



**HAL**  
open science

## Tracing and analyzing COVID-19 dissemination using knowledge graphs

Gabriel H.A. Medeiros, Lina Soualmia, Cecilia Zanni-Merk, Ramiz Hagverdiyev

### ► To cite this version:

Gabriel H.A. Medeiros, Lina Soualmia, Cecilia Zanni-Merk, Ramiz Hagverdiyev. Tracing and analyzing COVID-19 dissemination using knowledge graphs. Knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 26th International Conference KES2022, Sep 2022, Verona, Italy. <10.1016/j.procs.2022.09.277>. <hal-03827772>

**HAL Id: hal-03827772**

**<https://hal.science/hal-03827772v1>**

Submitted on 7 Apr 2026

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons CC BY-NC-ND 4.0 - Attribution - Non-commercial use - No Derivative Works - International License



26th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems (KES 2022)

## Tracing and analyzing COVID-19 dissemination using knowledge graphs

Gabriel H. A. Medeiros<sup>a,\*</sup>, Lina F. Soualmia<sup>a</sup>, Cecilia Zanni-Merk<sup>a</sup>, Ramiz Hagverdiyev<sup>a,b</sup>

<sup>a</sup>Normandy University, LITIS EA 4108, Saint-Étienne-du-Rouvray, France

<sup>b</sup>French-Azerbaijani University, Baku, Azerbaijan

### Abstract

The COVID-19 (SARS-CoV-2) spread around the globe could have been halted if we had had a better understanding of the situation and applied more restrictive measures for travel adapted to each country. This is due to a lack of efficient tools to visualize, analyze and control the virus dissemination. In the context of virus proliferation, analyzing flight connections between countries and COVID-19 data seems helpful to understand spatial and temporal information about the virus and its possible spread. To manage these complex, massive, and heterogeneous data, we propose a methodology based on knowledge graphs models. Several analyses and visualization tools can be applied, and our results show that these knowledge graph models may be a promising way to study the dissemination of any virus. These graphs can also be easily enriched with additional information that could be useful in the future to analyze or predict other interesting indicators.

© 2022 The Authors. Published by Elsevier B.V.

This is an open access article under the CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0>)

Peer-review under responsibility of the scientific committee of the 26th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems (KES 2022)

**Keywords:** Knowledge Graphs; COVID-19; Massive Heterogeneous Data Management; Data Visualization.

### 1. Introduction

The International Epidemiology Association's Dictionary of Epidemiology defines a *pandemic* as "an epidemic occurring worldwide" [1], and COVID-19 (SARS-CoV-2) is the most recent pandemic that has spread around the globe. Even if we live in a globalized era, where new information is quickly made available on the Internet, there are not enough restrictive measures, based on each country's situation, against the virus to stop its spread from outbreak to being a pandemic. This is primarily due to the lack of ways to visualize, analyze and control its proliferation, which would also help raise people's awareness of the possible severity of the virus. There are many tools to store and query data, like relational databases, but most lack efficiency in conveying complex and highly linked large-scale data.

\* Corresponding author. Tel.: +33-07-69-99-69-46.

E-mail address: [gabriel-henrique.alencar-medeiros@univ-rouen.fr](mailto:gabriel-henrique.alencar-medeiros@univ-rouen.fr)

There are approaches that propose a methodology to deal with viruses' high heterogeneous linked data for pandemic detection and spread visualization by merging multiple ontologies [2], which are "a formal, explicit specification of a shared conceptualization" [3], and it gives us knowledge about a domain of interest by describing the concepts in the domain and also the relationships between those concepts [4]. The only problem is that those approaches do not handle massive data. To address this issue, we could use a graph-based approach that would be able to manage big data [5].

In this paper, our objective is to address the management of complex data about COVID-19 and flight connections as a graph-based knowledge format, known as a knowledge graph, to better analyze the spread of the virus using different queries and graph algorithms to propose different measures to control its proliferation.

