Multifaceted Domain-Specific Document Embeddings

Julian Risch and **Philipp Hager** and **Ralf Krestel**
Hasso Plattner Institute, University of Potsdam
Potsdam, Germany
firstname.lastname@hpi.de

Abstract

Current document embeddings require large training corpora but fail to learn high-quality representations when confronted with a small number of domain-specific documents and rare terms. Further, they transform each document into a single embedding vector, making it hard to capture different notions of document similarity or explain why two documents are considered similar. In this work, we propose our Faceted Domain Encoder, a novel approach to learn multifaceted embeddings for domain-specific documents. It is based on a Siamese neural network architecture and leverages knowledge graphs to further enhance the embeddings even if only a few training samples are available. The model identifies different types of domain knowledge and encodes them into separate dimensions of the embedding, thereby enabling multiple ways of finding and comparing related documents in the vector space. We evaluate our approach on two benchmark datasets and find that it achieves the same embedding quality as state-of-the-art models while requiring only a tiny fraction of their training data.

1 Introduction

Many documents have an inherently multifaceted nature, a characteristic that domain experts could exploit when searching through large document collections. For example, doctors could search through medical archives for documents containing similar disease descriptions or related uses of a specific drug. However, one of the major challenges of information retrieval in such document collections is domain-specific language use:

1. Training datasets to learn document representations are limited in size,
2. documents might express the same information by using completely different terms (vocabulary mismatch) or different levels of granularity (granularity mismatch),
3. and the lack of context knowledge prevents drawing even simple logical conclusions.

Domain-specific embeddings are available for a variety of domains, including scientific literature (Beltagy et al., 2019), patents (Risch and Krestel, 2019), and the biomedical domain (Kalyan and Sangeetha, 2020). However, these approaches require large amounts of training data and computing resources. In this paper, we introduce and demonstrate our Faceted Domain Encoder, a document embedding approach that produces comparative results on considerably smaller document collections and requires fewer computing resources. Further, it provides a multifaceted view of texts while also addressing the challenges of domain-specific language use. To this end, we introduce external domain knowledge to the embedding process, tackling the problem of vocabulary and granularity mismatches. A screenshot of the demo is shown in Figure 1. The interactive demo, our source code, and the evaluation datasets are available online: https://hpi.de/naumann/s/multifaceted-embeddings and a screencast is available on YouTube: https://youtu.be/HHcsX2clEwg.

2 Related Work

A popular approach for introducing external domain knowledge to the embedding process uses retrofitting of word vectors based on a graph of semantic relationships as a post-processing step (Faruqui et al., 2015). Similarly, Zhang et al. (2019) train fastText embeddings on biomedical journal articles and additionally on sequences of medical terms sampled from a knowledge graph. Dis2Vec uses a lexicon of medical terms to bring Word2Vec vectors of domain terms closer together and to push out-of-domain vectors further away (Ghosh et al., 2016). Unlike Dis2Vec, which concerns only whether a word is in the domain vocabulary or not, our approach handles diverse types

Proceedings of NAACL-HLT 2021: Demonstrations, pages 78–83
June 6–11, 2021.

