. This is especially noticeable in the comparative intrinsic evaluation, in which this model performed the worst. Supervised MUSE mappings produce acceptable results, but semi-supervised VecMap mappings outperform them. Furthermore, MUSE is heavily relying on external evaluation data during the refinement steps, while VecMap training does not require this additional bias-prone data. Therefore, we infer that among the investigated approaches, semi-supervised

Table 2: Results of the comparative intrinsic evaluation. The highlighted row indicates the best-performing model, averaged over all neighborhoods.

Mapping	1-NN	3-NN	5-NN	10-NN
Naïve Bilingual Model	0.00	0.00	0.00	0.00
MUSE $_{\text{supervised; 5 refinements}}$	0.40	0.65	0.75	0.90
MUSE $_{\text{supervised; 20 refinements}}$	0.40	0.60	0.70	0.85
MUSE $_{\text{unsupervised}}$	0.00	0.00	0.00	0.00
VecMap $_{\text{supervised}}$	0.00	0.00	0.05	0.05
VecMap $_{\text{semi-supervised}}$	**0.60**	**0.75**	**0.90**	**0.95**
VecMap $_{\text{unsupervised}}$	0.60	0.70	0.85	0.85

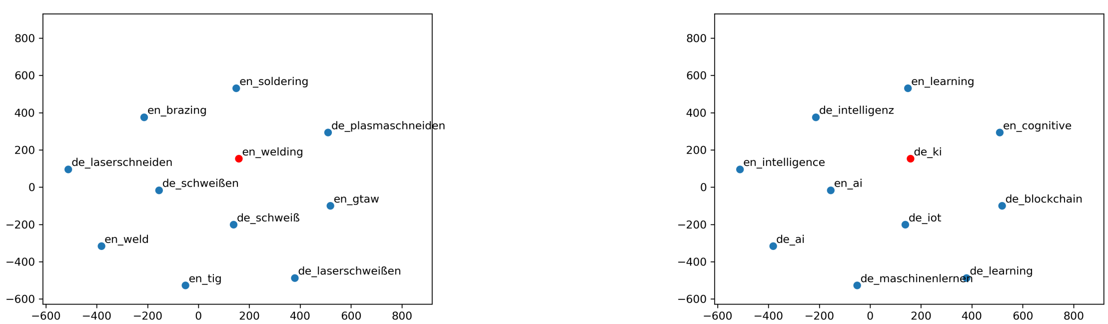

Figure 2: t-SNE projections of the 10 nearest neighbors (blue) of the semi-supervised VecMap generated bilingual engineering-specific word embeddings "en_welding" and "de_ki" (red) in two-dimensional space. Words originating from the German vocabulary are indicated by the prefix "de_" and words originating from the English vocabulary are indicated by the prefix "en_".

VecMap is most suitable to generate bilingual engineering-specific word embedding models. Figure 2 shows exemplary visualizations of bilingual engineering-specific word embeddings generated by semi-supervised VecMap.

Both word embeddings shown in Figure 2 have a balanced ratio of word embeddings of German and English origin in their immediate neighborhood. Furthermore, we can see that both word embeddings contain not only synonyms but also related words in their vicinity. Both aspects correspond to our previously defined use case of assisting engineers in their search for new technologies by recommending related search words to previously defined keywords in two languages.