This article is organized as follows. First, the literature review recalls the definitions of knowledge graphs and how it was used in other work against COVID-19. Later, we detail our methodology with some experiments to show the feasibility of our proposal. Finally, we detail our perspectives on future work and conclusions.

## 2. Literature Review

Let  $Con$  be a countably infinite set of constants, a knowledge graph can be mapped as triples of  $(v_1, e, v_2)$ , being  $v_1$  and  $v_2$  nodes ids from the set of nodes ids  $V \subseteq Con$ , and  $e$  an edge id from the set of edge ids  $E \subseteq Con$ , both nodes representing entities of a domain, and the edge representing the relationship between these entities. Different standard modelings can represent those graphs, each conveying the same information but handling the data differently. As we need to work with heterogeneous massive complex data from different sources, one clean and intuitive representation is needed, so we chose to use property graph models [6]. Let  $G$  be a knowledge graph, a property graph is a tuple:

$$G = (V, E, L, P, U, e, l, p) \quad (1)$$

where  $L \subseteq Con$  is a set of labels,  $P \subseteq Con$  is a set of properties/attributes that each node or edge could have,  $U \subseteq Con$  is a set of values of their respective properties,  $e : E \rightarrow V \times V$  maps an edge id to a pair of node ids,  $l : V \cup E \rightarrow 2^L$  maps a node or edge id to a set of labels, and  $p : V \cup E \rightarrow 2^{P \times U}$  maps a node or edge id to a set of property-value pairs, which allow us to avoid creating extra nodes and edges, as we put data directly inside the entities and relationships, so this modeling makes knowledge graphs more concise.

Knowledge graphs existed for a long time, and many works expanded and analyzed the theory as [7, 8, 9]. However, since Google made them popular [10], we can see the growing number of their applications in different domains. As for virus tracking, there are relevant works for COVID-19 data analysis, like patient information tracking [11], by using knowledge graphs to manage heterogeneous data about contact cases of patients that traveled inside their country, and by using graph algorithms to extract information about the propagation of the virus.

The authors in [12] study the field of social networks, where users can be ranked according to diffusion information using centrality algorithms. These algorithms are used to rank nodes in a knowledge graph according to their connections in the graph. Other examples from this same survey are given in other areas, such as biological networks, citation networks, and urban street networks. The application of these different algorithms is possible by simply changing the diffusion information in the network to the factor that we want to investigate. If we consider the number of COVID-19 cases as the diffusion information, we could observe how the virus spread around the network of countries. Therefore, the only remaining task would be to select graph algorithms that are, in fact, relevant to our work. The chosen algorithms and their usefulness are described in subsection 3.2.

Knowledge graphs can also convey spatial and temporal information that can be exploited by means of spatial and temporal reasoning [13]. In particular, the edges may contain temporal context annotations [14, 15] that can be exploited with *Allen's Logic* [16].

### 3. Methodology

As already expressed, the objective of the research is to manage heterogeneous massive complex data by using knowledge graphs to create a consistent model about COVID-19 focusing on the flight connections among countries. From that graph, we would be able to extract information to monitor the spread of the virus, leading to eventually raising alarms in the case of suspicious configurations being detected.

We will focus on open data about flight connections and epidemiological data (number of cases, deaths, ...) during the world's first wave of COVID-19 (in January 2020). These data will be pre-processed to assure coherence among them. Later, the knowledge graph will be built regarding the structure presented in subsection 3.1 in order to be analyzed according to the algorithms described in subsection 3.2 to obtain pertinent information.

#### 3.1. Knowledge Graph Construction

We show here which nodes and edges can be used to represent the COVID-19 data. It is not the only modeling we can have, but it can represent enough concepts/classes of elements to track viruses' cases, regional information, and flight information. For the entities/nodes, we have:

- Country: Holds the unique Wikipedia link identifier and other descriptive information, such as latitude, longitude, ISO2, and ISO3 codes;
- City: Holds the unique Wikipedia link identifier and the Wikipedia identifier of the respective country;
- Airport: Holds the ICAO unique identifier, its latitude, and longitude;
- Event: Holds diverse information about COVID-19 (such as the number of cases, deaths, vaccinated people, ...) for a given date in a certain country.

