Facilitating the Analysis of COVID-19 Literature Through a Knowledge Graph

Bram Steenwinckel[0000−0002−3488−2334], Gilles Vandewiele[0000−0001−9531−0623], Ilja Rausch[0000−0002−9170−3021], Pieter Heyvaert[0000−0002−1583−5719], Ruben Taelman[0000−0001−5118−256X], Pieter Colpaert[0000−0001−6917−2167], Pieter Simoens[0000−0002−9569−9373], Anastasia Dimou[0000−0003−2138−7972], Filip De Turck[0000−0003−4824−1199], and Femke Ongenae[0000−0003−2529−5477]

IDLab, Ghent University – imec, Technologiepark-Zwijnaarde 126, Ghent, Belgium
{firstname}.{lastname}@ugent.be

Abstract. At the end of 2019, Chinese authorities alerted the World Health Organization (WHO) of the outbreak of a new strain of the coronavirus, called SARS-CoV-2, which struck humanity by an unprecedented disaster a few months later. In response to this pandemic, a publicly available dataset was released on Kaggle which contained information of over 63,000 papers. In order to facilitate the analysis of this large mass of literature, we have created a knowledge graph based on this dataset. Within this knowledge graph, all information of the original dataset is linked together, which makes it easier to search for relevant information. The knowledge graph is also enriched with additional links to appropriate, already existing external resources. In this paper, we elaborate on the different steps performed to construct such a knowledge graph from structured documents. Moreover, we discuss, on a conceptual level, several possible applications and analyses that can be built on top of this knowledge graph. As such, we aim to provide a resource that allows people to more easily build applications that give more insights into the COVID-19 pandemic.

Keywords: COVID-19 · Knowledge Graph Creation · Network Analysis · Graph Embeddings

1 Introduction

In 2019, the World Health Organization (WHO) was alerted that an infectious disease was identified in Wuhan, Central China. Now, in 2020 this disease caused by severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) has spread globally, resulting in the commonly known COVID-19 pandemic [2].

This virus spread itself easily. Over 20,000,000 people, from all over the world, were infected in a short amount of time [5]. In response to this pandemic, on March 16th, 2020, researchers and leaders from the Allen Institute for AI, Chan Zuckerberg Initiative (CZI), Georgetown University's Center for Security and Emerging Technology (CSET), Microsoft, and the National Library of

Medicine (NLM) at the National Institutes of Health released a freely available dataset of scholarly literature about COVID-19, SARS-CoV-2, and the coronavirus group [1]. The goal of releasing such a dataset was to apply recent advances in Natural Language Processing (NLP) and other Artificial Intelligence (AI) techniques to generate new insights in support of the on-going fight against this infectious disease.

The goal of this study was to transform this original dataset into a knowledge graph. Having this data in a graph-based format allows us to reap several benefits. First, by linking concepts to external resources, the dataset can be enriched with knowledge that was initially not available. As an example, linking the studies in the dataset to DBpedia resources of their respective country allows us to explore potential correlations with, for example, geographic and demographic data. Second, the edges in the graph explicitly represent a relation between pairs of entities, which can be taken into account during analysis of the dataset. These edges can result in more precious insights.

This advantage has already been illustrated in several studies. It has been shown that taking into account citation information of a paper and its content can produce useful representations that adopt the idea and benefits of linked data [19]. In this paper, we show the full pipeline to construct such a graph and explain how we've made this resource publicly available, taking into account the Findable, Accessible, Interoperable, and Reusable (FAIR) principles. We also illustrate the advantages of having a knowledge graph of this data by conducting several preliminary analyses and giving potential directions for further applications.

The remainder of this paper is organized as follows. We first discuss some related initiatives that built upon this dataset in Section 2. Next, we provide an overview of our architecture used to transform the original dataset into a semantic representation in Section 3. In Section 4, we then list all resources which were integrated and linked to this knowledge graph, and what type of new information they bring to the knowledge graph. Finally, we discuss some potential applications and provide some preliminary analyses in Section 5. We conclude our work in Section 6.

2 Related Work

Based on the provided dataset, several other initiatives started to build a knowledge graph, build applications on top of them or used them in order to facilitate the analysis of other researchers of this vast amount of information. The Covid Graph project [17], led by a diverse team based in Germany, is probably the largest initiative. They created a COVID-19 knowledge graph by mining the CORD-19 dataset, linking it to the NCBI Gene Database and other gene ontologies to enable scientific analysis. They currently provide a visual graph explorer and a NEO4J browser as applications. Another notable initiative is the CORD-

19-on-FHIR dataset [10]: a Linked Data version of the CORD-19 represented in FHIR RDF. Here, the titles and abstracts were parsed, and more than 180,000 Condition, 32,000 medication and 100,000 procedure instances could be identified and linked. Similar to our initiative, the Dice research group started to build a knowledge graph by creating triples using RDFLIB [1] in Python [23]. The COVID19DS knowledge graph mainly links the papers, authors, refs and cites together in one knowledge graph without looking into the actual content of the papers.

