Can Language Inference Support Metadata Generation?

José María González Pinto[1[0000-0002-2908-3466]], Janus Wawrzinek[1[0000-0002-8733-2037]], Suma Kori[2] and Wolf-Tilo Balke[1[0000-0002-5443-1215]]

[1] IFIS TU-Braunschweig, Mühlenpfordstrasse 23, 38106 Braunschweig, Germany
{wawrzinek, pinto, balke}@ifis.cs.tu-bs.de}
[2]s.kori@tu-braunschweig.de

Abstract. As more papers get included in Digital collections satisfying information needs is becoming harder. In particular, when the user searches for information beyond bibliographic metadata. The situation is even worse when the information need requires *a key aspect* of a paper that first needs to be annotated for indexing purposes and thus, allow searching. For instance, in the biomedical field this might apply to structured abstracts, e.g. 'background', 'objectives', 'results', 'methods' and 'conclusion'. Current state-of-the-art deep learning approaches can only succeed if a sufficiently large amount of annotated data is available for training purposes. However, annotating several thousands of documents is not only expensive, but due to the limited availability of experts often even infeasible. To alleviate this problem, we explore the use of *Language Inference* as a *universal feature* that once applied to a limited number of annotated documents can help to achieve high accuracy to generate the desired metadata. We show through our experiments the degree of success on the difficult task of generating the *structured metadata* of biomedical papers and its performance stability as we increase the number of examples. We compare our suggested approach with deep learning approaches such as Doc2Vec and show that language inference is up to two orders of magnitude better achieving up to 0.82 F1 scores.

Keywords: digital libraries, metadata generation, language inference.

1 Introduction

To help users satisfy complex information needs and explore the richness contained in the knowledge of Digital collections, information providers rely on indexing mechanisms using high-quality metadata. Furthermore, as more critical aspects of scientific manuscripts are discovered and used by the community, for instance, scientific claims [7, 8] to ensure high-quality content in Digital collections or structured abstracts in the biomedical field [9, 10, 12], the need to annotate documents with such first-class citizens increases. However, annotating documents is an expensive and time-consuming process. Current state-of-the-art deep learning approaches [13, 26] have succeeded in related tasks such as document classification but with one major caveat common to most of them: they require several thousands of annotated examples to learn useful features to deliver high-quality results. It is excellent that manual feature engineering

with these deep learning models is not needed anymore. However, as resources are scarce annotating several thousands of documents is just unfeasible. However, current advances in the natural language community may help to solve this need in our community. In particular, language inference [2]. Language inference aims at instantiating computational models that help machines to learn useful general representations of language that can be used in different tasks. For instance, the case of InferSent [4] trained to distinguish whether a given pair of sentences contradict, are related, or one entails the other, have shown promising results. The question is whether this *universal representation of sentences* that were not explicitly trained on the biomedical domain can indeed be used as *features to generate high-quality biomedical metadata*?

Although some studies on using Language Inference to classification tasks exist see for instance [4], we still know little about its applicability in Digital libraries and in particular in scientific collections where the language complexity is higher with many acronyms and terminology variations. Furthermore, can language inference help when the number of examples is limited? In this paper, we shed new light into the suitability of language inference for metadata generation. In particular, we show in the biomedical field how InferSent can be used to generate high-quality structured abstracts using a handful of examples. In contrast, other deep learning approaches, such as Doc2Vec show lower performance. For information providers, our findings show promising results to alleviate the burden of not having enough annotated examples to deliver high-quality metadata.

In this paper, we explore *the potential of language inference, a computational model that learns general representations of natural language sentences,* as a useful feature extractor for metadata generation given a limited amount of labeled examples. In a nutshell, we provide through our work an interesting lens through which to look at the significant potential of Language Inference to benefit both information providers and users of scientific digital collections.

The rest of the paper is structured as follows: Section 2 is devoted to related work. Then we will introduce InferSent and our baseline Doc2Vec in more details. In Section 3, we describe the data used in our experiments where we empirically prove the potential of language inference for metadata generation. Finally, in Section 4, we provide a summary of our findings and point and future work.