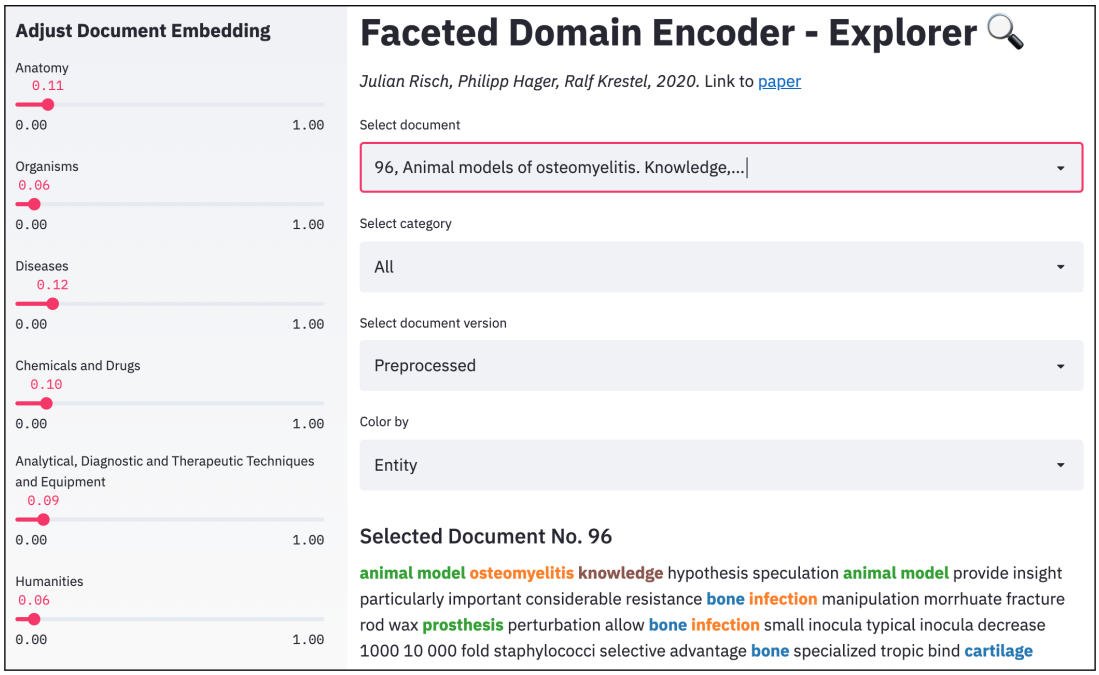

Figure 1: The demo shows nearest neighbor documents and highlights entities within the same categories ("facets"). Stop word removal and lemmatization can be turned off for increased readability. The user interface allows to adjust the weights of the facets of the document embeddings.

of relationships between domain terms. Nguyen et al. (2017) propose an extension of Doc2Vec, adding vectors for domain concepts as input for learning medical document embeddings. Roy et al. (2017) annotate words in the input text with a list of matching entities and relationships from a knowledge graph and extend Word2Vec to jointly learn embeddings for words and annotations. Their abstraction of the graph structure as text annotations enables the inclusion of different node types and edge connections into word embeddings. Another work (Liu et al., 2020) proposed K-BERT, which extends BERT (Devlin et al., 2019) by expanding input sentences with entities from a knowledge graph.

Multifaceted embeddings capture more than one view of a document. Yang et al. (2018) propose a multifaceted network embedding and apply community detection algorithms to learn separate embeddings for each community. Liu et al. (2019) suggest an extension to the deepwalk graph embedding, which learns separate node embeddings for different facets of nodes in a knowledge graph. Similar to our approach, they propose to concatenate the obtained facet embeddings into a single representation. We learn separate embeddings for types of domain knowledge and concatenate them into an overall document representation.

3 Faceted Domain Encoder

Our Faceted Domain Encoder is a supervised learning approach using a Siamese neural network to encode documents and a knowledge graph as a source for additional domain information. The architecture is visualized in Figure 2.

3.1 Overview

The network encodes two documents at-a-time with a bidirectional GRU layer and predicts a similarity score for each pair. By computing the pair's target similarity score based on our knowledge graph, we train the network to adjust its document representations to the relationships between domain terms in the graph. We introduce multiple facets in this process by grouping nodes in the graph into multiple categories. Our model represents different aspects of domain knowledge in different category embeddings by learning not a single embedding vector but an embedding per graph category. We train one embedding for each graph category per document and concatenate them into a single embedding vector to represent the entire document. This representation enables the fast discovery of related documents by performing a conventional nearest neighbor search either based on the whole document or specific category embeddings. To control which category contributes the most to the doc-