For the edges/relationships between nodes, we have:

- CITY\_OF: connects a city to its respective country;
- AIRPORT\_OF: connects an airport to its respective city;
- EVENT\_OF: connects an event to its respective country and holds the date in which the event took place;
- FLIGHT: describes a flight that connects two airports, with the date and other useful information. It can be either international or domestic.

#### 3.2. Graph Exploitation

Once the knowledge graph is built, we can extract, infer, and predict different pieces of information using graph algorithms. These algorithms are introduced in this section and will be experimented in this paper.

We can analyze the spread of the virus in two possible ways, either we consider all the flights in the graph (regardless of the fact that the countries may have zero COVID-19 cases), or we consider only the flights leaving from countries that can effectively spread the virus because they already have some cases.

In subsection 3.2.1, we show the proposed centrality algorithms to extract information about our knowledge graph, and in subsection 3.2.2 we will also propose another type of graph algorithm that could help us analyze our graph.

##### 3.2.1. Centrality Algorithms

The selected algorithms and the justification for this choice are the following:

*Degree Centrality.* This algorithm is used to rank nodes by the number of neighbor nodes (equation 2) [12].

$$C_D(u) = \frac{k_u}{(n - 1)} \quad (2)$$

where  $u$  is the node,  $k_u$  the degree (of edges leaving, entering, or both), and  $n$  the number of nodes. There can be multiple edges between two given nodes, implying that the degree centrality can be greater than one. This formula can also be modified in order to have a version of this algorithm that includes edge weights by replacing  $k_u$  with the sum of the edge weights between  $u$  and its neighbors.

We can use this algorithm to rank the countries according to the number of flights to other countries, so we would be able to know countries that could spread the virus to most other countries. We could also use this algorithm to rank countries according to the number of flights from the first day they have a virus case, so we would be able to know infected countries that could spread the virus to most other countries.

*Betweenness Centrality.* Betweenness centrality measures centrality in a graph based on shortest paths. For every node pair in a connected graph, there exists at least one shortest path between the nodes such that either the number of edges that the path passes through (for unweighted graphs) or the sum of the weights of the edges (for weighted graphs) is minimized. The betweenness centrality for each node is the number of these shortest paths that pass through the node (also called a bridge) [17]. We can use this algorithm to rank nodes as bridges (equation 3) [12].

$$C_B(u) = \frac{\sum_{s \neq u \neq t} \frac{\partial_{st}(u)}{\partial_{st}}}{(n-1)(n-2)/2} \quad (3)$$

being  $\partial_{st}(u)$  the number of shortest paths between nodes  $s$  and  $t$  with node  $u$  acting as an intermediate node in the shortest path, and  $\partial_{st}$  all shortest paths between  $t$  and  $s$ . There can be multiple edges between two given nodes, implying that the betweenness centrality can be greater than one. This formula can be modified to have a version of this algorithm that includes edge weights by considering the weights in each path, as in the Dijkstra's algorithm [18].

We could use it to analyze possible bridges among groups of countries in order to know where to apply restrictive measures to block the flow of the virus among those groups.

*Harmonic Centrality.* This algorithm gives the ranking of nodes with the shortest paths to all other nodes considering unconnected graphs (equation 4), as explained in [19].

$$C_H(x_i) = \frac{1}{n-1} \sum_{j \neq i} \frac{1}{\text{dist}(x_i, x_j)} \quad (4)$$

where  $\text{dist}$  is the distance between two nodes, which can consider the weights of each edge in its calculation.

This algorithm can rank countries whose infected people could reach the other countries the fastest. The spread of the virus could be slowed down by restricting travel from the most relevant countries classified by this ranking.