2 Related Work

The successful semantics learned by word embeddings such as word2vec Mikolov et al. [17], Glove by Pennington, J. et al. [19], FastTex by Bojanowski, P. et al. [1] and more recently a model called ELMo by Peters, M. et al. [20] have motivated a growing body of scientific literature to develop model representations to account for longer sequences of text such as sentences. One of the first attempts to perform such a task was relying on word embeddings and applying a simple average of the sentence's word vectors, some examples of this idea are the work of [16, 21]. Recent efforts have taken two different approaches to learn sentence embeddings, *d*-dimensional continu-

ous vector representations of sentences, unsupervised, and supervised. One representative example of such efforts that use an unsupervised approach is the work of Kiros, R. et al. [14] called Skip-thought. Skip-thought builds on the core idea presented by Mikolov et al. in [17] but instead of words as a core unit of information, Kiros, R. et al. use sentences. At its core, the model uses the current sentence to predict the sentence before and after it. Formally, Skip-thought uses the framework of encoder-decoder. Encoder-decoder models have shown a lot of success in neural machine translation [11, 22] . Thus, Skip-thought builds on these ideas to do the following: an encoder maps words to a sentence vector, and a decoder is used to generate the surrounding sentences. Another model inspired by Skip-thought is the work of [6] that instead of a recurrent neural model (RNN) used in Skip-thought uses a convolutional neural network as an encoder. Then, reconstruct the input sentence and its neighbor sentences using RNNs. The main problem with both approaches is that these models are prolonged to train in massive amounts of data. Thus, researches started to focus on methods that introduced important supervised tasks in the hope that the models will learn general representations of sentences that could be used on several tasks. An instantiation of these approaches is InferSent [4] and Google Universal Embeddings [3]. Both approaches were trained using the Stanford Natural Language Inference (SNLI) dataset (more details in the description of InferSent). The main difference between the two is that Google Universal Embeddings also uses unsupervised training. In particular, web sources such as Wikipedia, web news, web question-answer pages, and discussion forums. Secondly, the model that encodes the sentence uses an encoding sub-graph architecture. What is interesting about their sub-graph architecture is that it uses an attention mechanism to compute context-aware representations of words in a sentence that considers both the ordering and identity of other words. We decided to focus on InferSent because Google Universal Embeddings incorporate sources that go beyond our goal in this paper: to explore whether Language Inference can support metadata generation even when we have limited number of annotated documents.

In the rest of this section and for self-containment, we describe the two core models used in our experimental section. To do so, we provide a summary of the terminology used on the original papers. We will first describe InferSent [4] and then Doc2vec [15].

InferSent. The work of Conneau, A. et al. [4] introduces an approach called InferSent that aims at providing a universal sentence representation trained on the Stanford Natural Language Inference (SNLI) dataset [2]. Inspired by the success of word embeddings, where pre-trained word embeddings over a large corpus have been used to tackle other tasks, Conneau, A. et al. introduced a supervised task to learn sentence embeddings also from a large corpus. In particular, they used the SNLI dataset that comprises 570K human-generated English sentence pairs that were manually labeled with one of three categories: entailment, contradiction, and neutral. The idea behind this dataset is to capture language inference, previously known as Textual Entailment (TE). The hypothesis of the work of Conneau, A. et al. is that the semantic nature of Natural Language Inference can help computational models to learn sentence embeddings in a supervised way [4]. Similar to the idea of learning word embeddings by learning d-dimensional vectors of a target word predicting its context words, the

authors argue that sentence embeddings can be learned using Language Inference at its core. Afterward, the sentence embeddings model could be used in some general classification tasks such as sentiment on movies, product review, opinion polarity, among others. To discover a sentence encoder suitable for the task, they tried several different deep learning architectures including LSTM, GRU, BiLSTM with mean/max pooling, self-attentive network, and hierarchical convnet. Through a series of experiments, the BiLSTM with max pooling was found to outperform the other variants. Thus, the model that we used in our experiments is the one based on BiLSTM with max pooling. In a nutshell, InferSent is a model that once trained on a high-quality language inference supervised machine learning task, learns to represent sentences in *d*-dimensional semantic space with the potential to be used as a universal feature representation. More details about the model can be found in the paper that introduced the model [4]. In our work, we aim at providing insights about the potential of this model for metadata generation when we have a limited number of samples and how a given algorithm can be affected once it has more samples available.

