German Medical Data Sciences 2023 — Science. Close to People.
R. Röhrig et al. (Eds.)

doi:10.3233/SHTI230710

Concept Graphs: A Novel Approach for Textual Analysis of Medical Documents

Franz MATTHIES[a,1], Christoph BEGER[a,b], Ralph SCHÄFERMEIER[a], and Alexandr UCITELI[a]
[a] *Institute for Medical Informatics, Statistics and Epidemiology (IMISE), Leipzig University, Germany*
[b] *Growth Network CrescNet, Leipzig University, Germany*

Abstract. The task of automatically analyzing the textual content of documents faces a number of challenges in general but even more so when dealing with the medical domain. Here, we can't normally rely on specifically pre-trained NLP models or even, due to data privacy reasons, (massive) amounts of training material to generate said models. We, therefore, propose a method that utilizes general-purpose basic text analysis components and state-of-the-art transformer models to represent a corpus of documents as multiple graphs, wherein important conceptually related phrases from documents constitute the nodes and their semantic relation form the edges. This method could serve as a basis for several explorative procedures and is able to draw on a plethora of publicly available resources. We test it by comparing the effectiveness of these so-called *Concept Graphs* with another recently suggested approach for a common use case in information retrieval, document clustering.

Keywords. Word Embeddings, Transformer Models, Document Clustering, Natural Language Processing, Graphs, Medical Documents

1. Introduction

The use of Natural Language Processing (NLP) in the medical field can enable healthcare professionals to analyze and extract meaningful information from large volumes of unstructured medical data, such as electronic health records (EHRs), medical reports, and clinical notes. Although this domain-specific usage is not new, there are still major problems in accessing freely available language resources in non-English speaking contexts. However, these are a basic prerequisite for the follow-up development of algorithms and models capable of analyzing clinical speech. In Germany, some progress has been made in the last five years, not least in the context of the MII[2] – a large-scale national funding initiative – in particular in the SMITH consortium [1], and a few corpora are more or less freely available [2–5]. But the issue of data privacy protection is still a big roadblock for making these and similar resources available to the NLP community and, by extension, to researchers and research programs that want to gather information from unstructured medical data, e.g. free text fields in databases or privately donated EHRs, etc.

[1] Corresponding Author: Franz Matthies, franz.matthies@imise.uni-leipzig.de
[2] https://www.medizininformatik-initiative.de/en/start

Given the sparsity or non-existence of said domain-specific models, we in the MII junior research group 'Terminology and Ontology-based Phenotyping (TOP) looked for a way to use general-purpose models and algorithms to represent the structure of (clinical) documents as graphs and, moreover, to model these structures in such a way that they are both inherently informative on the one hand and that connections can be made across document boundaries on the other hand. Despite being an early prototype, this method seems to be promising with regard to both the former and the latter.

2. Related Work

Graph-based NLP is certainly nothing new in the scientific community. In fact, Mihalcea and Radev [6] dedicate an entire book to the synopsis of the two formerly distinct fields of Graph Theory and NLP and cite various theoretical and practical studies. Not least in the field of semantics, where our approach can be placed, they refer to methods that create networks by generating their links based on all nouns contained in large corpora either on the basis of conjunctions (i.e., and/or) or frequent co-occurrences. [6 p. 123 et sqq.]

More specifically however, when it comes to document clustering, approaches can be found that use WordNet®, for example, to create a document graph for each document, then build a dissimilarity matrix for each combination of document graphs, and use this to perform the clustering process. [7] Or so-called document embeddings are created and combined in a graph (with one document each as a node and edges between them, if their cosine similarity exceeds a certain threshold). Subsequently, a Graph Community detection algorithm is performed to find nodes (i.e. documents) that are well connected to each other. [8]

3. Methods

Whilst we share some aspects with each of the methods described in the previous section, our approach doesn't rely on outside resources (barring the general-purpose models for pre-processing and phrase-embedding) – which could be hard to come by in non-English languages or specific/niche domains. Furthermore, we neither regard only documents as nodes in a graph nor do we solely create intra-document network representation. Rather, we consider the phrases that make up documents, look at their relatedness to each other and the corpus-inherent concepts, and by extension embed documents into this web. This, on the one hand, allows us to generate document representations that are not graphs but rather vectors like in [9]. These, in turn, provide us the means to perform for instance common clustering algorithms. On the other hand, we get an interpretable visualization of a document corpus where important/common phrases and subsequent graph-driven search routines could be employed to find well-connected documents or even to generate a corpus inherent terminology.