*Eigenvector Centrality.* It assigns a relative index value to all nodes in the network based on the concept that connections with high indexed nodes contribute more to the score of the node than the connections with low indexed nodes, as explained in [12]. For a graph  $G(V, E)$ , the eigenvector centrality is calculated with equation 5.

$$C_E(u) = \frac{1}{\Lambda} \sum_{v \in V} A_{uv} C_E(v) \quad (5)$$

where  $v$  is a neighbor of  $u$ ,  $\Lambda$  is a constant,  $A$  is the adjacent matrix of  $uv$ , and  $V$  is the set of neighbors of  $u$ .

This algorithm can rank nodes connected to other nodes with the highest number of cases and deaths. It means that it can rank an uninfected country's susceptibility to having cases of the same viruses. The governments of the most susceptible countries should have the most attention to applying restrictive measures for arriving travelers.

### 3.2.2. Community Detection

We also can analyze the knowledge graph by using graph algorithms to find communities of countries, being a community a group of countries that are disjoint from other groups. We can identify connected subgraphs disjoint from each other by using the "Strongly Connected Components" algorithm [20]. We can use it to see disjoint communities of countries during a period of time, so the virus from one community would not reach other communities in the same period. We would also be able to identify the counties without any cases of a virus that could have future cases because they were part of a community with at least one country with cases of the virus.

## 4. Experimentation & Results

In this section, we experiment with the datasets explained in subsection 4.1, as they are managed to achieve the resulting knowledge graph, followed by the application of graph algorithms described before to extract relevant information about the spread of the virus.

### 4.1. The Data

We choose to use the following two datasets:

- Crowdsourced air traffic data from The OpenSky Network [21]: It is a dataset with flights between different countries from January 2019 to December 2021;
- Health World Organization dataset about COVID-19 [22]: It holds temporal data on COVID-19's number of cases, vaccinated people and deaths for each country, starting from January 2020 up to date.

In Figure 1, it is possible to see a timeline of events related to COVID-19. We chose to work only with data from January 2020 because it was the month that the virus spread from two countries to another twenty-one, according to the considered datasets. Therefore, we would be able to see the first spread of the virus around the globe and how it could impact the graph algorithms described early.

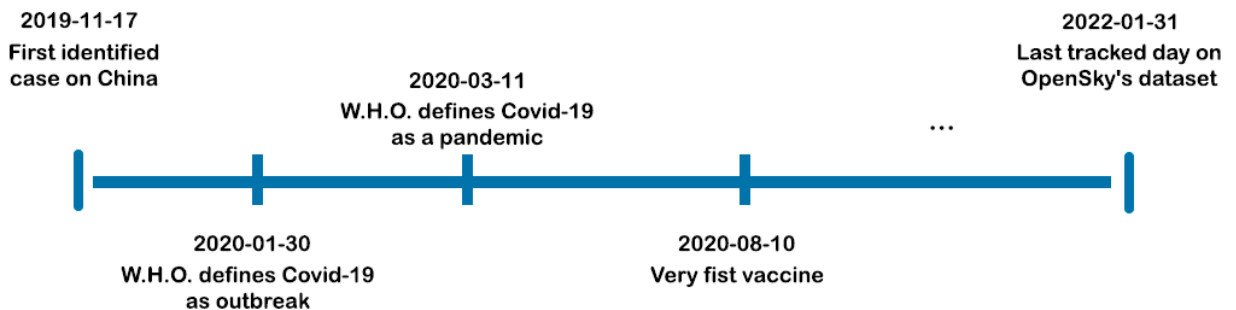


Fig. 1. Timeline of events about the spread of COVID-19.

### 4.2. The Pipeline

For this study, we will be working with the pipeline from Figure 2, which shows us how to get data from different datasets to construct the graph and extract information from it. The left cylinder is the database that contains data about

travels between different countries, and the right one contains information about COVID-19 cases around the globe. Each rectangular node is a sub-package that will handle the data by doing what their names describe, going from the collection, cleaning, and storing to creating the knowledge graph with instances of nodes of different classes. We also have a sub-package to send the graph to a knowledge graph tool. Finally, the edges represent the command sent between the nodes. The knowledge graph created by the "Graph Construction" node will be explored in subsection 4.3, while the "Graph Algorithm Applier" applies the algorithms explained in subsection 3.2 on the graph.