5.4 Influence of word frequencies on the evaluation results

In addition to the evaluation of bilingual word embedding mappings on unknown query words, we assess the respective model performances on query words that are included in the learning dictionary. The aim of this is to check the plausibility of the previous evaluation results. We anticipate that semi-supervised and supervised mapping approaches perform better on query words already known from the learning dictionary. As a result, we compute additional coherence and comparative intrinsic evaluation scores for 30 and 20 query words in the learning dictionary, respectively. We get the expected results from the supervised VecMap model. It yields coherence and comparative intrinsic scores $\geq$0.60 for each neighborhood with already known query words from the learning dictionary. However, it performs poorly on unknown query words. This is especially reflected in Table 2, where all supervised VecMap comparative intrinsic scores are $\leq$ 0.05. We conclude that this model is overfitted on our learning dictionary and therefore is not able to generalize its bilingual mapping on unknown word embeddings. As a result, we consider the supervised VecMap results as outliers and notice that semi-supervised and supervised mapping approaches, which use the engineering-specific learning dictionary, tend to generate better mappings of bilingual engineering-specific word embeddings than unsupervised approaches that do not use the engineering-specific learning dictionary. In contrast to the supervised VecMap approach, the semi-supervised VecMap approach in particular, behaves completely differently than expected. On average, it yields 44.55% higher coherence and comparative intrinsic scores over all neighborhoods on unknown query words than on known query words from the learning dictionary. To a lesser extent, this trend can also be observed for other bilingual word embedding models. We investigate a possible explanation for this observation because it is counterintuitive. We notice that the unknown query words have much higher word frequencies than the known query words from the learning dictionary. The average word frequency of used unknown query words is 81,891.02, while the average word frequency of used known query words from the learning dictionary is 23,850.99. This means that query words used for evaluation that are not included in the learning dictionary appear on average 3.43 times more often in our document dataset than the query words used for the plausibility check. We conclude from this data that the word frequencies of query words affect the evaluation results. Words with higher frequency of occurrence are more likely to have their direct translation words mapped in their neighborhood. Furthermore, query words that occur more frequently in the document dataset make it easier for human evaluators to detect intruders.

6 CONCLUSION

In this paper, we evaluated different approaches to creating bilingual word embedding models for engineering. We used a document dataset consisting of 3,132,016 engineering related articles in German and English and a learning dictionary from the engineering domain to train bilingual word embedding models. We created word embeddings with Word2Vec and compared naïve bilingual modeling, MUSE, and VecMap approaches to mapping German and English word embeddings. We have adapted the comparative intrinsic evaluation approach, among other things, to assess the ability of bilingual word embedding models to map languages in a shared vector space. Subsequently, we showed that supervised MUSE can yield decent bilingual engineering-specific word embeddings. However, the semi-supervised VecMap approach performed best in mapping engineering-specific word embeddings across German and English. It is very well able to represent semantic relationships between words and to map languages in a shared vector space.

More generally speaking, we have shown that using a domain-specific learning dictionary can improve the mappings of bilingual engineering-specific word embeddings. However, it is critical to avoid overfitting the learning dictionary when training the word embedding mapping. Using a semi-supervised approach may aid preventing this. By recommending related search words to previously defined ones in two languages, we can assist engineers in their search for new technologies.

7 OUTLOOK

In this work, it became apparent that the semi-supervised VecMap approach is very well suited to create bilingual word embeddings for engineering. Future work could adopt this approach to map bilingual engineering-specific document embeddings in a shared vector space. The documents could be embedded with Doc2Vec [24] and mapped using a dictionary of translated document pairs. The document translations could be generated manually or automatically using translation software. Furthermore, future research could look into the ability of state-of-the-art language models to generate bilingual engineering-specific document embeddings. For our use case described in this paper, these context-aware language models are unsuitable as we need to model unique word embeddings for search keywords independently from their position in a context sentence. However, using transformer-based models that are capable of understanding multilingual documents could be investigated in future work to model the relatedness of engineering-specific documents in a bilingual setting.

ACKNOWLEDGMENTS

This work has been supported by funds from the Bavarian Ministry of Economic Affairs, Regional Development and Energy as part of the program “Bayerischen Verbundfoŕderprogramms (BayVFP) – Foŕderlinie Digitalisierung – Foŕderbereich Informations- und Kommunikationstechnik”.

