Leveraging knowledge graphs to update scientific word embeddings using latent semantic imputation

Jason Hoelscher-Obermaier[*], Edward Stevinson[*]
Valentin Stauber, Ivaylo Zhelev
Victor Botev[†], Ronin Wu[†], Jeremy Minton[†]
Iris AI, Bekkestua, Norway
jason@iris.ai

Abstract

The most interesting words in scientific texts will often be novel or rare. This presents a challenge for scientific word embedding models to determine quality embedding vectors for useful terms that are infrequent or newly emerging. We demonstrate how latent semantic imputation (LSI) can address this problem by imputing embeddings for domain-specific words from up-to-date knowledge graphs while otherwise preserving the original word embedding model. We use the Medical Subject Headings (MeSH) knowledge graph to impute embedding vectors for biomedical terminology without retraining and evaluate the resulting embedding model on a domain-specific word-pair similarity task. We show that LSI can produce reliable embedding vectors for rare and out of vocabulary (OOV) terms in the biomedical domain.

1 Introduction

Word embeddings are powerful representations of the semantic and syntactic properties of words that facilitate high performance in natural language processing (NLP) tasks. Because these models completely rely on a training corpus, they can struggle to reliably represent words which are infrequent, or missing entirely, in that corpus. The latter will happen for any new terminology emerging after training is complete.

Rapid emergence of new terminology and a long tail of highly significant but rare words are characteristic of technical domains, but these terms are often of particular importance to NLP tasks within these domains. This drives a need for methods to generate reliable embeddings of rare and novel words. At the same time, there are efforts in many scientific fields to construct large, highly specific and continuously updated knowledge graphs that capture information about these exact terms. Can we leverage these knowledge graphs to mitigate the short-comings of word embeddings on rare, novel and domain-specific words?

We investigate one method for achieving this information transfer, latent semantic imputation (LSI) (Yao et al., 2019). In LSI the embedding vector for a given word, w, is imputed as a weighted average of existing embedding vectors, where the weights are inferred from the local neighborhood structure of a corresponding embedding vector, $\mathbf{w}_d$, in a domain-specific embedding space. We study how to apply LSI in the context of the biomedical domain using the Medical Subject Headings (MeSH) knowledge graph (Lipscomb, 2000), but expect the methodology to be applicable to other scientific domains.

2 Related work

Embeddings for rare/out of vocabulary (OOV) words. Early methods for embedding rare words relied on explicitly provided morphological information (Alexandrescu and Kirchhoff, 2006; Sak et al., 2010; Lazaridou et al., 2013; Botha and Blunsom, 2014; Luong and Manning, 2016; Qiu et al., 2014). More recent approaches avoid dependence on explicit morphological information by learning representations for fixed-length character n-grams that do not have a direct linguistic interpretation (Bojanowski et al., 2017; Zhao et al., 2018). Alternatively, the subword structure used for generalization beyond a fixed vocabulary can be learnt from data using techniques such as byte-pair encoding (Sennrich et al., 2016; Gage, 1994) or the WordPiece algorithm (Schuster and Nakajima, 2012). Embeddings for arbitrary strings can also be generated using character-level recurrent networks (Ling et al., 2015; Xie et al., 2016; Pinter et al., 2017). These approaches, as well as transformer-based methods mentioned below, provide some OOV generalization capability but are unlikely to be a general solution since they will struggle with

[*] Co-first authors
[†] Co-PIs

Proceedings of the 1st Workshop on Information Extraction from Scientific Publications, pages 43–53
Nov 20, 2022. ©2022 Association for Computational Linguistics

novel terms whose meaning is not implicit in the subword structure such as, *e.g.*, eponyms. Note that we experimented with fastText and it performed worse than our approach.

Word embeddings for the biomedical domain. Much research has focused on how to best generate biomedical-specific embeddings and provide models to improve performance on downstream NLP tasks (Major et al., 2018; Pyysalo et al., 2013; Chiu et al., 2016; Zhang et al., 2019). Work in the biomedical domain has investigated optimal hyperparameters for embedding training (Chiu et al., 2016), the influence of the training corpus (Pakhomov et al., 2016; Wang et al., 2018; Lai et al., 2016), and the advantage of subword-based embeddings (Zhang et al., 2019). Word embeddings for clinical applications have been proposed (Ghosh et al., 2016; Fan et al., 2019) and an overview was provided in Kalyan and Sangeetha (2020). More recently, transformer models have been successfully adapted to the biomedical domain yielding contextual, domain-specific embedding models (Peng et al., 2019; Lee et al., 2019; Beltagy et al., 2019; Phan et al., 2021). Whilst these works highlight the benefits of domain-specific training corpora this class of approaches requires retraining to address the OOV problem.

Improving word embeddings using domain information. Our problem task requires improving a provided embedding model for a given domain, without detrimental effects on other domains.

Zhang et al. (2019) use random walks over the MeSH headings knowledge graph to generate additional training text to be used during the word embedding training. Similar ideas have led to using regularization terms that leverage an existing embedding during training of a new embedding to preserve information from an original embedding during training on a new corpus (Yang et al., 2017). Of course, these methods require the complete training of one or more embedding models.

Faruqui et al. (2014) achieve a similar result more efficiently by defining a convex objective function that balances preserving an existing embedding with decreasing the distance between related vectors, based on external data sources such as a lexicon. This technique has been applied in the biomedical domain (Yu et al., 2016, 2017), but has limited ability to infer new vocabulary because without the contribution from the original embedding this reduces to an average of related vectors.