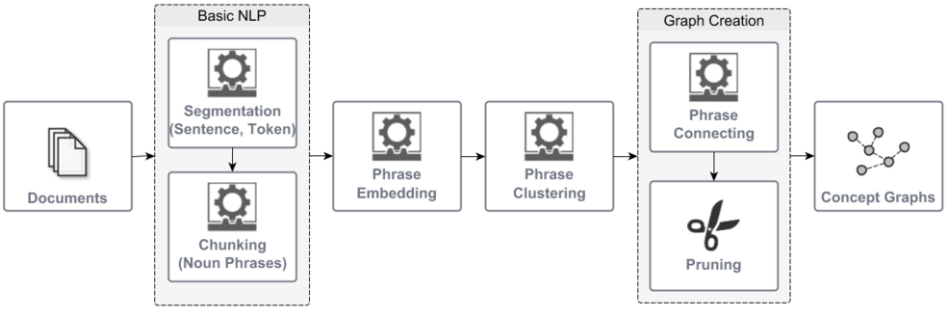

Figure 1. Schematic workflow for the creation of Concept Graphs

3.1. Concept Graphs

We have conceived a multi-graph representation of a document corpus in which the respective nominal phrases are grouped into conceptually expressive clusters and represented as nodes with their semantic proximity (enriched/modified by various methods) realized in the form of the edges. As outlined in Figure 1, the creation process of these so-called *Concept Graphs* is first to extract nominal phrases for each document[3] and embed the resulting phrases into a vector space. We tested some methods to generate these word embeddings and found that models trained with Siamese BERT-Networks (SBERT) [10] produced the best results. For this, there are a variety of models for different domains and tasks available (we utilized the ones in Table 1) and depending on which one is used, a vector can have many dimensions (e.g. 1024 in the upper spectrum), which would be unfavorable for the preliminary clustering task that generates the concept clusters. Therefore, in the second step, we reduced the dimensions with the UMAP algorithm [11] and then deduced the number of potential groupings of semantically related phrases (i.e. the aforementioned concept clusters) with the elbow method (using the distortion score as a scoring parameter) to have a target number of clusters for the subsequent clustering algorithm[4].

Table 1. The models from https://huggingface.co/ that were used for the Phrase Embedding step according to the domain and language

	Medical	**Non-Medical**
English	FremyCompany/BioLORD-STAMB2-v1	paraphrase-albert-small-v2
German	Sahajtomar/German-semantic	—

Now, the thusly-generated clusters of phrases undergo the graph creation step. Each phrase connects to other phrases within its cluster dependent on specific parameters (for instance, cosine similarity between their affiliated embedding vectors and their string similarity). We now end up with heavily connected graphs that contain too much noise

[3] This linguistic preprocessing step was realized with SPACY (https://spacy.io/)

[4] Naturally, the deduction of the number of clusters is not needed if a clustering algorithm is used that does not require this in advance (e.g. Affinity Propagation). However, we found a simple K-Means clustering method to be favorable and this one needs a target cluster count.

in the form of less meaningful edges between phrases, so we need a way to remove a portion of them. Instead of naively cutting low-weighted edges iteratively up to a specific threshold, we choose a different algorithm where the edges will be sorted and subsequently removed according to their *significance* to the graph as a whole. [12] These pruned graphs now have fewer and more relevant connections, as can be seen in Figure 2, where the edge thickness depicts the strength of a connection from thick to dashed (in the actual implementation each edge carries in fact even more information like e.g. a floating point value for the strength of the connection, its significance value, etc.).

Each one of those graphs may describe a certain topic dependent on the overall corpus, e.g. a cardio related graph, a graph that contains phrases associated with gastroenterology, etc. At this point, we can already envision building a corpus dependent ontology or exploring the document corpus by means of its terms via graph specific algorithms, as each phrase node also connects (invisible in Figure 2) to a set of documents.