Doc2Vec. The work of Le and Mikolov [15] introduced a model known as Paragraph Vector, popularized as Doc2Vec due to its connection with its predecessor word2vec, see [18]. The idea behind Doc2Vec is to learn in an unsupervised way, fixed-length feature representations from variable-length pieces of texts, such as sentences, paragraphs, and documents [15]. At the heart of the model is the idea of learning the paragraph vectors by predicting the surrounding words in contexts sampled from the paragraph.

In a nutshell, Doc2Vec introduces the idea of mapping each paragraph as a vector, and its incorporation with the concatenation of the word vectors within the paragraph are used to predict the next word in context. Thus, the paragraph vector is shared across all contexts generated from the same paragraph but not across paragraphs. However, the word vectors are shared across paragraphs [15].

Two models were proposed by the authors of Doc2Vec to learn paragraph vectors, building on the ideas mentioned above: the Distributed memory version (PV-DM) that uses the concatenation of the paragraph vector with the word vectors to predict the next word in a text window.

The second model named Distributed Bag of Words (PV-DBOW) does not preserve word order. In other words, it ignores the context words in the input, and instead, the model is trained to predict words randomly sampled from the paragraph output.

More details of the model can be found in the paper [15]. In our experiments, we use both models and compare them with InferSent.

3 Experimental Setting

First, we describe the Evaluations Corpus followed by presenting our Evaluations Datasets. Hereafter we explain our algorithm which we use to determine a class label using Doc2Vec and InferSent. Finally, we compare different models and present the results followed by a discussion.

3.1 Experimental Set-Up

Corpus. With 29 million citations, PubMed is one of the largest digital libraries in the biomedical field of which 4,226,200 are structured abstracts. Structured abstracts, as stated by the US National Library of Medicine[1], have several advantages for authors and readers. Structured abstracts assist health professionals in selecting clinically relevant and methodologically valid journal articles. They also guide authors in summarizing the content of their manuscripts precisely, facilitate the peer-review process for manuscripts submitted for publication, and enhance computerized literature searching [9, 10]. Each structured abstract consists of five different metadata classes and each class are labeled either with the class Background, Methods, Objective, Results, or Conclusions. Also, each section can consist of several sentences. In our evaluation, we use only the structured PubMed abstracts.

Evaluation-Datasets. We aim to assign one of the five section metadata classes to the individual sentences of an unstructured abstract, e.g., sentence s belongs to the metadata class Methodology. Therefore, our datasets generally consist of the pair $< s, c >$ where s is a sentence and c is the metadata class. Our focus in the investigations is on the use case that only a few labeled data may be available, and this may affect the classification accuracy. Therefore, we examine the different approaches with data sets of different sizes. Thus, by sampling the $< s, c >$ pairs from the corpus we generate a total of 6 datasets of the following sizes: 1) 4000 pairs, 2) 24,000 pairs, 3) 48,000 pairs, 4) 100,000 pairs, 5) 200,000 pairs, and 6) 400,000 pairs. The datasets are disjoint and balanced for each metadata class.

Sentence-Embedding Models. In our evaluation, we compare the accuracy of Doc2Vec with the results that can be achieved with InferSent. Furthermore, we train Doc2Vec with a DBOW as well as with a DM architecture. For InferSent, we use the pre-trained models from [3] where one model was trained using Glove, and the second Model was trained using FastText.

Model Optimization. InferSent models are already optimized regarding hyperparameters. In order to allow a fair comparison between the different approaches, we have investigated in our experiments with Doc2Vec the hyperparameters window-size and dimension-size using grid search. We achieved the best values with a window size of 15 and a dimension size of 300. Therefore, we used in all our experiments this setting for Doc2Vec training.