Other knowledge graphs were designed with a particular task in mind. The CORD-SEMANTICTRIPLES initiative derived knowledge from the Semantic MEDLINE database (SemMedDB), reflecting documents also in the COVID-19 corpus [2]. SemMedDB contains concept-relation-concept semantic triples, or predications. After extracting 106K semantic predications, they imported these into a network and applied network centrality metrics (degree, closeness, betweenness) to identify and substantiate association factors related to COVID-19 for biological plausibility.

The CORD-ReDrugS project enhanced ReDrugS [14] to use the concepts and relations from extracted entities of the COVID dataset to repurpose potential therapies. A knowledge graph to define a cause-and-effect knowledge model of COVID-19 pathophysiology comprising information encoded in Biological Expression Language (BEL) was made for a selected corpus around COVID-19. Mappings, mainly for viral proteins, were made to the NCBI database [8].

Additional efforts incorporated knowledge extracted from the COVID dataset into already existing knowledge graphs. COVID related research findings were added into the Open Research Knowledge Graph [4]. Covid-19 and associated publications were also made available in the Microsoft Academic Knowledge Graph (MAKG) [9].

3 Creating a Knowledge Graph of The CORD-19 Dataset

To create a knowledge graph from an already existing data source, we have to combine both a transparent architecture to generate the linked data and have a correct idea about the originally used data format. In this section, we first describe the original dataset and underlying data format. Next, we give detailed information about the knowledge graph modeling procedure.

3.1 COVID 2019 Open Research Dataset (CORD-19)

At the beginning of March 2020, Kaggle released the COVID 2019 Open Research Dataset (CORD-19) dataset in collaboration with several research groups, such

[1] https://github. com/RDFLib/rdflib
[2] https://github.com/kingfish777/COVID19

as Microsoft Research. The dataset contains information of over 63,000 papers concerning COVID-19, SARS-CoV-2, or any other related coronaviruses. The papers stem from various sources, such as PubMed Central (PMC) and medRxiv, and are from different research domains. Information about these papers is provided in the form of a CSV file. For more than 51,000 of these papers, a JSON file is provided that contains detailed information about the authors, the content and the other studies that were cited. A schematic overview of such a JSON file is visualized in Figure 1.

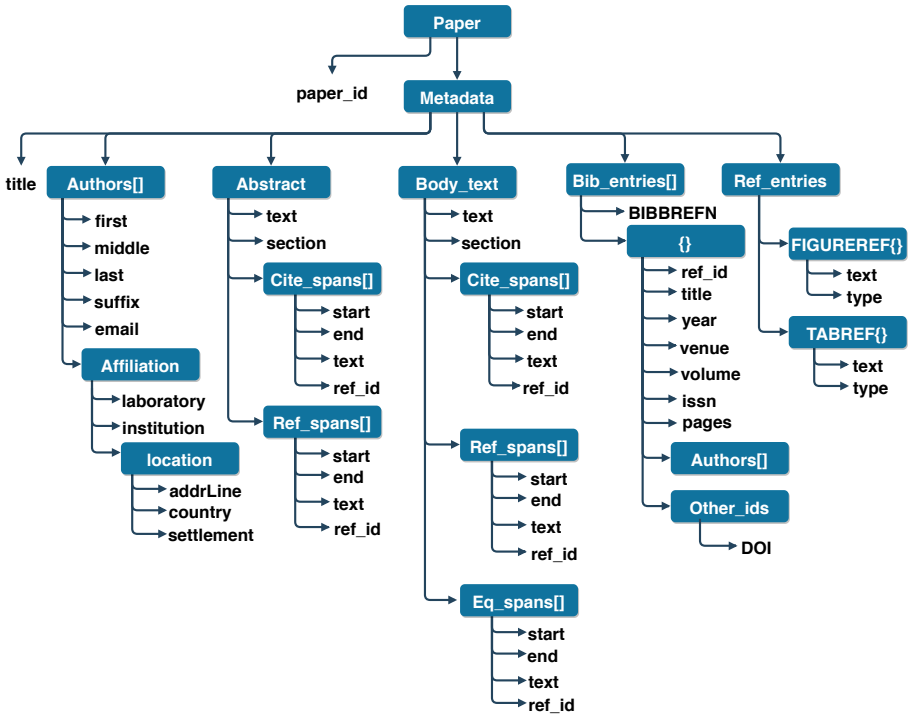


Fig. 1: JSON structure of a given paper. Five main components were nested: abstract, author, body, cites and reference information.