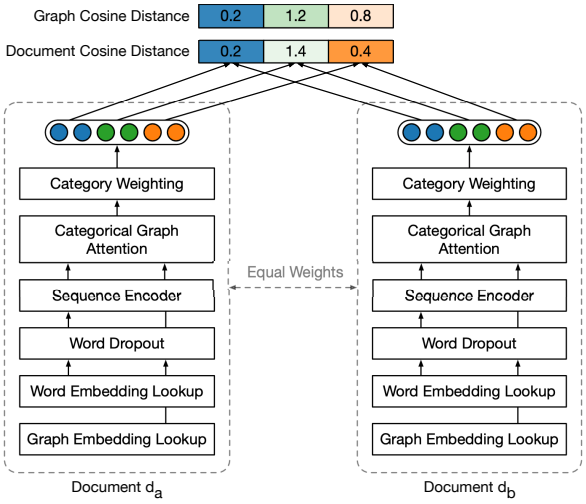

Figure 2: Our model is based on a Siamese network architecture, which encodes two documents in parallel and compares them in the last (top) layer. It is trained to minimize the difference between the documents' cosine distance in the embedding space and their graph-based ground-truth distance. Colors symbolize different facets of the embeddings, which are learned based on node categories in the knowledge graph.

ument vector's overall direction, we apply corpus normalization inspired by Liu et al. (2019).

To cope with limited amounts of training data, our approach leverages external domain knowledge during the training process. We represent this external domain knowledge in the form of a knowledge graph. Each node in the graph represents an entity, e.g., the name of a disease. Each entity belongs to a category, modeled as a node attribute. For example, entities in a medical graph are grouped into diseases, chemicals, or body parts. Categories define the different types of domain knowledge that the model learns to embed into different subparts of the document embedding. Edges between nodes represent relationships, e.g., chemicals in the same group in the periodic table. The entity linking requires a dictionary mapping from words to entities and handles synonyms mapping to the same entity. For the demo, we created a knowledge graph from the taxonomy underlying Medical Subject Headings (MeSH). Figure 3 shows a small excerpt of the graph.

After parsing and deduplicating the official dataset, MeSH comprises 29,641 concepts (entities) and 271,846 synonyms, which are organized in a hierarchy ranging from broad concepts to specific sub-concepts. Following previous work (Guo et al., 2020), we transform the hierarchy into a network graph prevailing the relationships between concepts.

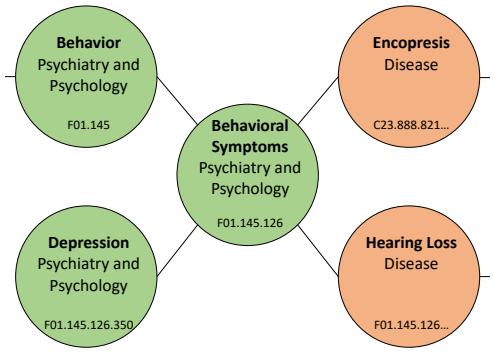

Figure 3: This excerpt of our graph representation of the Medical Subject Headings (MeSH) hierarchy visualizes entities as nodes with their color corresponding to categories ("facets"). The edges and the node numbers reveal the hierarchical relationships, e.g., the broader concept of "Behavior" and the specific mental illness "Depression".

3.2 Equally Weighted Categories

Our approach learns separate embeddings for different categories of domain terms. However, not all categories might be useful when it comes to representing the overall document. We illustrate this problem with a fictional example from the medical domain. Our approach might learn that an article covers a seldom form of cancer (disease category) in the lung and stomach (anatomy category), and the study originates in the United States (location category). Concatenating these three embeddings gives equal weight to each category. The closest document in embedding space needs to be similar in all of the three categories. This might lead to counterintuitive results with the most relating article covering a stomach disease in a small town in Ohio, instead of a document just covering lung cancer. When reading the text again, we might weigh the given information differently based on its specificity and expect the form of cancer to be more important than the geographic location of the study. Note that this problem is magnified when combining up to sixteen categories in the case of our medical dataset. We illustrate the problem with an actual example from our demo in Figure 4.

A second problem can arise when a single, seemingly unimportant category dominates the document embedding. Some documents mention a single term very often, e.g., the word "patient". A high frequency of less-informative words can lead to individual categories collecting vastly more word

embeddings than others and taking over the entire document embedding.

The root cause of both issues is an unintended difference in magnitude between the category embeddings. When concatenating multiple embeddings into a new vector, the category embeddings with the highest magnitude will decide the overall direction of the embedding vector. We address this issue with a simple normalization and weighting process to control which category embeddings contribute the most to the overall direction of the document vector. This approach is similar to what Liu et al. proposed in their work on multifaceted graph embeddings but differs in that we also apply normalization and propose new weighting strategies.