[1] https://www.nlm.nih.gov/bsd/policy/structured_abstracts.html

3.2 Evaluation of the models

The steps that we used to evaluate the models can be divided into the following four steps (Figure 1):

1. Text Pre-Processing: Our pre-processing of the individual sentences of a dataset is limited to the removal of (default) stop words using the Natural Language Tool Kit (NLTK). Also we lowercase all words before training.
2. Model Training: InferSent models are already pre-trained so that we only have to train the Doc2Vec models (DBOW, DM) on our datasets. For the Doc2Vec training, we divide the described datasets into a training and a test dataset. In our experiments, we train the Doc2Vec models on 75% of the data and use the remaining data for later testing.
3. Vector Extraction: In the next step, we extract the vector representations for each sentence as well as from each model. In the case of the Doc2Vec models, we extract the sentence vectors using sentence-ids. Since the Doc2Vec models were trained with 75% of a dataset, we also need a vector representation of the remaining 25% of the dataset's records for later testing. Thus, to get a vector representation from Doc2Vec model after training, we use the $infer\_vector$ function. In the case of InferSent since we use pre-trained model, we rely on the $encoder$ function to get a vector representation for all the sentences.
4. Logistic Regression Model Training: Next, we use the extracted sentence-vectors to train a regression model. The goal is for the model to learn a metadata class from the extracted vectors. In our experiments, we train one regression model for each Sentence Embedding Model we described, using 75% of the data set for training and the remaining 25% for testing of the regression model. For each dataset and model, we measure the precision, recall, and F1 score.

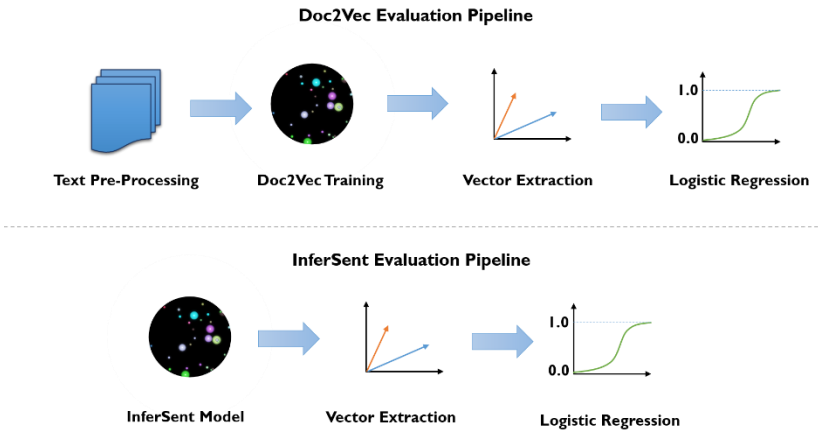
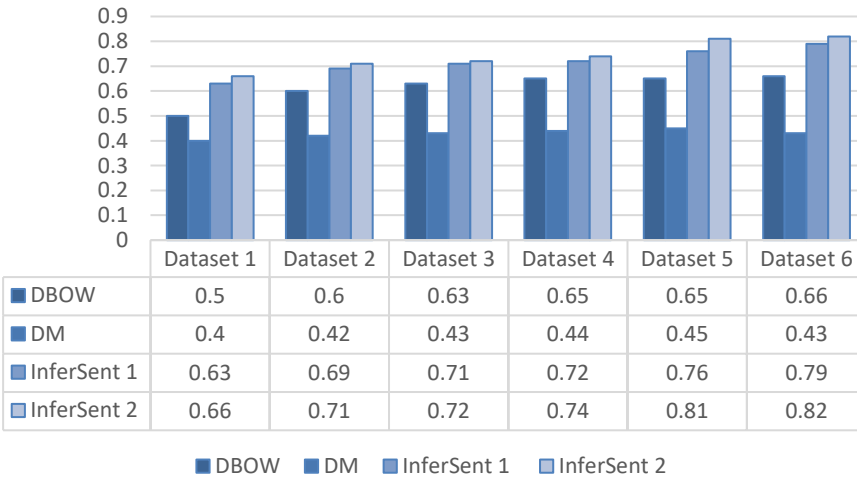


Fig. 1. Summarization of the different evaluation steps for Doc2Vec and InferSent models.