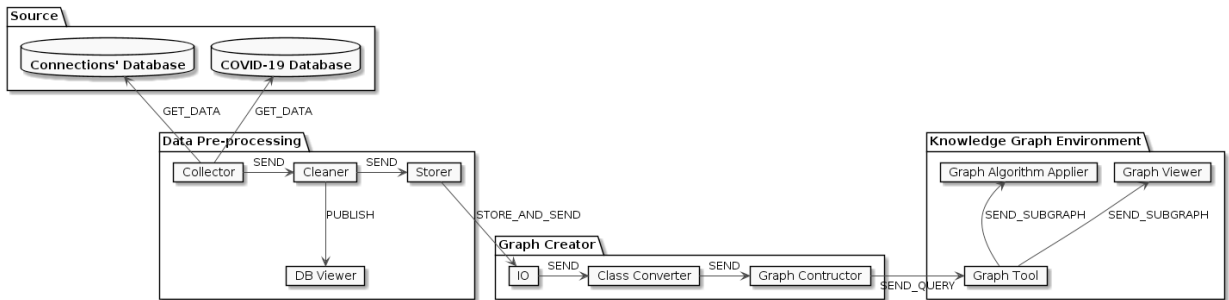


Fig. 2. Pipeline to extract information from sources and create the knowledge graph.

As for the implementation, the sub-packages from the "Data Pre-Processing" and "Graph Creator" packages were made with Python. As for the "Knowledge Graph Environment" package, we have chosen the Neo4j graph database to integrate our graph. It is one of the world's leading open-source NoSQL (Not only SQL) Graph Database Management Systems [23] (GDBMS)<sup>1</sup>. More information about the implementation can be found on (<https://github.com/Gabriel382/COVID19-tracking-knowledge-graph>).

### 4.3. Knowledge Graph

Figure 3 shows a subset of the created knowledge graph from the aforementioned data, focusing on the US node and some of its relationships. The number of nodes per class and edges per relationship associated with the whole knowledge graph are presented in Tables 1 and 2. The data and code can be found in the github repository mentioned above.

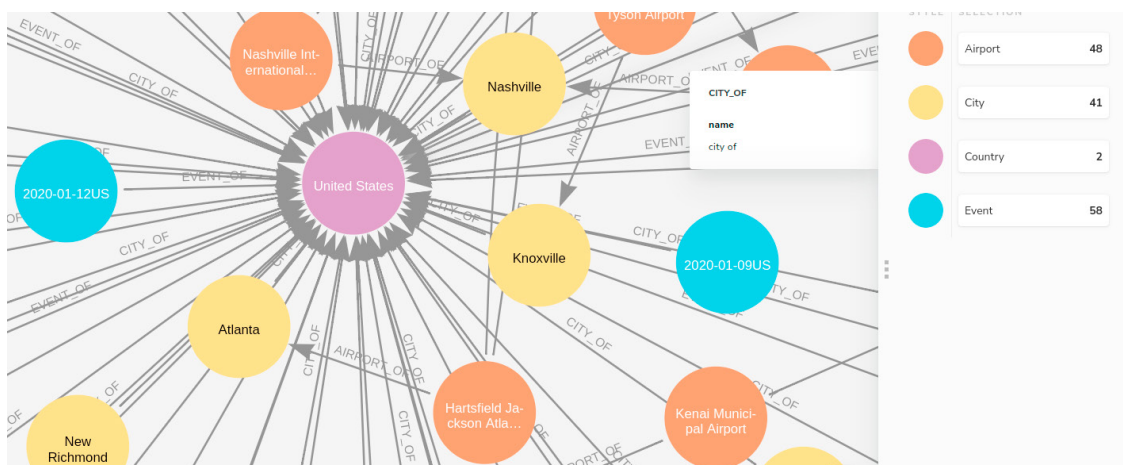


Fig. 3. Subset of our knowledge graph.