Another approach is to extend the embedding dimension to create space for encoding new information. This can be as simple as vector concatenation from another embedding (Yang et al., 2017), possibly followed by dimensionality reduction (Shalaby et al., 2018). Alternatively, new dimensions can be derived from existing vectors based on external information like synonym pairs (Jo and Choi, 2018). Again, this has limited ability to infer new vocabulary.

All of these methods change the original embedding, which limits applicability in use-cases where the original embedding quality must be retained or where incremental updates from many domains are required. The optimal alignment of two partially overlapping word embedding spaces has been studied in the literature on multilingual word embeddings (Nakashole and Flauger, 2017; Jawanpuria et al., 2019; Alaux et al., 2019) and provides a mechanism to patch an existing embedding with information from a domain-specific embedding. Unfortunately, it assumes the embedding spaces have the same structure, meaning it is not suitable when the two embeddings encode different types of information, such as semantic information from text and relational information from a knowledge base.

3 Latent Semantic Imputation

LSI, the approach pursued in this paper, represents embedding vectors for new words as weighted averages over existing word embedding vectors with the weights derived from a domain-specific feature matrix (Yao et al., 2019). This process draws insights from Locally Linear Embedding (Roweis and Saul, 2000). Specifically, a local neighborhood in a high-dimensional word embedding space E_s (s for semantic) can be approximated by a lower-dimensional manifold embedded in that space. Hence, an embedding vector $\mathbf{w}_s$ for a word w in that local neighborhood can be approximated as a weighted average over a small number of neighboring vectors.

This would be useful to construct a vector of a new word w if we could determine the weights for the average over neighboring terms. But since, by assumption, we do not know w's word embedding vector $\mathbf{w}_s$, we also do not know its neighborhood in E_s. The main insight of LSI is that we can use the local neighborhood of w's embedding $\mathbf{w}_d$ in a domain-specific space, E_d, as a proxy for that neighborhood in the semantic space of our word-embedding model, E_s. The weights used for constructing an embedding for w in E_s are calculated from the domain space as shown in Fig. 1: a k-nearest-neighbors minimum-spanning-tree (kNN-MST) is built from the domain space features. Then the L2-distance between $\mathbf{w}_d$ and a weighted average over its neighbors in the kNN-MST is minimized using non-negative least squares. The resulting weights are used to impute the missing embedding vectors in E_s using the power iteration method. This procedure crucially relies on the existence of words with good representations in both E_s and E_d, referred to as anchor terms, which serve as data from which the positions of the derived embedding vectors are constructed.

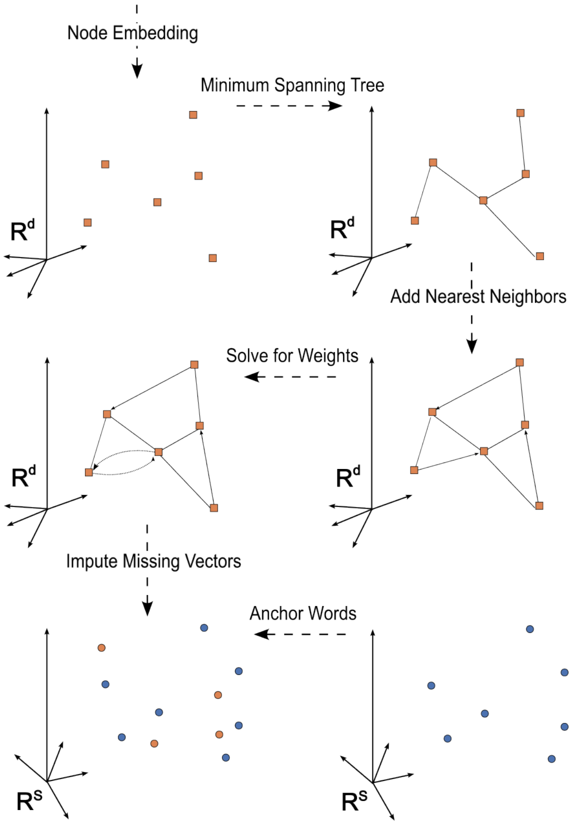


Figure 1: Latent Semantic Imputation. $\mathbb{R}^d$ is the domain space and $\mathbb{R}^s$ is the semantic space.

4 Methodology

We extend the original LSI procedure described above in a few key ways. Instead of using a numeric data matrix as the domain data source of LSI, we use a node embedding model trained on a domain-specific knowledge graph to obtain E_d. As knowledge graphs are used as a source of structured information in many fields, we expect our method to be applicable to many scientific domains. Knowledge graphs are prevalent in scientific fields as they serve as a means to organise and store scientific data, as well as to aid downstream tasks such as reasoning and exploration. Their structure and ability to represent different relationship types makes it relatively easy to integrate new data, meaning they can evolve to reflect changes in a field and as new data becomes available.

We use the 2021 RDF dump of the MeSH knowledge graph (available at https://id.nlm.nih.gov/mesh/). The complete graph consists of 2,327,188 nodes and 4,272,681 edges, which we reduce into a simpler, smaller, and undirected graph to be fed into a node embedding algorithm. We extract a subgraph consisting of solely the nodes of type "ns0__TopicalDescriptor" and the nodes of type "ns0__Concept" that are directly connected to the topical descriptors via any relationship type. The relationship types and directionality were removed. This results in 58,695 nodes and 113,094 edges.