We have performed the steps described above on all data sets, and show a summary of the results in Figure 2. As expected, the quality, measured by F1 scores, for all models increases with the size of the dataset. What surprised us, however, was how much better the InferSent models perform in comparison to Doc2Vec models, although Doc2Vec was trained on the medical abstracts and InferSent was not. The pre-trained InferSent models seem to have learned a universal and inherent sentence semantics, which is also reflected in the abstracts' sentences and in which the metadata classes can be determined with a certain degree of success (F1 score up to 82%). This makes these models interesting as an alternative approach for complex classification tasks for metadata generation if only a few training data is available.



	Dataset 1	Dataset 2	Dataset 3	Dataset 4	Dataset 5	Dataset 6
DBOW	0.5	0.6	0.63	0.65	0.65	0.66
DM	0.4	0.42	0.43	0.44	0.45	0.43
InferSent 1	0.63	0.69	0.71	0.72	0.76	0.79
InferSent 2	0.66	0.71	0.72	0.74	0.81	0.82

Fig. 2 Overall F1 Score Summary

Why does Doc2Vec trained with DBOW lead to better results compared to DM? The DBOW model performs much better than the DM model, although it was shown in [15] that the DM architecture on average leads to better results because the order of the words is preserved. How can this be reconciled with our results? In [15], the documents used for training consist not of single sentences but instead of many paragraphs. In our case, the document size seems to be the decisive factor for the rather poor performance of the DM architecture.

Doc2Vec vs. InferSent Analysis

We contrast the results of the best Doc2Vec model (DBOW) and the best InferSent model (InferSent 2) in Tables 1 and 2. Let us first consider Table 1, where we show the

performance of the models per metadata class using the smallest dataset. We can observe that DBOW is outperformed in every metadata class by a large margin. In average, the InferSent model achieves 16 points above the DBOW model. When contrasting per metadata class we can observe that both models achieved the lowest F1 score in the 'Background' metadata class. However, even in this case the differences are interesting to discuss: InferSent 2 achieving a 0.53 F1 score while DBOW fifteen points below. Indeed, the differences that we can observe are remarkable.

Furthermore, both models achieve the highest F1 score in the 'Methods' metadata class. However, InferSent 2 with an F1 score of 0.76, thirteen points above DBOW. We can observe similar behavior in Table 2, where we show the results of the models using the biggest dataset available. Once more, the 'Background' metadata class was the most difficult one for both models. However, InferSent 2 performed up to 25 points better than DBOW achieving an F1 score of 0.80. We observe the same behavior when comparing the metadata class with the highest F1 score: InferSent 2 with more than 20 points above DBOW.

What is also remarkable is that DBOW obtained an F1 score average of 0.66 with the biggest dataset while InferSent 2 was already able to obtain the same average but with the smallest dataset and that means with 1% of the data that DBOW had to use to reach 0.66. This finding reveals the impact of Language Inference and its potential to help ongoing efforts for metadata generation in different fields.

In summary, we can observe that models using Language Inference outperformed Doc2Vec models by a large margin no matter the size of the dataset used. To assess the statistical significance of the differences between DBOW and InferSent 2 over all the datasets, we used a t-test. With a p-value of 1.7397E-17 ($p < 0.001$) we can conclude that differences between the models are significant. Furthermore, we also performed a t-test analysis over the two InferSent models because we observed small differences and even no differences in F1 scores, for instance, in Dataset 4. With a p-value of 0.0735 ($p>0.001$) we can conclude that differences observed are not significant. Thus, using a pre-trained model with Glove vectors or FastText vectors resulted in negligible margin differences.

For completeness of our analysis, we also calculated a t-test between DBOW and DM, and with a p-value of 3.0975E-16, we can conclude that differences between the two models are indeed statistically significant.

Despite an average of 0.66 F1 score in the smallest dataset with the InferSent model, the result obtained look promising. Indeed, the model could be a valuable alternative in cases where getting more annotated data is unrealistic.

One way to improve our current results is to enhance the InferSent model by adding concept embeddings from the Unified Medical Language System UMLS, such as in [24] but constraining them to maintain the semantic associations that exist. This strategy is known in the community as retrofitting, see [5], and it has been shown to improve word embeddings in general.