3.2 Knowledge Modeling

In order to facilitate the (meta-)analysis of this significant body of literature, we semantically enriched the data by mapping it to the Resource Description Framework (RDF). Since the data was available in structured formats (JSON and CSV), we can define rules that map chunks of structured information to RDF triples. The RDF Mapping Language (RML) [7] allows intuitively specifying

these rules. We now discuss the two main steps of our conversion procedure, which is visualized in Figure 2.

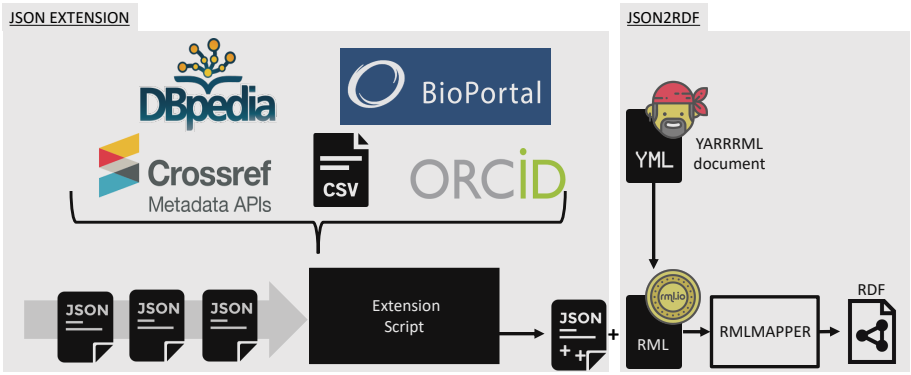


Fig. 2: Overview of each component used to transform the dataset from Section 3.1 into an RDF representation.

Extending the JSON First, each JSON file representing a paper is loaded by a Python script in order to extend it with the information provided by external resources. For example, useful information from the metadata CSV file, such as the journal and DOI, is incorporated into the JSON file. Further on, several modules, interacting with external APIs, use some of the JSON fields displayed in Figure 1 in their request bodies to acquire even more information. The full list of external APIs and the result values used to extend these papers are described in Section 4. When this step is finished, a new extended JSON file for each paper in the original dataset has been made.

Mapping to RDF In a second step, the extended JSON files are converted to RDF. To make this transformation adaptable, transparent and reusable, a mapping document was created that contains rules on how each element in the JSON can be mapped on a corresponding semantic output value. The mapping document was created with YARRRML, a human-readable text-based representation that can be used to represent RDF Mapping Language (RML) rules [12]. A part of the YARRRML document is being displayed in Listing 1.1. Such a YARRRML document usually consists of two main parts: a part listing all the used prefixes (lines 1 till 6) followed by a part describing the actual semantic mapping (lines 7 till 25). Within this mapping section, we describe 1) the source of the input file (e.g., the file path to the JSON-formatted paper at line 10), 2) the subject mapping (here in lines 11 till 15, we took the DOI from the JSON file to create a unique identifier in and 3) all possible predicate-object relations.

In the YARRRML example in Listing 1.1, two such predicate-object relations are defined. One specifying each paper as a `fabio:Work` at line 17, and a second predicate-object relation described by a function which checks whether or not the paper is a `fabio:JournalArticle` (lines 19 till 24). This simple example shows how concepts of fabio (FRBR-aligned Bibliographic Ontology), describing entities that are published or potentially publishable [20], can be mapped using the values of the extended JSON files. The eventually used mapping script outlines a lot more concepts from different domains and ontologies.

As this YARRRML document is only a human-readable text-based representation of RML rules, we have to convert this YARRRML document to an RML document by using the YARRRML Parser[3]. Note that it is possible to write RML rules in this setup directly, but by using YARRRML, we created the ability to let others extend the mapping documents with reduced human effort and without requiring a lot of specific knowledge about semantic web formats.

```
1  prefixes:
2    idlab-fn: "http://example.com/idlab/function/"
3    fabio: "http://purl.org/spar/fabio/"
4    grel: "http://users.ugent.be/~bjdmeest/function/grel.ttl#"
5    ...
6
7  mappings:
8    Realization:
9      sources:
10       - ['tmp/transform_data.json~jsonpath', '$']
11     s:
12       function: grel:array_join
13       parameters:
14         - [grel:p_array_a, "http://dx.doi.org/"]
15         - [grel:p_array_a, "$(doi)"]
16     po:
17       - [a, fabio:Work]
18       - p: a
19         o:
20           - function: idlab-fn:decide
21             parameters:
22               - [idlab-fn:str, $(type)]
23               - [idlab-fn:expectedStr, "journal-article"]
24               - [idlab-fn:result, fabio:JournalArticle]
25   ...
```

Listing 1.1: YAML script to represent the relation between already existing ontological concepts and the JSON values.