<sup>1</sup> <https://db-engines.com/en/ranking>

Table 1. Number of nodes in each class.

Class	Number of nodes
Country	244
City	33,291
Events	180,861
Airports	65,730

Table 2. Number of edges of each relationship type.

Relationship type	Number of edges
CITY_OF	33,291
AIRPORT_OF	65,730
EVENT_OF	180,861
FLIGHT	2,583,438

#### 4.4. Graph Algorithm Results

The knowledge graph from the last subsection makes it possible to extract information by either simple queries or by applying graph algorithms. For instance, by queries, we know that in January of 2020, the virus started in China and Finland and spread to other twenty-one countries over the month. All those countries and other eighty-two countries are part of a community extracted with the "Strongly Connected Components" algorithm. This means that these eighty-two countries risk having new cases of COVID-19 soon, which was indeed the case.

The centrality algorithms operate on subgraphs composed of only one class of nodes and one type of relationship. As we want to analyze the spread of the virus among country nodes, we create the following relationships:

1. EXTERIOR\_FLIGHT: edge created between two countries for each FLIGHT edge between airports from the considered pair of countries. This edge has a property called "from\_infected", whose value is equal to 1 if the country from where the plane takes off has at least one case of the virus and equal to 0 otherwise;
2. INFECTED\_EXTERIOR\_FLIGHT: edge created between two countries for each existent EXTERIOR\_FLIGHT edge with property "from\_infected" with value 1.

We run the centrality algorithms analyzing the EXTERIOR\_FLIGHT edge to know which countries could efficiently spread the virus, regardless of whether they have it. We also run the algorithms on EXTERIOR\_FLIGHT (using the "from\_infected" property as weight) or INFECTED\_EXTERIOR\_FLIGHT (when the algorithm does not use weights) to know which infected countries could effectively spread the virus, as they already have cases. Both results will be shown in two charts having the names of countries on the ordinate axis, while the abscissa is the value of the result of the algorithm.

For this article, we focus on analyzing the virus's global spread. However, it would also be possible to study the spread between cities in the same country by analyzing domestic flights.

It is also essential to remember that even if the algorithms show that a country could have helped disseminate the virus, it does not mean it happened. Nevertheless, the following results are indicators that can be taken into account by each country's government to establish appropriate countermeasures against the virus.

##### 4.4.1. Degree Centrality

We can analyze the results of the unweighted and weighted versions of the algorithm in Figure 4. Even if the United Kingdom has the most outbound flights during the considered period of time, the United States is the country that has

the highest number of outbound travels since the day of the first confirmed case. By focusing on more restrictive measures for travelers in those countries, the spread of the virus could have been reduced.

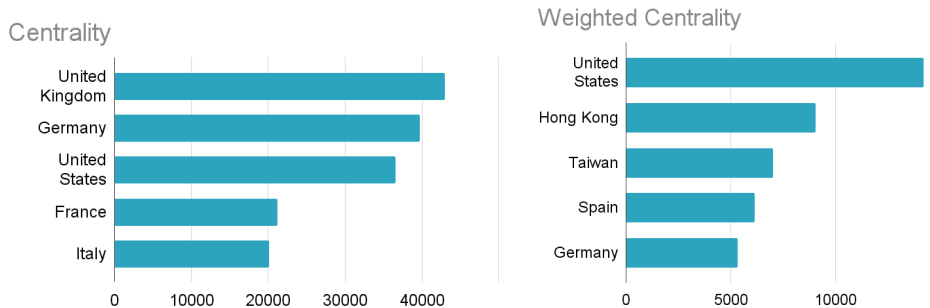


Fig. 4. At the left, the centrality considers the EXTERIOR\_FLIGHT relationship without weight, and at the right, considering the "from\_infected" as the weight.

#### 4.4.2. Betweenness Centrality

From the results in Figure 5, we can observe that the United States is ranked as the most relevant bridge between different communities of countries. Unfortunately, the virus is already present in the US during the considered period of time, meaning that the country contributed strongly to the spreading of COVID-19 since the very beginning of the pandemic.