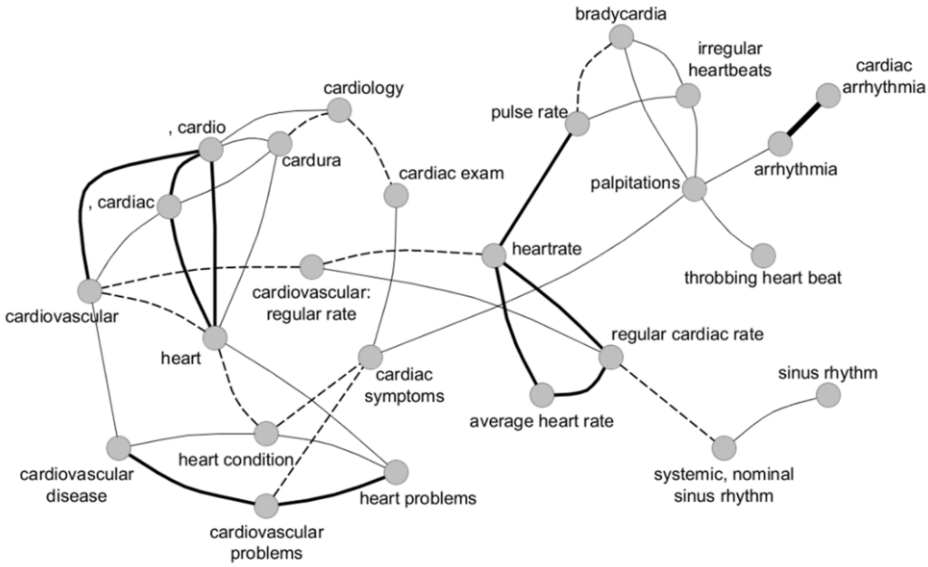

Figure 2. An exemplary close-up view for a pruned cardio-related *Concept Graph*

3.2. Document Clustering

In the preceding section we talked about meaningful connections between nodes in the graphs but omitted a definition thereof. We could look manually at each one and deduce whether the connections between phrases make sense but naturally, that is not really feasible. So, one idea is to evaluate this meaningfulness by utilizing the graph structure for a document clustering task, as described in the following.

In order to group documents according to their content, you need a suitable representation of their relationship to each other in terms of content. At a basic level, we could simply represent the former by the (non-)occurrence of words from a fixed

vocabulary (of the entire corpus), the so-called *bag-of-words* model. Alternatively, we could go a step further and keep track of *word n-grams* in each document, that is, the co-occurrence of words in a specific window. Other methods include the construction of *tf-idf* matrices that measure how important each word is to a document given a corpus and furthermore the *embedding of words* in a vector space and their subsequent combination to create document embeddings, where one can derive similarity by means of their distance in said space.

Table 2. Basic (linguistic) properties of the corpora used for the document-clustering task. An important property is the ratio between #Documents and #Tokens as it gives an indication of how wordy the documents are.

Corpus	Categories	Documents	Sentences	Token	Token/Doc	Language
germed	19	241	16,511	167,632	696	de
engmed	15	289	8,940	135,924	470	en
ng20	4	700	30,671	282,202	403	en
ng20 (long)	9	8,131	293,457	2,698,029	332	en
scopus	5	500	1,000	113,238	227	en
scopus (long)	7	2,800	5,599	617,622	221	en

We present another approach in which we utilize the aforementioned *Concept Graphs* of phrases that are taken from the actual document corpus and use the similarity between phrases, which in turn relate to a set of documents, to calculate a score of how strongly a document is associated with a concept. Our procedure is similar to [13] and even though they use concept clusters as well, they generate the matrices used for clustering (so-called document concept matrices) based on the connection between the words that occur in a document and a concept using *tf-idf* and only single words and not phrases, i.e. groups of words. In contrast, we generate the matrices by either computing a value for each concept to each document based on the *eigenvector centrality* of the document node in the *Concept Graph* (one implementation of this measure would be Google's *PageRank* algorithm) or by performing *shortest path* analyses on the nodes to determine which documents are close to each other in a given graph given their phrases.

3.2.1. Data

Table 2 gives a concise overview of some basic linguistic properties of the corpora we used for our document clustering task. Both *ng20* and *scopus* are the same as in [13]. Since we wanted to focus on medical/clinical data, we only selected these two as a system's comparison baseline for their relative category spread (which could be varied, i.e. increased) and their potential to improve the score.

- *germed*: This corpus consists of 63 semi-synthetic documents from *GraSSCo* [2] and 158 in-house discharge summaries. Medical professionals categorized them into 19 different categories.
- *engmed*: These medical transcription samples were scraped from a website[5] and then transformed into a CSV file by the user Tara Boyle[6]

[5] https://www.mtsamples.com/

[6] https://www.kaggle.com/tboyle10/medicaltranscriptions

- For the other four corpora (*ng20*, *ng20 (long)*, *scopus* and *scopus (long)*) please see [13] as they are already described there in detail.