The RMLMapper [6] takes both the extended JSON files and the RML document generated using the above YARRRML document as input and produces a set of N-Triples for each paper. Finally, all these N-Triple files were concatenated together to form a single large knowledge graph. A snippet extracted from such an N-Triple file, but represented in a turtle format to improve readability, is provided in Listing 1.2. As defined in the YARRRML subject mapping, all papers are described by a single URI, which is the DOI. All the code used and the input

[3] `https://github.com/rmlio/yarrrml-parser`

```
@prefix doi: <http://dx.doi.org/> .
@prefix fabio: <http://purl.org/spar/fabio/> .
@prefix COVID19: <http://idlab.github.io/covid19#> .
@prefix orcid: <https://orcid.org/> .
@prefix spar: <http://purl.org/spar/>
@prefix foaf: <http://xmlns.com/foaf/0.1>
@prefix dbr: <http://dbpedia.org/resource/>

doi:10.3390/molecules21121629 a fabio:Work.
doi:10.3390/molecules21121629 a fabio:JournalArticle.
doi:10.3390/molecules21121629 COVID19:hasConcept dbr:Amide.
doi:10.3390/molecules21121629 COVID19:hasConcept dbr:3i.
doi:10.3390/molecules21121629 spar:pro/creator orcid:0000−0002−8523−6340.
orcid:0000−0002−8523−6340 a foaf:Person.
orcid:0000−0002−8523−6340 foaf:surname "Jane".
orcid:0000−0002−8523−6340 foaf:firstName "Smith".
doi:10.3390/molecules191219292 spar:cito/isCitedBy doi:10.3390/molecules21121629.
```

Listing 1.2: Turtle representation of the N-TRiple file for http://dx.doi.org/10.3390/molecules21121629 extracted by the RML mapper.

files required in order to perform this conversion are available open-source on Github[4].

3.3 Knowledge graph availability

In order to make the data FAIR, we have set up a Linked Data Fragments (LDF) server with a HDT back-end to expose a Triple Pattern Fragments (TPF) interface at the following URL: `https://query-covid19.linkeddatafragments.org/`. This allows users to query the constructed knowledge graph in a comfortable and scalable fashion. Moreover, the use of TPF guarantees the availability of the resource [25]. We used Comunica [22] to set up the LDF server. Comunica is a query engine platform that offers a plethora of modules for users to design a query engine that fits their needs. The example query which searches for the concept "protein" inside our knowledge graph returns 100 results containing both the DOI identifier and publisher in 12 seconds.

4 External Resources

As shown in Figure 2, in addition to the information available in the provided JSON and CSVs, we link the papers to external resources to enrich our dataset with more information. In this section, we give more details on these external resources, as well which fields of the JSON schema visualized in Figure 1 were used to obtain this additional information.

[4] `https://github.com/GillesVandewiele/COVID-KG`

DBpedia [3] DBpedia resources were linked to several concepts of each paper. On the one hand, the country, institute and research labs of each of the authors were linked to their respective DBpedia resources. Heuristics were used to check if the DBpedia URI exists by concatenating the domain name with the JSON value or comparing the JSON value with the results of DBpedia lookup [21]. On the other hand, DBPedia Spotlight [15] was used to identify general terms in the title, abstract and body of the paper. A new JSON key `hasConcept` was added for each text block, with all the found values in a list. This list indicated all the DBpedia concepts that were detected within that block.

BioPortal [16] The title, abstract and body text were processed with the BioPortal annotation tool in order to identify concepts. This annotator returns annotations, especially for biomedical text with classes from biomedical ontologies. We limited the scope of concepts to the COVID-19 surveillance (COVID19), Coronavirus Infectious Disease (CIDO) and Influenza (FLU) ontologies. Similar to the DBpedia concepts, a `hasConcept` JSON key was added in each text block to list all these newfound concepts.

CrossRef [13] The citation information was often provided in the form of titles and the authors' abbreviated names. In order to link these papers to their respective DOI, the CrossRef API was used. Moreover, the CrossRef API provides additional metadata, such as the authors' full name and the journal in which it was published. These papers were then linked together using the `isCitedBy` or `cites` predicate between two DOIs.