Our proposed enhancement constitutes part of our current efforts in our attempt to tune the model and assess the impact on its performance.

Table 1. Results with DBOW and InferSent 2 using Dataset 1

	Precision	Recall	F1
Background			
-DBOW	0.37	0.38	0.38
-InferSent 2	0.54	0.52	0.53
Conclusions			
-DBOW	0.45	0.50	0.47
-InferSent 2	0.64	0.70	0.67
Methods			
-DBOW	0.61	0.65	0.63
-InferSent 2	0.74	0.79	0.76
Objective			
-DBOW	0.48	0.45	0.47
-InferSent 2	0.65	0.56	0.61
Results			
-DBOW	0.59	0.50	0.54
-InferSent 2	0.72	0.73	0.73
AVG			
-DBOW	0.50	0.51	0.50
-InferSent 2	0.66	0.66	0.66

Table 2. Results with DBOW and InferSent 2 using Dataset 6

	Precision	Recall	F1
Background			
-DBOW	0.57	0.52	0.55
-InferSent 2	0.81	0.79	0.80
Conclusions			
-DBOW	0.68	0.68	0.68
-InferSent 2	0.83	0.84	0.84
Methods			
-DBOW	0.72	0.77	0.75
-InferSent 2	0.84	0.88	0.86
Objective			
-DBOW	0.67	0.62	0.64
-InferSent 2	0.79	0.84	0.80
Results			
-DBOW	0.67	0.74	0.70
-InferSent 2	0.85	0.85	0.83
AVG			
-DBOW	0.66	0.67	0.66
-InferSent 2	0.82	0.83	0.82

4 Conclusions and Future Work

In this paper, we have shown the potential of Language Inference for metadata generation. In particular, we evaluated InferSent considering the limited availability of annotated data. We focus on the biomedical field in the challenging task of annotating abstracts with distinct labels such as 'background', 'objective', 'methods', 'results', and 'conclusions'.

We showed the stability of performance of language inference as we vary the number of samples that a classification algorithm can have compared to the deep learning model Paragraph vector to assess the value. To our surprise, the language inference model outperformed Doc2Vec by a significant margin in the experiments that we performed. Even with a handful of examples using a logistic regression model trained using language inference vector representations, we were able to achieve promising results. These findings suggest the value of the role language inference as the basis for metadata generation, opening a new opportunity for information providers to support users and their complex information needs.

To further assess the quality of language inference in our task, we will conduct a user study with our experts to evaluate a representative sample of manuscripts that currently do not have structured metadata. To do so, we will contrast the level of agreement between our experts and the results of the different models trained using different sizes of samples. As another future line of work, we will also explore the potential of language inference to generate descriptive summaries of our ongoing efforts on semantic facettation in Pharmaceutical Collections [25].

Finally, we believe we have just started to see the potential of Language Inference for Digital Libraries using metadata generation in biomedicine as a use case. However, some other tasks relevant to our Digital library community could also benefit and may be worth investigating. For instance, consider the task of ranking documents or assessing the semantic similarity between documents.

We hypothesize that Language Inference could also have an impact on some other relevant, challenging tasks within our community, such as advancing ongoing efforts for automatic subject indexing of shorts texts [23].

References

1. Bojanowski, P. et al.: Enriching Word Vectors with Subword Information. Trans. Assoc. Comput. Linguist. 5, 135–146 (2016).
2. Bowman, S.R. et al.: A large annotated corpus for learning natural language inference. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics (2015).
3. Cer, D. et al.: Universal Sentence Encoder. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. pp. 169–174 Association for Computational Linguistics, Brussels, Belgium (2018).
4. Conneau, A. et al.: Supervised Learning of Universal Sentence Representations from Natural Language Inference Data. In: Proceedings of the 2017 Conference on Empirical