Table 3. Results for the presented system (*ConceptGraph*) for each corpus measured by two common metrics, *ARI* & *Purity*, in comparison with [9] – reimplemented with word embeddings generated from SBERT (*WEClustering*), as well as the scores reported in their paper (*WEClustering (paper)*). Only the best scores (higher average) from either KMeans (K) or Agglomerative (A) Clustering are given for each system.

Corpus	System	ARI	Purity	Cluster Type
germed	**ConceptGraph**	**0.694**	**0.618**	A
	WEClustering	0.551	0.573	K
	WEClustering (paper)	–	–	–
engmed	**ConceptGraph**	**0.243**	**0.492**	K
	WEClustering	0.220	0.462	K
	WEClustering (paper)	-	-	-
ng20	**ConceptGraph**	**0.446**	**0.680**	A
	WEClustering	0.420	0.660	K
	WEClustering (paper)	0.344	0.637	K
ng20 (long)	**ConceptGraph**	**0.558**	**0.664**	K
	WEClustering	0.397	0.626	K
	WEClustering (paper)	0.165	0.409	K
scopus	ConceptGraph	0.960	0.984	K
	WEClustering	**0.975**	**0.990**	K
	WEClustering (paper)	0.893	0.956	K
scopus (long)	ConceptGraph	0.899	0.938	A
	WEClustering	**0.921**	**0.964**	K
	WEClustering (paper)	0.611	0.775	A

4. Results

To compare the results of our system, the system from [13] as implemented by us with the new *word embedding* generation method, and the results as documented in the aforementioned paper, we use two clustering evaluation metrics, *ARI* (*adjusted rand score*) and *Purity*. The former, in this specific case, reflects the similarity between the gold labels of the documents, when considered as a clustering result, and the assignment by the clustering method – one can think of it as the accuracy of the algorithm where a value of 1 denotes a perfect match. As the name implies, the latter in turn indicates the extent to which each individual cluster of a whole result contains identical document label data points, and a value of 1 means that a cluster contains only one class.

Although the results as seen in Table 3 are not entirely conclusive, they do seem to suggest a tendency. On the one hand, it can be seen that vectorizing the documents using *Concept Graphs* by utilizing graph algorithms leads to a higher score in both metrics for most of the corpora used here. On the other hand, using phrases instead of words and/or take advantage of *transformer models* to generate the *word embeddings* also seems to result in significantly better scores (the results reported in [13] are consistently worse than those generated by either its re-implementation or those generated by using *Concept Graphs*). Yet another observation is that where *Concept Graphs* perform worse against the simpler *tf-idf* variant (i.e. *scopus* corpus), the delta is low (<0.03 for both metrics) for

one, and additionally, both methods achieve near-perfect clustering (>0.9 for both metrics).

5. Discussion

In the previous section, an outlier in the results was highlighted that needs at least a little scrutiny: we have already remarked on the low delta between the *scopus* results and the strong performance per se for both systems (even compared to those noted in [13]). If we look at Table 2, there is one value that catches the eye, it's the relatively low ratio of tokens per document. So it might be that the fewer tokens (and therefore phrases) there are in a document, the less meaningful *Concept Graphs* become. But there seems to be at least no clear correlation since the corpus with the next lowest ratio (*ng20 (long)*) has indeed the largest delta in favor of our approach. Another consideration is that the *scopus* corpus is the only one in the present collection that comes from scholarly, peer-reviewed texts (i.e. mainly journals) and thus generally uses high-quality language from the standpoint of spelling and syntax. Both of these circumstances taken together could be a starting point for further research. But since clinical texts de facto contain more ungrammatical sentence structures, spelling errors, and the like, *Concept Graphs* could be most beneficial for this domain.

To induce an improvement in the performance of our method, we could further investigate the creation and subsequential exhaustive use of the graph structures. Our prototype for instance uses a relatively naive method to create the edges in the graphs. Moreover, the values for the *document concept matrices* (that serve as a basis for the clustering task) are created using only simple shortest-path algorithms. For the latter, other graph algorithms or even graph clustering methods can be investigated.

6. Conclusion

We have demonstrated a method that provides a different perspective of the content structure of a document, called *Concept Graphs* and in an experimental setup, using the scenario of document clustering, we provide strong evidence that a document representation derived from them is – at least for the medical domain – predominantly better suited than a comparable state-of-the-art representation inferred from *tf-idf* matrices.