We use the node2vec graph embedding algorithm (Grover and Leskovec, 2016) on this subgraph to produce an embedding matrix of 58,695 vectors with dimension 200 (orange squares in Fig. 2). The hyperparameters are given in Appendix 8.1. These node embeddings form the domain-specific space, E_d, as described in the previous section. We note that in preliminary experiments, the adjacency matrix of the knowledge graph was used directly as E_d but this yielded imputed embeddings that performed poorly.

To provide the mapping between the MeSH nodes and the word embedding vocabulary we normalize the human-readable "rdfs__label" node property by replacing spaces with hyphens and lower-casing. The anchor terms are then identified as the normalized words that match between the graph labels and the vocabulary of the word-embedding model; resulting in 12,676 anchor terms. As an example, "alpha-2-hs-glycoprotein" appears as both a node in the reduced graph and in the word-embedding model, along with its neighbors in the kNN-MST, which include "neoglycoproteins" and "alpha-2-antiplasmin". These serve to stabilise the positions of unknown word embedding vectors for domain space nodes which did not have corresponding representations in the semantic

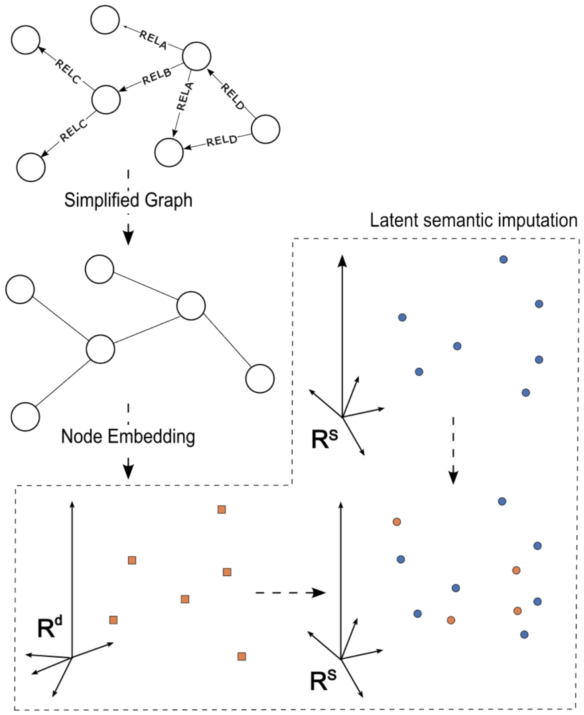


Figure 2: Extended latent semantic imputation pipeline. A knowledge graph is simplified to a smaller, undirected graph. This is used to derive the node embedding model used in LSI (see Fig. 1) to impute missing terms in the semantic space.

space during the LSI procedure.

LSI has one key hyper-parameter: the minimal degree of the kNN-MST graph, k. The stopping criterion of the power iteration method is controlled by another parameter, η, but any sufficiently small value should allow adequate convergence and have minimal impact on the resulting vectors. Following Yao et al. (2019) we set $\eta = 10^{-4}$ but we use a larger $k = 50$ since initial experiments showed a better performance for larger values of k.

5 Experiments

We aim to answer two questions to evaluate our imputation approach: Do the imputed embeddings encode semantic similarity and relatedness information as judged by domain experts? And, can the imputed embeddings be reliably used alongside the original, non-imputed word embeddings?

We use the UMNSRS dataset to answer these questions (Pakhomov et al., 2010). It is a collection of medical word-pairs annotated with a relatedness and similarity score by healthcare professionals, such as medical coders and clinicians; some examples are shown in Table 1. For each word-pair we calculate the cosine similarity between the corresponding word embedding vectors and report the Pearson correlation between these cosine similarities and the human scores.

Term1	Term2	Similarity	Relatedness
Acetylcysteine	Adenosine	256.25	586.50
Anemia	Coumadin	623.75	926.50
Rales	Lasix	742.00	1379.50
Tuberculosis	Hemoptysis	789.50	1338.50

Table 1: Examples of UMNSRS word pairs. Scores range from 0 to 1600 (larger = more similar/related).

To obtain additional insight into the performance of the imputation procedure we split the words in the UMNSRS dataset into two groups of roughly the same size: one group of words (*trained*) which we train directly as part of the word embedding training and another group of words (*imputed*) which we obtain via imputation. This split results in three word-pair subsets that contain *imputed/imputed* word pairs, *trained/trained* word pairs, and *imputed/trained* word pairs. Note that due to an incomplete overlap of the UMNSRS test vocabulary with both the MeSH node labels and our word embedding vocabulary we cannot evaluate on every word pair in UMNSRS (see Table 4 for more details). Applying the UMNSRS evaluation to these three groups of word pairs we aim to measure the extent to which the imputation procedure encodes domain-specific semantic information.

For word embedding training we prepare a corpus of 74.4M sentences from open access publications on PubMed (from `https://ftp.ncbi.nlm.nih.gov/pub/pmc/oa_bulk/`; accessed on 2021-08-30). To simulate the problem of missing words as realistically as possible we then prepare a filtered version of this corpus by removing any sentence containing one of the *imputed* terms (in either singular or plural form). This filtering removes 2.36M of the 74.4M sentences (3.2%).

We then train 200-dimensional skip-gram word embedding models on both the full and the filtered version of the training corpus. In addition, we also train fastText embeddings (Bojanowski et al., 2017) on both the full and the filtered corpus. For details on the hyper-parameters see Appendix 8.2. Since fastText, which represents words as n-grams of their constituent characters, has been shown to give reasonable embedding vectors for words which are rare or missing in the training corpus it represents a suitable baseline to which we can compare our imputation procedure.