Methods in Natural Language Processing. pp. 670–680 Association for Computational Linguistics, Copenhagen, Denmark (2017).
5. Faruqui, M. et al.: Retrofitting Word Vectors to Semantic Lexicons. In: Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 1606–1615 Association for Computational Linguistics (2015).
6. Gan, Z. et al.: Learning Generic Sentence Representations Using Convolutional Neural Networks. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. pp. 2390–2400 Association for Computational Linguistics, Copenhagen, Denmark (2017).
7. González Pinto, J.M.; Balke, W.-T.: Can Plausibility Help to Support High Quality Content in Digital Libraries? In: TPDL 2017 – 21st International Conference on Theory and Practice of Digital Libraries. pp. 169–180 Springer International Publishing, Thessaloniki, Greece. (2017).
8. González Pinto, J.M., Balke, W.-T.: Assessing plausibility of scientific claims to support high-quality content in digital collections. Int. J. Digit. Libr. 19, 59, 1–14 (2018).
9. Haynes, R.B. et al.: More Informative Abstracts Revisited. Ann. Intern. Med. 113, 1, 69–76 (1990).
10. Hayward, R.S.A. et al.: More Informative Abstracts of Articles Describing Clinical Practice Guidelines. Ann. Intern. Med. 118, 9, 731–737 (1993).
11. Kalchbrenner, N., Blunsom, P.: Recurrent Continuous Translation Models. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. pp. 1700–1709 Association for Computational Linguistics, Seattle, Washington, USA (2013).
12. Kilicoglu, H. et al.: SemMedDB: A PubMed-scale repository of biomedical semantic predications. J. Bioinforma. 28, 23, 3158–3160 (2012).
13. Kim, Y.: Convolutional Neural Networks for Sentence Classification. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 1746–1751 Association for Computational Linguistics, Doha, Qatar (2014).
14. Kiros, R. et al.: Skip-Thought Vectors. In: Cortes, C. et al. (eds.) Advances in Neural Information Processing Systems 28. pp. 3294–3302 Curran Associates, Inc. (2015).
15. Le, Q., Mikolov, T.: Distributed Representations of Sentences and Documents. In: Jebara, E.P.X. and T. (ed.) International Conference on Machine Learning - ICML 2014. pp. 1188–1196 PMLR, Bejing, China (2014).
16. Lev, G. et al.: In defense of word embedding for generic text representation. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics). 9103, 35–50 (2015).
17. Mikolov, T. et al.: Distributed Representations of Words and Phrases and Their Compositionality. In: Proceedings of the 26th International Conference on Neural Information Processing Systems. pp. 3111--3119 Curran Associates Inc., Lake Tahoe, Nevada (2013).
18. Mikolov, T. et al.: Efficient Estimation of Word Representations in Vector Space. In: Proceedings of the International Conference on Learning Representations (ICLR 2013). pp. 1–12 arXiv, Scottsdale, Arizona USA (2013).

19. Pennington, J. et al.: Glove: Global Vectors for Word Representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 1532–1543 Association for Computational Linguistics, Doha, Qatar (2014).
20. Peters, M. et al.: 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). pp. 2227–2237 Association for Computational Linguistics, New Orleans, Louisiana (2018).
21. S. Arora, Y. Liang, and T.M.: A simple but tough-to-beat baseline for sentence embeddings. In: 5th International Conference on Learning Representations (ICLR 2017). , Toulon, France (2017).
22. Sutskever, I. et al.: Sequence to Sequence Learning with Neural Networks. NIPS. 9 (2014).
23. Toepfer, M., Seifert, C.: Content-Based Quality Estimation for Automatic Subject Indexing of Short Texts Under Precision and Recall Constraints. In: Méndez, E. et al. (eds.) Digital Libraries for Open Knowledge (TPDL 2018). pp. 3–15 Springer International Publishing, Cham (2018).
24. De Vine, L. et al.: Medical Semantic Similarity with a Neural Language Model. In: Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management. pp. 1819–1822 ACM, New York, NY, USA (2014).
25. Wawrzinek, J., Balke, W.T.: Semantic facettation in pharmaceutical collections using deep learning for active substance contextualization. In: International Conference on Asian Digital Libraries. pp. 41–53 Springer, Bangkok, Thailand (2017).
26. Zhang, Y., Wallace, B.: A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification. In: Proceedings of The 8th International Joint Conference on Natural Language Processing. pp. 253–263 Asian Federation of Natural Language Processing, Taipei, Taiwan (2017).