Unlike other methods that use recent (sentence-)transformer-based approaches to embed entire documents into a vector space and in turn create graphs from them, for example, our approach however does not abstract away from the important phrases that make up a document. Thus, our *Concept Graphs* are not only a means to an end for clustering, for instance, but could also be used to extract corpus-specific medical terminologies or to support keyword-guided document retrieval through appropriate graph algorithms. And all that even without domain-specific algorithms or models, which are few and far between in the medical domain for most languages other than English. However, we only applied basic graph operations/analyses to make them more meaningful or to derive corresponding information, and further research on what potential lies in them should be conducted.

Declarations

Conflict of Interest: The authors declare that there is no conflict of interest.
Contribution of the Authors: FM: conception & implementation of the method, conduct of the experiments and writing of the manuscript; CB, RS, AU: discussion contributions to the conception and proof-reading of the manuscript. All authors have approved the manuscript as submitted and accept responsibility for the scientific integrity of the work.
Funding: The SMITH-consortium was funded by the German Federal Ministry of Education and Research (grant number: 01ZZ1803A), the 'Terminology and Ontology-based Phenotyping (TOP) project was funded by the German Federal Ministry of Education and Research (grant number: 01ZZ2018)
Implementation: https://github.com/fmatthies/concept-graphs.git

References

[1] Winter A, Stäubert S, Ammon D, Aiche S, Beyan O, Bischoff V, et al. Smart Medical Information Technology for Healthcare (SMITH). Methods Inf Med. 2018;57(S 01):92–105.
[2] Modersohn L, Schulz S, Lohr C, Hahn U. GRASCCO — The First Publicly Shareable, Multiply-Alienated German Clinical Text Corpus. German Medical Data Sciences 2022 – Future Medicine: More Precise, More Integrative, More Sustainable! 2022;66–72.
[3] Lohr C, Buechel S, Hahn U. Sharing Copies of Synthetic Clinical Corpora without Physical Distribution — A Case Study to Get Around IPRs and Privacy Constraints Featuring the German JSYNCC Corpus. In: Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018) [Internet]. Miyazaki, Japan: European Language Resources Association (ELRA); 2018. Available from: https://aclanthology.org/L18-1201
[4] Borchert F, Lohr C, Modersohn L, Langer T, Follmann M, Sachs JP, et al. GGPONC: A Corpus of German Medical Text with Rich Metadata Based on Clinical Practice Guidelines. arXiv:200706400 [cs] [Internet]. 2020; Available from: http://arxiv.org/abs/2007.06400
[5] Kittner M, Lamping M, Rieke DT, Götze J, Bajwa B, Jelas I, et al. Annotation and initial evaluation of a large annotated German oncological corpus. JAMIA Open [Internet]. 2021;4(2). Available from: https://doi.org/10.1093/jamiaopen/ooab025
[6] Mihalcea RF, Radev DR. Graph-based natural language processing and information retrieval. Cambridge New York Melbourne Madrid Cape Town Singapore São Paulo Delhi Mexico City: Cambridge University Press; 2011. 192 p.
[7] Hossain MS, Angryk RA. GDClust: A Graph-Based Document Clustering Technique. In: Seventh IEEE International Conference on Data Mining Workshops (ICDMW 2007). Omaha, NE, USA: IEEE; 2007. p. 417–22.
[8] Rao RN, Chakraborty M. Vec2GC -- A Graph Based Clustering Method for Text Representations [Internet]. arXiv; 2023. Available from: http://arxiv.org/abs/2104.09439
[9] Wang L, Gao C, Wei J, Ma W, Liu R, Vosoughi S. An Empirical Survey of Unsupervised Text Representation Methods on Twitter Data. In: Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020) [Internet]. Online: Association for Computational Linguistics; 2020. p. 209–14. Available from: https://aclanthology.org/2020.wnut-1.27
[10] Reimers N, Gurevych I. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. 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) [Internet]. Hong Kong, China: Association for Computational Linguistics; 2019. p. 3982–92. Available from: https://aclanthology.org/D19-1410
[11] McInnes L, Healy J, Saul N, Großberger L. UMAP: Uniform Manifold Approximation and Projection. Journal of Open Source Software. 2018;3(29):861.
[12] Dianati N. Unwinding the hairball graph: Pruning algorithms for weighted complex networks. Phys Rev E [Internet]. 2016;93(1). Available from: https://link.aps.org/doi/10.1103/PhysRevE.93.012304
[13] Mehta V, Bawa S, Singh J. WEClustering: word embeddings based text clustering technique for large datasets. Complex Intell Syst. 2021;7(6):3211–24.