We check that the embedding models (both skip-gram and fastText) trained on the filtered corpus perform roughly on par with those trained on the full corpus when evaluated using the *trained/trained* subset of the UMNSRS test data. We also check that the skip-gram model trained on the full corpus performs comparable to the BioWordVec model (Zhang et al., 2019) across all subsets of UMNSRS. See Appendix 8.3 for details.

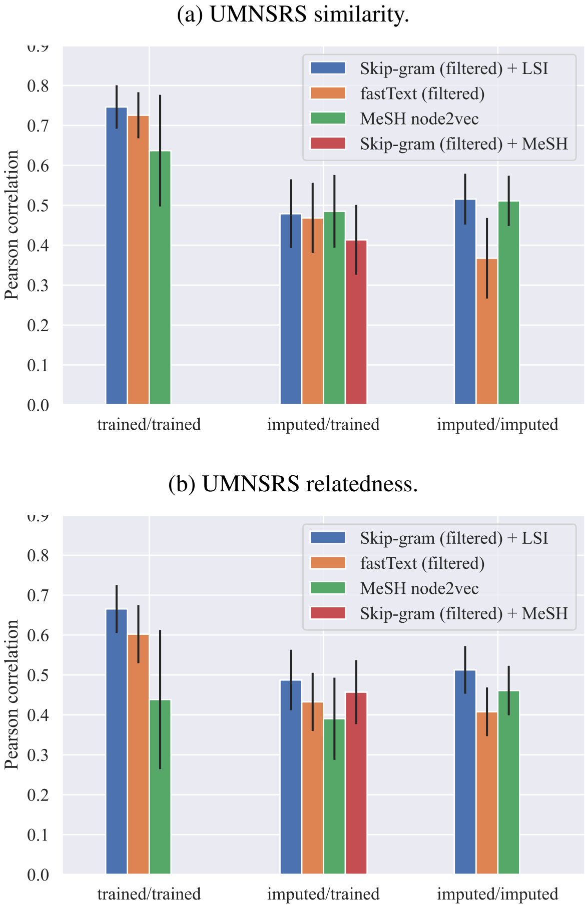
LSI is a means of leveraging the domain space to create OOV embedding vectors. As a simple alternative baseline, we directly use the domain space embeddings for the OOV words. We need to align the domain space onto the semantic space, which we do with a rotation matrix derived from the anchor term embeddings in the two spaces via singular value decomposition.

5.1 Results

The main results are displayed in Fig. 3 which shows the Pearson correlation between cosine similarities and human annotator scores for UMNSRS similarity and relatedness. The error bars are standard deviations across 1,000 bootstrap resamples of the test dataset. From left to right we show results for the *trained/trained*, *imputed/trained*, and *imputed/imputed* subsets.

We compare two models trained on the filtered corpus (which does not contain any mentions of the *imputed* words): a skip-gram model extended by LSI and a fastText model. For reference we also show the correlation strengths obtained when directly using the MeSH node embeddings which form the basis of the imputation. Note that for this last model, the test cases we evaluate are different, since the MeSH model cannot represent all word pairs in UMNSRS (see appendix 8.3 for details). Uncertainties on the MeSH model are high for the *trained/trained* subset due to the limited overlap of the MeSH model with the words in the *trained* subset (see Table 4).

In Fig. 3 the *imputed/trained* group also includes the performance of the simple baseline, *Skip-gram (filtered) + MeSH*, formed of a mixture of aligned embeddings. We do not show the performance of this baseline on the other two groups since, by construction, it is identical to that of *Skip-gram (filtered) + LSI* for *trained/trained* and that of *MeSH node2vec* for *imputed/imputed*.


(a) UMNSRS similarity.

(b) UMNSRS relatedness.

Figure 3: Correlation with UMNSRS scores.

Three things stand out:

1. The LSI-based model is competitive on novel vocabulary: it performs significantly better than the fastText model on word pairs containing only imputed terms (*imputed/imputed*) and modestly better on mixed word pairs (*imputed/trained*). It also outperforms the simple but surprisingly strong baseline, *Skip-gram (filtered) + MeSH*.

2. There is a significant difference in Pearson correlation between the different word pair categories. Note that the same trend in correlation across word pair categories can be seen in the word embedding model trained on the full corpus without imputation (see Fig. 4).

3. The LSI-based model obtains better scores than the underlying MeSH node embeddings across most categories. This proves that the similarity and relatedness information directly encoded in the domain embedding does not limit the similarity and relatedness information encoded in the resulting imputed model.

5.2 Discussion

In this paper we use a significantly larger subset of the MeSH graph compared to related work on MeSH-based embeddings (Guo et al., 2021; Zhang et al., 2019) by including more than just the topical descriptor nodes. Using a larger graph for the imputation allows us to impute a wider variety of words and evaluate the imputation procedure on a larger subset of UMNSRS. The graph we use for imputation is also much larger than the domain data used in previous work on LSI (Yao et al., 2019). This shows that LSI can apply to knowledge graphs and scale to larger domain spaces which is crucial for real-world applications.