ORCID [11] As the author information both obtained by the CrossRef API and provided in the original dataset was sometimes limited or missing, each of the authors was linked to their respective ORCID identifier, when possible. Using this identifier, the ORCID API was used to provide additional information, such as the institution or lab they are working for, which, in turn, could be linked to their DBpedia resources.

5 Applications

Having the original structured data formats in the form of a Knowledge Graph, with additional knowledge linked from external resources, allows for the creation of applications that were a priori not possible. In this section, we discuss some potential applications that could facilitate the analysis of researchers studying the body of COVID-19 literature. All the examples used in this application section are just chosen for information purposes. They merely illustrate the possible applications that can be used by experts within this field to get more insights within the COVID domain.

5.1 Network analysis

Network analysis is a powerful tool that can reveal interesting patterns hidden in graph datasets. In order to perform such an analysis, we converted our

knowledge graph in a regular directed graph by retaining only citation information. This removes the multi-relational aspect of the graph. The conversion is needed as the current network analysis tools can not deal with different labeled edges. The newly constructed graph consists of nodes that represent the papers and edges between these nodes that represent citations from one paper to the other. Below we describe how network analysis on a graph consisting of information on COVID-19 literature could allow, for example, to find communities of related publications or to identify influential scientific contributions. More detailed examples of network analysis, network visualizations and analysis results highlighting central papers can be found online[5].

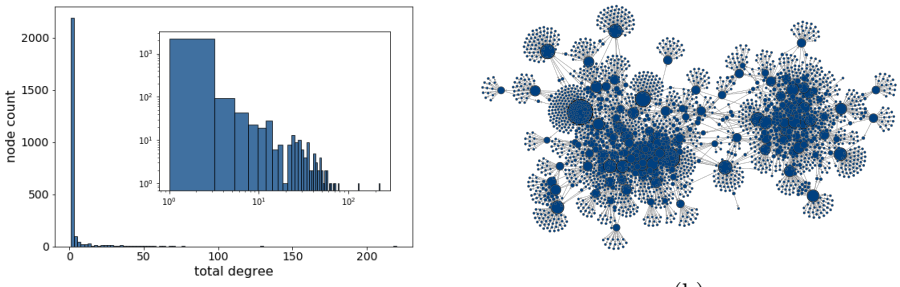
Detecting communities Clustering reveals information on how tightly some groups of publications are interconnected through citations. Such groups often create communities or even cliques, where every paper cites every other paper within that group. The clustering coefficient can be measured on a local and global scale. In the former case, clustering is considered within each node's neighborhood separately, while the latter is an average over the entire network. The article with the highest local clustering is: *Human Bocavirus infection in hospitalized children during winter*. The total global clustering coefficient for our citation network is 0.024009 ($\pm$ 0.007078).

Identifying influential publications In order to identify the most influential publications, we study the node centrality. Centrality can be interpreted as a measure of a node's importance. Hence, we conjecture that an influential paper is highly connected, thus, *central* in one way or another. Several metrics can be used to estimate the centrality of a node.

A first, straightforward, metric uses the node degree as a proxy for centrality. The in-degree indicates the number of articles citing a specific article. The out-degree counts the number of articles cited from a specific article. Fig. 3 shows the distribution of the total degree (i.e., in-degree plus out-degree) of a part of our knowledge graph and a visualization of the corresponding network. In this example, the highest connected paper stands out visibly. It should be noted that these degrees do not correspond to the number of citations or cited articles provided by, for example, Google Scholar, but rather to the number of citations within the body of COVID-19 literature. The article with the highest number of links is: *Human rhinoviruses: the cold wars resume*. PageRank and Hyperlink-Induced Topic Search (HITS) can be seen as more sophisticated variants of these measures. They also evaluate the importance of a node based on the rank of its neighbors.

One other exciting metric is closeness centrality, which can be used to identify influential publications within the network clusters or communities, as defined

[5] `www.kaggle.com/iljara/covid-19-knowledge-graph-a-network-analysis`



(a) (b)

Fig. 3: (a) Distribution of the total degree within the sample network. The inset shows the same data on a double-log scale. (b) Network visualization, the node size is proportionate to the node's total degree.

earlier. Related to the concept of closeness is that of betweenness, which measures how much a certain node acts as a connector or joint. Nodes with high betweenness centrality are often review papers or interdisciplinary works that bridge between several research areas. The article with the highest betweenness centrality in our knowledge graph is: *A novel pancoronavirus RT-PCR assay: frequent detection of human coronavirus NL63 in children hospitalized with respiratory tract infections in Belgium.*

5.2 Embedding concepts

The knowledge graph created from the CORD-19 dataset excels in representing structured data. However, the underlying symbolic nature of this triple-based format usually makes knowledge graphs hard to manipulate and impractical for machine learning systems. To tackle this issue, knowledge graph embeddings have been proposed, where components of a knowledge graph, including entities and relations, are embedded into continuous vector spaces. The most common technique to build such embeddings is RDF2Vec [18]. In this section, we highlight two applications which benefit from creating these embeddings.