REFERENCES

[1] Y. Liu, Z. Liu, T.-S. Chua and M. Sun, "Topical Word Embeddings," in Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, Austin, Texas, USA, 2015.
[2] T. Schopf, D. Braun and F. Matthes, "Lbl2Vec: An Embedding-based Approach for Unsupervised Document Retrieval on Predefined Topics," in Proceedings of the 17th International Conference on Web Information Systems and Technologies - WEBIST, 2021.
[3] T. Mikolov, K. Chen, G. Corrado and J. Dean, "Efficient Estimation of Word Representations in Vector Space," in Proceedings of Workshop at ICLR, 2013.
[4] J. Pennington, R. Socher and C. Manning, "GloVe: Global Vectors for Word Representation," in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 2014.
[5] P. Bojanowski, E. Grave, A. Joulin and T. Mikolov, "Enriching Word Vectors with Subword Information," in Transactions of the Association for Computational Linguistics, Volume 5, 2017.
[6] M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee and L. Zettlemoyer, "Deep Contextualized Word Representations," in Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), New Orleans, Louisiana, 2018.
[7] V. Efstathiou , C. Chatzilenas and D. Spinellis, "Word embeddings for the software engineering domain," in Proceedings of the 15th International Conference on Mining Software Repositories, 2018.
[8] I. Chalkidis and D. Kampas, "Deep learning in law: early adaptation and legal word embeddings trained on large corpora," in Artificial Intelligence and Law 27, 2019.
[9] Y. Wang, S. Liu, N. Afzal, M. Rastegar-Mojarad, L. Wang, F. Shen, P. Kingsbury and H. Liu, "A comparison of word embeddings for the biomedical natural language processing," in Journal of Biomedical Informatics, 2018.
[10] D. Braun, O. Klymenko, T. Schopf, Y. Kaan Akan and F. Matthes, "The Language of Engineering: Training a Domain-Specific Word Embedding Model for Engineering," in 2021 3rd International Conference on Management Science and Industrial Engineering (MSIE 2021), Osaka, Japan, 2021.
[11] W. Zou, R. Socher, D. Cer and C. Manning, "Bilingual Word Embeddings for Phrase-Based Machine Translation," in Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, Seattle, Washington, USA, 2013.
[12] T. Mikolov, Q. Le and I. Sutskever, "Exploiting Similarities among Languages for Machine Translation," in CoRR, 2013.
[13] M. Artetxe, G. Labaka and E. Agirre, "Learning principled bilingual mappings of word embeddings while preserving monolingual invariance," in Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, Texas, 2016.
[14] M. Artetxe, G. Labaka and E. Agirre, "Learning bilingual word embeddings with (almost) no bilingual data," in Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Vancouver, Canada, 2017.
[15] M. Artetxe, G. Labaka and E. Agirre, "Generalizing and Improving Bilingual Word Embedding Mappings with a Multi-Step Framework of Linear Transformations," in Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18), 2018.
[16] M. Artetxe, G. Labaka and E. Agirre, "A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings," in Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Melbourne, Australia, 2018.
[17] A. Conneau, G. Lample, M. A. Ranzato, L. Denoyer and H. Jégou, "WORD TRANSLATION WITHOUT PARALLEL DATA," in 6th International Conference on Learning Representations, Vancouver, BC, Canada, 2018.
[18] W. Shi, M. Chen, Y. Tian and K.-W. Chang, "Learning Bilingual Word Embeddings Using Lexical Definitions," in Proceedings of the 4th Workshop on Representation Learning for NLP, Florence, Italy, 2019.

[19] T. Schnabel, I. Labutov, D. Mimno and T. Joachims, "Evaluation methods for unsupervised word embeddings," in Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing , Lisbon, Portugal, 2015.
[20] A. Bakarov, "A Survey of Word Embeddings Evaluation Methods," in CoRR, 2018.
[21] B. Li, A. Drozd, Y. Guo, T. Liu, S. Matsuoka and X. Du, "Scaling Word2Vec on Big Corpus," in Data Science and Engineering 4 (2019), 2019.
[22] J. Camacho-Collados, M. T. Pilehvar, N. Collier and R. Navigli, "Multilingual and Cross-lingual Semantic Word Similarity," in Proceedings of the 11th International Workshop on Semantic Evaluations (SemEval-2017), Vancouver, Canada, 2017.
[23] L. van der Maaten and G. Hinton, "Visualizing Data using t-SNE," in Journal of Machine Learning Research 9, 2008.
[24] Q. Le and T. Mikolov, "Distributed Representations of Sentences and Documents," in 31st International Conference on Machine Learning, Beijing, China, 2014.