We observe that the UMNSRS similarity and relatedness correlations of the MeSH node embedding models do not constitute an upper bound on the correlations obtained for the imputed word embeddings. This is intuitively plausible since LSI combines the global structure of the trained word embedding vectors with the local structure of the domain embeddings. This is in contrast to the original LSI paper in which the domain data alone was sufficient to obtain near perfect scores on the evaluation task and, as such, could have been used directly which obviates the need for LSI. This observation reduces the pressure for an optimal knowledge graph and associated embedding, although a systematic search for better subgraphs to use is likely to yield improved imputation results.

It is also of note that most of the trends displayed by the LSI model hold for both the similarity and relatedness scores, despite these being distinctly separate concepts. Relatedness is a more general measure of association between two terms whilst similarity is a narrower concept tied to their likeness. This might not be the case if the graph construction had been limited to particular relationship types or if direction of the relations had been retained.

There are noteworthy differences between our experiment and the use cases we envisage for LSI. The words we impute in our experiment are taken from the constituent words of the UMNSRS word pairs rather than being solely defined by training corpus statistics. This is a necessary limitation of our evaluation methodology. It remains a question for further research to establish ways of evaluating embedding quality on a larger variety of OOV words and use this for a broader analysis of the performance of LSI.

6 Strengths and weaknesses of LSI

Our experiments highlight several beneficial features of LSI. It is largely independent of the nature of the domain data as long as embeddings for the domain entities can be inferred. It does not rely on retraining the word embedding and is therefore applicable to cases where retraining is not an option due to limitations in compute or because of lack of access to the training corpus. It allows word embeddings to be improved on demand for specific OOV terms, thus affording a high level of control. In particular, it allows controlled updates of word embeddings in light of new emerging research.

The current challenges we see for LSI are driven by limited research in the constituent steps of the imputation pipeline. Specifically, there is not yet a principled answer for the optimal selection of a subgraph from the full knowledge graph or the optimal choice of node embedding architecture. The answer to these may depend on the domain knowledge graph. Also, there are not yet generic solutions for quality control of LSI. This problem is likely intrinsically hard since the words which are most interesting for imputation are novel or rare and thus exactly the words for which little data is available.

7 Conclusion

In this paper, we show how LSI can be used to improve word embedding models for the biomedical domain using domain-specific knowledge graphs. We use an intrinsic evaluation task to demonstrate that LSI can yield good embeddings for domain-specific out of vocabulary words.

We significantly extend the work of Yao et al. (2019) by showing that LSI is applicable to scientific text where problems with rare and novel words are particularly acute. Yao et al. (2019) assumed a small number of domain entities and a numeric domain data feature matrix. This immediately yields the metric structure required to determine the nearest neighbors and minimum spanning tree graph used in LSI. We extend this to a much larger number of domain entities and to domain data which does not have an a priori metric structure but is instead given by a graph structure. We demonstrate that LSI can also work with relational domain data thus opening up a broader range of data sources. The metric structure induced by node embeddings trained on a domain knowledge graph provides an equally good starting point for LSI.

This shows that LSI is a suitable methodology for controlled updates and improvements of scientific word embedding models based on domain-specific knowledge graphs.

8 Future work

We see several fruitful directions for further research on LSI and would like to see LSI applied to other scientific domains, thereby testing the generalizability of our methodology. This would also provide more insight on how the domain knowledge graph as well as the node embedding architecture impact the imputation results.

The use of automatic methods for creating medical term similarity datasets (Schulz and Juric, 2020) would facilitate the creation of large-scale test sets. The UMNSRS dataset, along with the other human-annotated, biomedical word pair similarity test sets used in the literature, all consist of fewer than one thousand word pairs (Pakhomov et al., 2016, 2010; Chiu et al., 2018). The use of larger test sets would remove the aforementioned evaluation limitations.

Further research could elucidate how to best utilize the full information of the domain knowledge graph in LSI. This includes information about node and edge types, as well as literal information such as human-readable node labels and numeric node properties (such as measurement values). It also remains to be studied how to optimally choose the anchor terms (to be used in the imputation step) to maximize LSI performance. Our methodology could also be generalized from latent semantic imputation to what might be called latent semantic information fusion where domain information is used for incremental updates instead of outright replacement of word embedding vectors.

Finally, LSI could also be extended to provide alignment between knowledge graphs and written text by using the spatial distance between imputed vectors of knowledge graph nodes and trained word embedding vectors as an alignment criterion.

Acknowledgements

This paper was supported by the AI Chemist funding (Project ID: 309594) from the Research Council of Norway (RCN). We thank Shibo Yao for helpful input and for sharing raw data used in (Yao et al., 2019) and Dr. Zhiyong Lu and Dr. Yijia Zhang of the National Institute of Health for sharing their word embedding models. We thank the three anonymous reviewers for their careful reading and helpful comments.

References

Jean Alaux, Edouard Grave, Marco Cuturi, and Armand Joulin. 2019. Unsupervised Hyperalignment for Multilingual Word Embeddings.

Andrei Alexandrescu and Katrin Kirchhoff. 2006. Factored Neural Language Models. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 1–4, New York City, USA. Association for Computational Linguistics.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A Pretrained Language Model for Scientific Text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

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.

Jan A. Botha and Phil Blunsom. 2014. Compositional Morphology for Word Representations and Language Modelling.

Billy Chiu, Gamal 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*, pages 166–174, Berlin, Germany. Association for Computational Linguistics.

Billy Chiu, Sampo Pyysalo, Ivan Vulić, and Anna Korhonen. 2018. Bio-simverb and bio-simlex: wide-coverage evaluation sets of word similarity in biomedicine. *BMC Bioinformatics*, 19(1):33.