Retrieving nearest neighbors When creating embeddings for the paper nodes within our knowledge graph, RDF2Vec ensures that papers that are related or similar have closely related embedded vectors. If we display these vectors in a two-dimensional space, we see that that vectors of similar or related papers are visually near each other. This allows us to search for papers that are close to or highly associated with a given paper of interest. Merely searching the nearest neighbors of a given paper embedding can already be a useful application. For example, from the generated RDF2Vec embeddings, we have searched for the nearest neighbors of the following paper: *SARS-related Virus Predating SARS Out-*

break, Hong Kong (http://dx.doi.org/10.3201/eid1002.030533) and found that based on the embeddings the following papers are closely related:

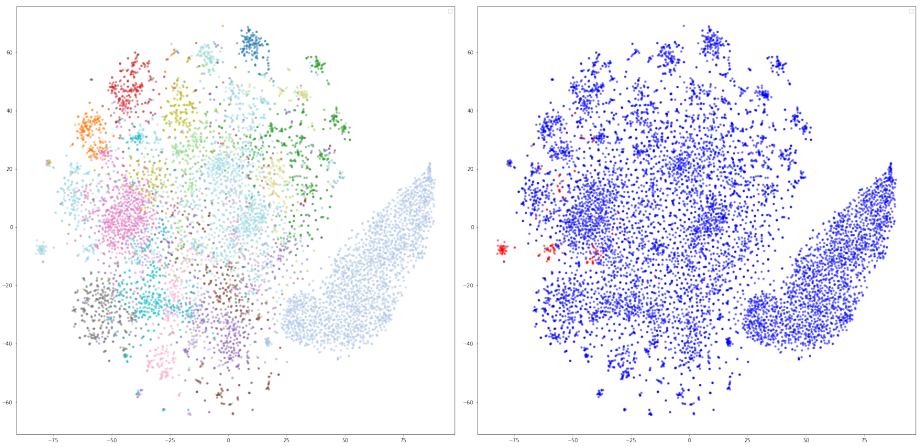
```
Development and Evaluation of Novel Real-Time Reverse Transcription-PCR
Assays with Locked Nucleic Acid Probes Targeting Leader Sequences of
Human-Pathogenic Coronaviruses
(http://dx.doi.org/10.1128/jcm.01224-15)

Crystal structure and mechanistic determinants of SARS coronavirus
nonstructural protein 15 define an endoribonuclease family
(http://dx.doi.org/10.1073/pnas.0601708103)

Antigenic and Immunogenic Characterization of Recombinant Baculovirus-
Expressed Severe Acute Respiratory Syndrome Coronavirus Spike Protein:
Implication for Vaccine Design
(http://dx.doi.org/10.1128/jvi.00083-06)
```

They are all highly correlated due to the performed Sars-Cov experiments.

Advanced clustering By searching for the nearest neighbors of all given papers, clusters can be identified that indicate groups of papers related to each other, e.g., referring or citing one specific paper or by sharing similar defined concepts. Some experiments with k-Means clustering were performed to show this application potential. The results with a predefined k of 20 are visualized in Figure 4(a). Papers closely related to each other concerning their created embedding will have the same cluster label. Clustering is, therefore, a lot more informative because the embeddings take into account the whole neighborhood of the node. In the previous section, the network analysis only considered the citation links between nodes to define possible clusters.



(a) (b)

Fig. 4: (a) Clustering outcome by k-means (k=20) on the generated RDF2Vec embeddings, (b) Example of a selected cluster.