3.3 Category Normalization Strategies

We propose two strategies to compute category weights: corpus-idf and document-tfidf. The first strategy, corpus-idf, sums the inverse-document-frequency of all terms in the category across the entire vocabulary. We normalize the resulting values for all categories to sum to one. This strategy applies the same category weights to all documents in the entire corpus. The motivation is to identify categories that contain the most important words in a collection of documents. This strategy is closely related to the number of unique mentioned tokens in each category.

The second strategy, document-tfidf, computes category weights for individual documents by summing the inverse-document-frequency value of all category terms in the document. Since terms can occur multiple times, the result is similar to the tf-idf value when computed for each category. Additionally, we sum the idf of all words without a category and split the weight equally among all categories. Thereby, we avoid zero weights for categories in the overall embedding. The idea behind this weighting scheme is to have a document-level proxy metric to indicate which categories are important for the document.

4 Experiments

For our experiments, we use two Semantic Textual Similarity (STS) benchmarks from the biomedical domain, BIOSSES (Soğancıoğlu et al., 2017) and Med-STS (Wang et al., 2020). The benchmarks comprise sentence pairs with relatedness scores assigned by domain experts. They measure embedding quality by comparing the annotator score with the embedding similarity of both sentences based on Pearson correlation.

To this end, BIOSSES contains 100 sentence pairs collected from medical articles and judged by five domain experts at a scale of 0 to 4. We perform stratified 10-fold cross-validation as proposed by the benchmark authors. We divide the dataset into ten equally-sized subsets using the annotator scores for stratification. Stratification ensures that each split has a similar distribution of related and unrelated sentence pairs. We train ten separate models on the subsets, always using one subset for testing and the remaining nine for training. Note that we still use 30 percent of the training dataset for validation and early stopping: we stop the training process after the first epoch in which the loss on the validation set stops decreasing. Med-STS contains 1,068 sentence pairs from medical records collected internally in the U.S. Mayo Clinics. Two domain experts judged each sentence pair on a scale from 0 to 5. The dataset authors proposed a train-test split of 750 to 350 sentence pairs. Additionally, we use 30 percent, or 225 pairs, of our training set for validation and early stopping.

The experiment results listed in Table 1 show that our Faceted Domain Encoder outperforms the domain-agnostic embeddings from fastText (Bojanowski et al., 2017) and Universal Sentence Encoder (Cer et al., 2018) on both benchmarks. The corpus-idf normalization is better than the document-tfidf normalization strategy on the BIOSSES dataset but not on the Med-STS dataset. In comparison with the domain-specific embeddings from BioWordVec (Zhang et al., 2019) and BioSentVec (Chen et al., 2019), our approach achieves almost the same performance on Med-STS, which is remarkable given that our Faceted Domain Encoder requires no pre-training on large corpora in contrast to the other presented models. For BIOSSES, only BioSentVec outperforms our approach by a large margin.

5 Interactive User Interface

The user interface comprises three main parts: top center, bottom center, and sidebar. In the top center, the user can select a source document and one or all of the categories ("facets"). Further, either a preprocessed (stop word removal, lemmatization) or a raw document version can be selected for the viewed documents and word highlighting can be

Table 1: Pearson correlation on STS benchmarks (* marks results reported by Chen et al. (2019)).

Embedding	Pre-Trained	BIOSSES	Med-STS
Avg. fastText English (Bojanowski et al., 2017)	✓	0.51	0.68
Universal Sentence Encoder (Cer et al., 2018)	✓	0.35*	0.71*
Avg. BioWordVec (Zhang et al., 2019)	✓	0.69*	0.75*
BioSentVec (Chen et al., 2019)	✓	0.82*	0.77*
Faceted Domain Encoder, Document Normalization		0.53	0.75
Faceted Domain Encoder, Corpus Normalization		0.62	0.72

switched between coloring by entities and coloring by attention scores. The bottom center shows the selected document and the top ten documents that are closest to the selected document in the embedding space. Depending on the selected facet, the documents' distance is calculated based on one specific facet or on the entire document embedding. The sidebar at the left-hand side provides an option to adjust the document embedding in detail. It allows the user to specify what impact the individual facets have on the document's overall embedding.