Yadan Fan, Serguei Pakhomov, Reed McEwan, Wendi Zhao, Elizabeth Lindemann, and Rui Zhang. 2019. Using word embeddings to expand terminology of dietary supplements on clinical notes. *JAMIA Open*, 2(2):246–253.

Manaal Faruqui, Jesse Dodge, Sujay K. Jauhar, Chris Dyer, Eduard Hovy, and Noah A. Smith. 2014. Retrofitting word vectors to semantic lexicons. *arXiv preprint arXiv:1411.4166*.

Philip Gage. 1994. A new algorithm for data compression. *The C Users Journal archive*, 12:23–38.

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 25th ACM International on Conference on Information and Knowledge Management*, CIKM '16, page 1129–1138, New York, NY, USA. Association for Computing Machinery.

Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks.

Zhen-Hao Guo, Zhu-Hong You, De-Shuang Huang, Hai-Cheng Yi, Kai Zheng, Zhan-Heng Chen, and Yan-Bin Wang. 2021. MeSHHeading2vec: A new method for representing MeSH headings as vectors based on graph embedding algorithm. *Briefings in Bioinformatics*, 22(2):2085–2095.

Pratik Jawanpuria, Arjun Balgovind, Anoop Kunchukuttan, and Bamdev Mishra. 2019. Learning Multilingual Word Embeddings in Latent Metric Space: A Geometric Approach. *Transactions of the Association for Computational Linguistics*, 7:107–120.

Hwiyeol Jo and Stanley Jungkyu Choi. 2018. Extrofitting: Enriching Word Representation and its Vector Space with Semantic Lexicons. *arXiv:1804.07946 [cs]*.

Katikapalli Subramanyam Kalyan and S. Sangeetha. 2020. SECNLP: A survey of embeddings in clinical natural language processing. *Journal of Biomedical Informatics*, 101:103323.

Siwei Lai, Kang Liu, Shizhu He, and Jun Zhao. 2016. How to generate a good word embedding. *IEEE Intelligent Systems*, 31(6):5–14.

Angeliki Lazaridou, Marco Marelli, Roberto Zamparelli, and Marco Baroni. 2013. Compositional-ly Derived Representations of Morphologically Complex Words in Distributional Semantics. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1517–1526, Sofia, Bulgaria. Association for Computational Linguistics.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. BioBERT: A pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, page btz682.

Wang Ling, Chris Dyer, Alan W Black, Isabel Trancoso, Ramón Fermandez, Silvio Amir, Luís Marujo, and Tiago Luís. 2015. Finding Function in Form: Compositional Character Models for Open Vocabulary Word Representation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1520–1530, Lisbon, Portugal. Association for Computational Linguistics.

Carolyn E. Lipscomb. 2000. Medical Subject Headings (MeSH). *Bulletin of the Medical Library Association*, 88(3):265–266.

Minh-Thang Luong and Christopher D. Manning. 2016. Achieving Open Vocabulary Neural Machine Translation with Hybrid Word-Character Models. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1054–1063, Berlin, Germany. Association for Computational Linguistics.

Vincent Major, Alisa Surkis, and Yindalon Aphinyanaphongs. 2018. Utility of General and Specific Word Embeddings for Classifying Translational Stages of Research. *AMIA Annual Symposium Proceedings*, 2018:1405–1414.

Ndapandula Nakashole and Raphael Flauger. 2017. Knowledge Distillation for Bilingual Dictionary Induction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2497–2506, Copenhagen, Denmark. Association for Computational Linguistics.

Serguei V. S. Pakhomov, Greg Finley, Reed McEwan, Yan Wang, and Genevieve B. Melton. 2016. Corpus domain effects on distributional semantic modeling of medical terms. *Bioinformatics (Oxford, England)*, 32(23):3635–3644.

Serguei V. S. Pakhomov, Bridget T. McInnes, T. Adam, Y. Liu, Ted Pedersen, and G. Melton. 2010. Semantic Similarity and Relatedness between Clinical Terms: An Experimental Study. In *AMIA ... Annual Symposium Proceedings. AMIA Symposium.*

Yifan Peng, Shankai Yan, and Zhiyong Lu. 2019. Transfer Learning in Biomedical Natural Language Processing: An Evaluation of BERT and ELMo on Ten Benchmarking Datasets.

Long N. Phan, James T. Anibal, Hieu Tran, Shaurya Chanana, Erol Bahadroglu, Alec Peltekian, and Grégoire Altan-Bonnet. 2021. SciFive: A text-to-text transformer model for biomedical literature.

Yuval Pinter, Robert Guthrie, and Jacob Eisenstein. 2017. Mimicking Word Embeddings using Subword RNNs. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 102–112, Copenhagen, Denmark. Association for Computational Linguistics.

Sampo Pyysalo, Filip Ginter, Hans Moen, Tapio Salakoski, and Sophia Ananiadou. 2013. Distributional Semantics Resources for Biomedical Text Processing. In *Proceedings of LBM 2013*, page 5.

Siyu Qiu, Qing Cui, Jiang Bian, Bin Gao, and Tie-Yan Liu. 2014. Co-learning of Word Representations and Morpheme Representations. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 141–150, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Sam T. Roweis and Lawrence K. Saul. 2000. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326.

Haşim Sak, Murat Saraçlar, and Tunga Güngör. 2010. Morphology-based and sub-word language modeling for Turkish speech recognition. In *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5402–5405.

Claudia Schulz and Damir Juric. 2020. Can embeddings adequately represent medical terminology? new large-scale medical term similarity datasets have the answer! *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8775–8782.