Clustering papers together is beneficial to find groups of related research. However, the embeddings give limited insight into why these papers are grouped together. To increase the understanding of the generated clusters, we experimented with interpretable node classification techniques such as MINDWALC [24]. Concrete labels based on the defined clusters are assigned to the papers to shift the dataset to a binary classification task. In such a task, we try to discriminate the clustered papers from all other papers in our dataset using neighborhood graph walks. Based on the example visualized in Figure 4(b), MINDWALC outputs the walks, which are the most discriminating in classifying the selected cluster concerning all other papers. The output is visualized in Listing 1.3 and shows that for the specified cluster, the DBpedia concepts *Aerosol*, *Airborne_disease* and *Hand_washing* can be found following four links starting from any of the papers within this cluster.

```
{
 ('http://dbpedia.org/resource/Aerosol', 4),
 ('http://dbpedia.org/resource/Airborne_disease', 4),
 ('http://dbpedia.org/resource/Airborne_disease', 6),
 ('http://dbpedia.org/resource/Antiseptic', 6),
 ('http://dbpedia.org/resource/Direct_Contact', 4),
 ('http://dbpedia.org/resource/Engineering_controls', 6),
 ('http://dbpedia.org/resource/Hand_sanitizer', 6),
 ('http://dbpedia.org/resource/Hand_sanitizer', 8),
 ('http://dbpedia.org/resource/Hand_washing', 4),
 ('http://dbpedia.org/resource/Hand_washing', 6),
 ('http://dbpedia.org/resource/Hospital-acquired_infection', 4),
 ('http://dbpedia.org/resource/Huy', 8),
 ('http://dbpedia.org/resource/Hypochlorite', 6),
 ('http://dbpedia.org/resource/Methicillin-resistant_Staphylococcus_aureus', 4),
 ('http://dbpedia.org/resource/Methicillin-resistant_Staphylococcus_aureus', 6),
 ('http://dbpedia.org/resource/Mycobacterium_tuberculosis', 4),
 ('http://dbpedia.org/resource/Nebulizer', 6),
 ('http://dbpedia.org/resource/Nebulizer', 8),
 ('http://dbpedia.org/resource/Particulates', 2),
 ('http://dbpedia.org/resource/Particulates', 4),
 ('http://dbpedia.org/resource/Personal_protective_equipment', 4),
 ('http://dbpedia.org/resource/Personal_protective_equipment', 6),
 ('http://dbpedia.org/resource/Respirator_fit_test', 6),
 ('http://dbpedia.org/resource/Seto_Inland_Sea', 6),
 ('http://dbpedia.org/resource/Transmission_(medicine)', 6),
 ('http://dbpedia.org/resource/Tuberculosis_management', 6)
}
```

Listing 1.3: MINDWALC results for the classification of the nodes in the cluster defined in Figure 4(b).

6 Conclusion

The original CORD-19 dataset delivered a mass of information regarding the COVID-19 pandemic. By transforming the data and available metadata into a knowledge graph, a wide range of useful applications are made possible. The procedure used in this study is generic in such a way that it can be used as a guideline to enrich any structured dataset and transform it into a knowledge graph. New information can be integrated quickly and the whole procedure is transparent as minimal knowledge is required to extend the currently available graph further.

Some potential directions were provided in this paper to show the graph's application potential. Hence, precise research questions must be defined for such

applications as this is an essential condition to have insightful results. The created knowledge graph is only as good as the applications built on top of it. Besides the availability of an endpoint, there is still a need for front-ends that allow non-technical people, which many biomedical researchers are, to interact with this resource and to reveal its connected knowledge.

7 Code and dataset availability

Both the knowledge graph and the code to enhance and transform the original CORD-19 dataset are made available and is summarized on the resource website http://covid-kg.tools:

- The dataset is available on Kaggle and can be reached by the following: `http://doi.org/10.34740/kaggle/dsv/1166450`. Tutorial notebooks on how to interact with the knowledge graph using python, how to generate embeddings and how to apply network analysis are also available under the kernels tab.
- The scripts on how the knowledge graph was constructed can be found on Github: `https://github.com/GillesVandewiele/COVID-KG`
- The TPF interface through which the created knowledge graph can be easily accessed: `https://query-covid19.linkeddatafragments.org/`
- All RML and YARRRML tools are publicly available: `https://rml.io/tools/`
- additionally, the embeddings where generated using pyRDF2Vec, which is available open-source: `https://github.com/IBCNServices/pyRDF2Vec`, as well MINDWALC: `https://github.com/IBCNServices/MINDWALC`