6 Conclusion

Current document embeddings require large amounts of training data and provide only a single view of document similarity, which prevents searches with different notions of similarity. In this paper, we introduced and demonstrated an approach for multifaceted domain-specific document embeddings. It is tailored to small document collections of only a few hundred training samples and leverages knowledge graphs to enhance the learned embeddings. Experiments on two benchmark datasets show that our model outperforms state-of-the-art domain-agnostic embeddings and is on par with specialized biomedical document embeddings trained on extensive document collections while only using a tiny fraction of their training data. Our demo provides a faceted view into documents by learning to identify different types of domain knowledge and encoding them into specific dimensions of the embeddings. Thereby, it enables novel ways to compare documents and provides a comparatively high level of interpretability of neural-network-based document similarity measures. A promising path for future work is to remove our neural networks' reliance on ground truth data by designing a semi-supervised approach in which the model learns to update its training goal while discovering new domain terms by itself.

References

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A Pretrained Language Model for Scientific Text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3606–3611.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics (TACL)*, 5:135–146.

Daniel Cer, Yinfei Yang, Sheng yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Céspedes, Steve Yuan, Chris Tar, Yun Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Universal Sentence Encoder for English. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 169–174.

Qingyu Chen, Yifan Peng, and Zhiyong Lu. 2019. BioSentVec: Creating Sentence Embeddings for Biomedical Texts. In *Proceedings of the International Conference on Healthcare Informatics (ICHI)*, pages 1–5.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 4171–4186.

Manaal Faruqui, Jesse Dodge, Sujay K. Jauhar, Chris Dyer, Eduard Hovy, and Noah A. Smith. 2015. Retrofitting word vectors to semantic lexicons. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 1606–1615.

Saurav Ghosh, Prithwish Chakraborty, Emily Cohn, John S. Brownstein, and Naren Ramakrishnan. 2016. Characterizing diseases from unstructured text: A vocabulary driven Word2vec approach. In *Proceedings of the International Conference on Information and Knowledge Management (CIKM)*, pages 1129–1138.