Mike Schuster and Kaisuke Nakajima. 2012. Japanese and korean voice search. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5149–5152.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural Machine Translation of Rare Words with Subword Units. *arXiv:1508.07909 [cs]*.

W. Shalaby, Wlodek Zadrozny, and Hongxia Jin. 2018. Beyond word embeddings: Learning entity and concept representations from large scale knowledge bases. *Information Retrieval Journal*.

Yanshan Wang, Sijia Liu, Naveed Afzal, Majid Rastegar-Mojarad, Liwei Wang, Feichen Shen, Paul Kingsbury, and Hongfang Liu. 2018. A comparison of word embeddings for the biomedical natural language processing. *Journal of Biomedical Informatics*, 87:12–20.

Ruobing Xie, Zhiyuan Liu, Jia Jia, Huanbo Luan, and Maosong Sun. 2016. Representation Learning of Knowledge Graphs with Entity Descriptions.

Wei Yang, Wei Lu, and Vincent Zheng. 2017. A Simple Regularization-based Algorithm for Learning Cross-Domain Word Embeddings. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2898–2904, Copenhagen, Denmark. Association for Computational Linguistics.

Shibo Yao, Dantong Yu, and Keli Xiao. 2019. Enhancing Domain Word Embedding via Latent Semantic Imputation. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 557–565.

Zhiguo Yu, Trevor Cohn, Byron C. Wallace, Elmer Bernstam, and Todd Johnson. 2016. Retrofitting word vectors of mesh terms to improve semantic similarity measures. In *Proceedings of the Seventh International Workshop on Health Text Mining and Information Analysis*, pages 43–51.

Zhiguo Yu, Byron C. Wallace, Todd Johnson, and Trevor Cohen. 2017. Retrofitting concept vector representations of medical concepts to improve estimates of semantic similarity and relatedness. *Studies in health technology and informatics*, 245:657.

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):52.

Jinman Zhao, Sidharth Mudgal, and Yingyu Liang. 2018. Generalizing Word Embeddings using Bag of Subwords.

Appendix

8.1 Hyper-parameters for MeSH node2vec

We train node2vec (`https://github.com/thibaudmartinez/node2vec`) embeddings with the hyperparameters shown in Table 2 from a subgraph of MeSH containing 58,695 nodes and 113,094 edges.

Hyperparameter	Variable name	Value
Training epochs	epochs	50
No. of random walks	n_walks	10
Return parameter	p	0.5
Inout parameter	q	0.5
Context window	context_size	15
Dimension	dimension	200

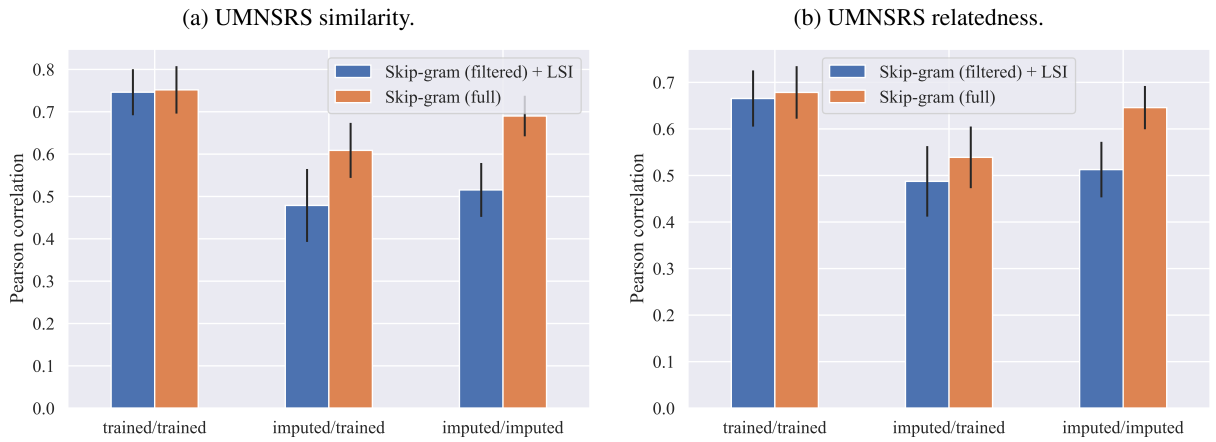
Table 2: Hyperparameters for MeSH node2vec training

8.2 Hyper-parameters for word embeddings

We use gensim (`https://radimrehurek.com/gensim`; version 4.1.2.) for training skipgram and fastText word embedding models with the hyperparameters provided in Table 3. All other hyperparameters are set to the default values of the gensim implementation. For the skipgram model we use the hyperparameters from Chiu et al. (2016), which are reported to be optimal for the biomedical domain. For fastText we are not aware of literature on optimal hyperparameters for the biomedical domain so we use the default values except for the embedding dimension which we set to 200 to ease comparison with the skipgram model. We trained the fastText models for 10 epochs but found that the performance of the fastText model on UMNSRS saturates after epoch 1. We use the fastText model after the first epoch for the remainder of our experiments and analysis.