References

1. AI, A.I.F.: Covid-19 open research dataset challenge (cord-19). `https://www.kaggle.com/allen-institute-for-ai/CORD-19-research-challenge`
2. Andersen, K.G., Rambaut, A., Lipkin, W.I., Holmes, E.C., Garry, R.F.: The proximal origin of sars-cov-2. Nature medicine **26**(4), 450–452 (2020)
3. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.: Dbpedia: A nucleus for a web of open data. In: The semantic web, pp. 722–735. Springer (2007)
4. Auer, S., Kovtun, V., Prinz, M., Kasprzik, A., Stocker, M., Vidal, M.E.: Towards a knowledge graph for science. In: Proceedings of the 8th International Conference on Web Intelligence, Mining and Semantics. pp. 1–6 (2018)
5. coronavisrus resource center, J.: Covid-19 dashboard by the center for systems science and engineering (csse) at johns hopkins university (jhu). `https://gisanddata.maps.arcgis.com/apps/opsdashboard/index.html\#/bda7594740fd40299423467b48e9ecf6`
6. Dimou, A., De Meester, B., Heyvaert, P., Verborgh, R., Latré, S., Mannens, E.: RMLMapper: a tool for uniform Linked Data generation from heterogeneous data
7. Dimou, A., Vander Sande, M., Colpaert, P., Verborgh, R., Mannens, E., Van de Walle, R.: RML: a generic language for integrated RDF mappings of heterogeneous data. In: Proceedings of the 7th Workshop on Linked Data on the Web. vol. 1184 (2014)

8. Domingo-Fernandez, D., Baksi, S., Schultz, B., Gadiya, Y., Karki, R., Raschka, T., Ebeling, C., Hofmann-Apitius, M., et al.: Covid-19 knowledge graph: a computable, multi-modal, cause-and-effect knowledge model of covid-19 pathophysiology. BioRxiv (2020)
9. Färber, M.: The microsoft academic knowledge graph: A linked data source with 8 billion triples of scholarly data. In: International Semantic Web Conference. pp. 113–129. Springer (2019)
10. Guoqian Jiang, Harold Solbrig, F.t.: Cord-19-on-fhir – semantics for covid-19 discovery. `https://github.com/fhircat/CORD-19-on-FHIR`
11. Haak, L.L., Fenner, M., Paglione, L., Pentz, E., Ratner, H.: Orcid: a system to uniquely identify researchers. Learned Publishing **25**(4), 259–264 (2012)
12. Heyvaert, P., De Meester, B., Dimou, A., Verborgh, R.: Declarative Rules for Linked Data Generation at Your Fingertips! In: European Semantic Web Conference. pp. 213–217. Springer (2018)
13. Lammey, R.: Crossref text and data mining services. Science Editing (2015)
14. McCusker, J.P., Dumontier, M., Yan, R., He, S., Dordick, J.S., McGuinness, D.L.: Finding melanoma drugs through a probabilistic knowledge graph. PeerJ Computer Science **3**, e106 (2017)
15. Mendes, P.N., Jakob, M., García-Silva, A., Bizer, C.: Dbpedia spotlight: shedding light on the web of documents. In: Proceedings of the 7th international conference on semantic systems. pp. 1–8 (2011)
16. Noy, N.F., Shah, N.H., Whetzel, P.L., Dai, B., Dorf, M., Griffith, N., Jonquet, C., Rubin, D.L., Storey, M.A., Chute, C.G., et al.: Bioportal: ontologies and integrated data resources at the click of a mouse. Nucleic acids research **37** (2009)
17. Preusse, M.: COVID-19 Knowledge Graph. `https://covidgraph.org` (2020)
18. Ristoski, P., Paulheim, H.: Rdf2vec: Rdf graph embeddings for data mining. In: International Semantic Web Conference. pp. 498–514. Springer (2016)
19. Schlichtkrull, M., Kipf, T.N., Bloem, P., Van Den Berg, R., Titov, I., Welling, M.: Modeling relational data with graph convolutional networks. In: European Semantic Web Conference. pp. 593–607. Springer (2018)
20. Shotton, D., Peroni, S.: Fabio, the frbr-aligned bibliographic ontology (2011)
21. Steenwinckel, B., Vandewiele, G., De Turck, F., Ongenae, F.: Csv2kg: Transforming tabular data into semantic knowledge. SemTab, ISWC Challenge (2019)
22. Taelman, R., Van Herwegen, J., Vander Sande, M., Verborgh, R.: Comunica: a modular sparql query engine for the web. In: International Semantic Web Conference. pp. 239–255. Springer (2018)
23. at UPB, D.S.G.: Covid19ds: Rdf file generation is based on papers related to the covid-19 and coronavirus-related research (2020)
24. Vandewiele, G., Steenwinckel, B., Ongenae, F., De Turck, F.: Inducing a decision tree with discriminative paths to classify entities in a knowledge graph. In: SEPDA2019, the 4th International Workshop on Semantics-Powered Data Mining and Analytics. pp. 1–6 (2019)
25. Verborgh, R., Vander Sande, M., Hartig, O., Van Herwegen, J., De Vocht, L., De Meester, B., Haesendonck, G., Colpaert, P.: Triple pattern fragments: a low-cost knowledge graph interface for the web. Journal of Web Semantics **37**, 184–206 (2016)