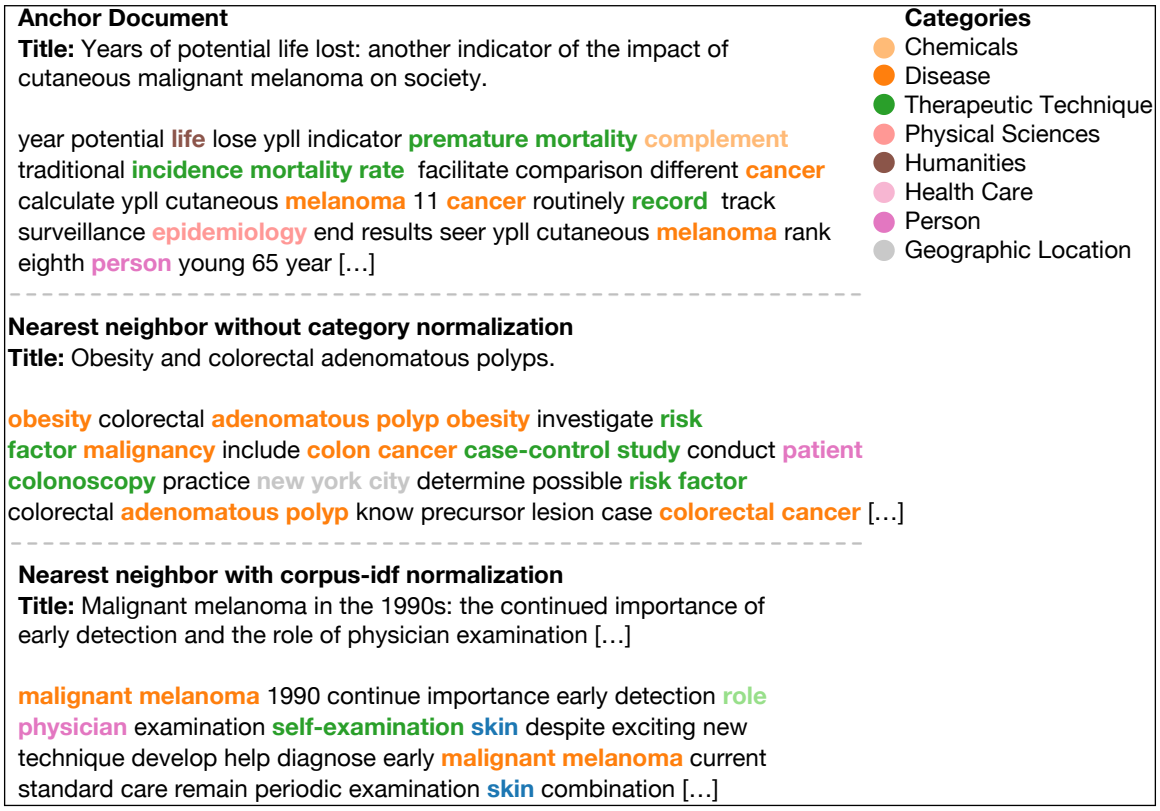

Figure 4: Different weighting of the categories ("facets") changes the distances of the documents in the embedding space and the nearest neighbors of the anchor document. Corpus-idf normalization allows to take into account the frequency of the entities within the corpus. The impact of the most frequent words on the embeddings can thus be reduced. Stop word removal and lemmatization can be turned off for increased readability.

Zhen-Hao Guo, Zhu-Hong You, De-Shuang Huang, Hai-Cheng Yi, Kai Zheng, Zhan-Heng Chen, and Yan-Bin Wang. 2020. MeSHHeading2vec: a new method for representing MeSH headings as vectors based on graph embedding algorithm. *Briefings in Bioinformatics*.

Katikapalli Subramanyam Kalyan and S. Sangeetha. 2020. SECNLP: A survey of embeddings in clinical natural language processing. *Bioinformatics*, 101:1–21.

Ninghao Liu, Qiaoyu Tan, Yuening Li, Hongxia Yang, Jingren Zhou, and Xia Hu. 2019. Is a single vector enough? Exploring node polysemy for network embedding. In *Proceedings of the International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 932–940.

Weijie Liu, Peng Zhou, Zhe Zhao, Zhiruo Wang, Qi Ju, Haotang Deng, and Ping Wang. 2020. K-bert: Enabling language representation with knowledge graph. In *Proceedings of the Conference on Artificial Intelligence (AAAI)*, pages 2901–2908.

Gia Hung Nguyen, Lynda Tamine, Laure Soulier, and Nathalie Souf. 2017. Learning concept-driven document embeddings for medical information search. *Lecture Notes in Computer Science*, 10259(17).

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

Arpita Roy, Youngja Park, and SHimei Pan. 2017. Learning Domain-Specific Word Embeddings from Sparse Cybersecurity Texts. In *arXiv preprint: 1709.07470*.

Gizem Soğancıoğlu, Hakime Öztürk, and Arzucan Özgür. 2017. BIOSSES: A semantic sentence similarity estimation system for the biomedical domain. *Bioinformatics*, 33(14):49–58.

Yanshan Wang, Naveed Afzal, Sunyang Fu, Liwei Wang, Feichen Shen, Majid Rastegar-Mojarad, and Hongfang Liu. 2020. MedSTS: a resource for clinical semantic textual similarity. *Language Resources and Evaluation*, 54(1):57–72.

Liang Yang, Xiaochun Cao, and Guo Yuanfang. 2018. Multi-facet Network Embedding: Beyond the General Solution of Detection and Representation. In *Proceedings of the Conference on Artificial Intelligence (AAAI)*, pages 499–506.

Yijia Zhang, Qingyu Chen, Zhihao Yang, Hongfei Lin, and Zhiyong Lu. 2019. BioWordVec, improving biomedical word embeddings with subword information and MeSH. *Scientific Data*, 6(1):1–9.