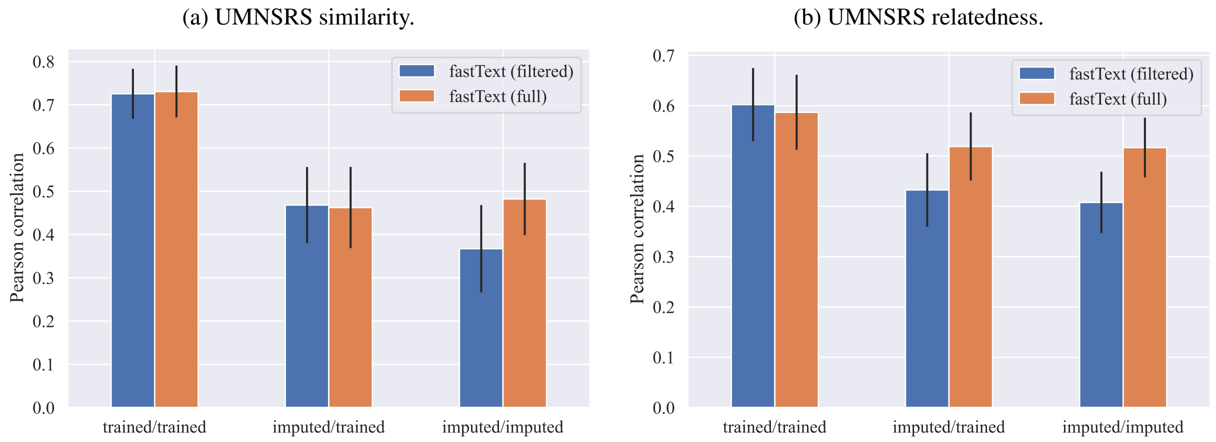
Variable name	fastText	skipgram
epochs	1	10
negative	5	10
vector_size	200	200
alpha	0.025	0.05
sample	1E-03	1E-04
window	20	30

Table 3: Hyperparameters for skipgram and fastText training. See the gensim documentation for the definition of the hyperparameters.


(a) UMNSRS similarity.

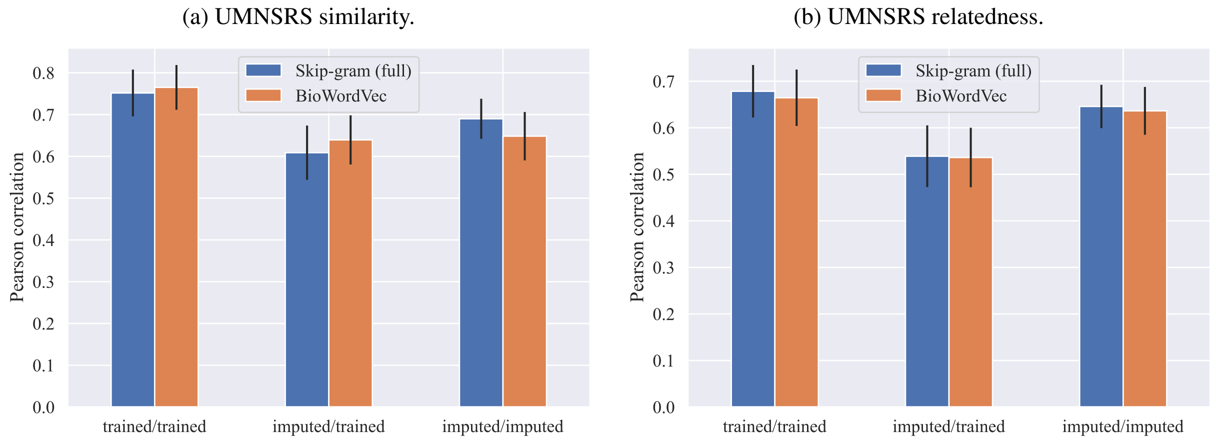
(b) UMNSRS relatedness.

Figure 4: UMNSRS correlations for skipgram models.


(a) UMNSRS similarity.

(b) UMNSRS relatedness.

Figure 5: UMNSRS correlations for fastText models.


(a) UMNSRS similarity.

(b) UMNSRS relatedness.

Figure 6: UMNSRS correlations for BioWordVec.

Model	UMNSRS relatedness			UMNSRS similarity		
	trained/ trained	imputed/ trained	imputed/ imputed	trained/ trained	imputed/ trained	imputed/ imputed
MeSH node2vec	28	70	133	30	72	135
all other models	83	99	124	84	101	126

Table 4: Number of test cases per model and test set split for UMNSRS evaluation.

8.3 Details on the UMNSRS evaluation

Table 4 shows the number of test cases per model and UMNSRS test data split. All models have been evaluated on the same subsets of UMNSRS except for the MeSH node embeddings model where limited overlap with the UMNSRS test vocabulary prevents us from evaluating on exactly the same subsets.

The embedding models (both skip-gram and fastText) trained on the filtered corpus perform roughly on par with those trained on the full corpus when evaluated using the *trained/trained* subset of the UMNSRS test data (see Fig. 4 and 5). When comparing the performance of the filtered skipgram model + LSI to the full skipgram model on the subset of test data involving imputed words (*imputed/trained* and *imputed/imputed*) the full model outperforms LSI (see Fig. 4). This suggests that, if training text for the OOV words were available, we should make use of it. Similarly, and as expected, when comparing the performance of the filtered and full fastText models on the subset of test data involving imputed words (*imputed/trained* and *imputed/imputed*) the full model again outperforms the filtered model (see Fig. 5).

As a sanity check, we also compare the skip-gram model trained on the full corpus to BioWordVec, a recent state-of-the-art word embedding model for the biomedical domain (Zhang et al., 2019) and find similar performance across all subsets of UMNSRS (see Fig. 6).