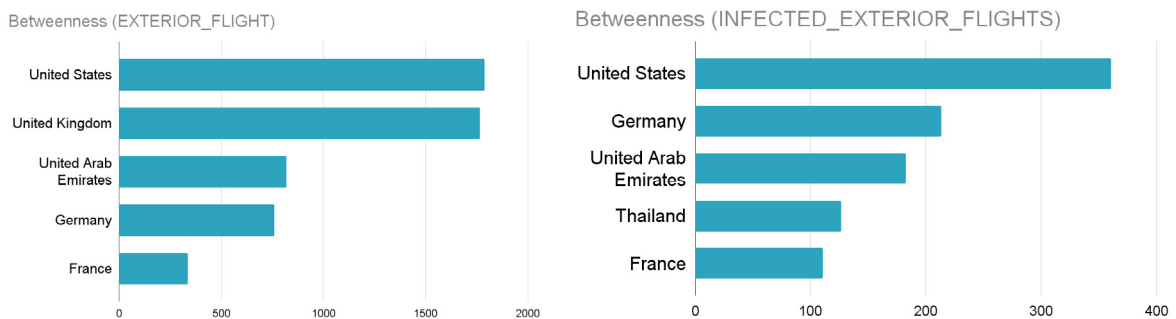


Fig. 5. On the left, the betweenness only considers the EXTERIOR\_FLIGHT relationships without weight, and on the right, it only considers the INFECTED\_EXTERIOR\_FLIGHT relationships.

#### 4.4.3. Harmonic Centrality and Eigenvector Centrality

The Harmonic Centrality shows that Germany is the country with the shortest path to all other countries, which means that it has direct flights to most of them, so it is also a country that should have restrictive measures for travelers. The Eigenvector Centrality gives us the information that the United States is connected to countries with many cases, which is dangerous as it is also the best bridge for the virus during the considered period of time.

## 5. Conclusion

In this paper, we have presented a methodology to manage massive heterogeneous complex data about the proliferation of COVID-19 using knowledge graph modeling by merging data from different trustworthy sources using simple identifiers. This knowledge graph can easily receive more data from the same and different sources by using

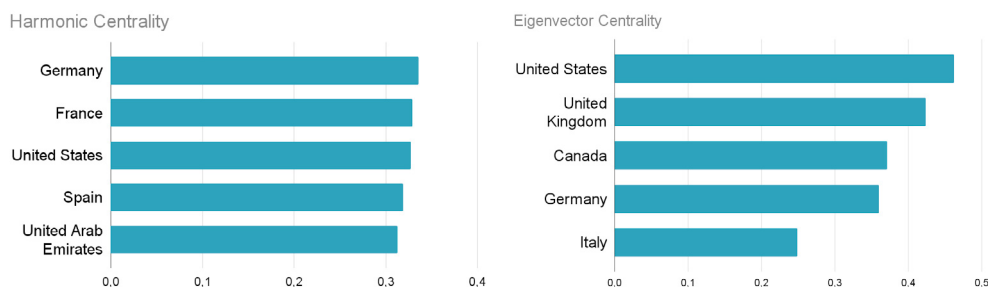


Fig. 6. At the left is the Harmonic Centrality, and at the right is the Eigenvector Centrality, both applied over the EXTERIOR\_FLIGHT relationship with "from\_infected" as weight.

the modeling and node identifiers shown above, and it can also be easily modified to fit into other types of modeling. This gives data scientists a database that conveys more information by visualization than relational databases. It gives those who work in the medical field a chance to analyze the spread of the virus through certain periods of time. The knowledge graph is also a tool to see the evolution of the proliferation as new data is added to it, so it can be used to make people aware of the severity of the virus.

We proposed to use queries and graph algorithms to extract useful pieces of information from the knowledge graph. The centrality algorithms can be used to find countries that could effectively diffuse the virus to other countries around the globe due to a certain period of time. The same algorithms can also be used to visualize how countries with cases of viruses can influence the dissemination to other countries. We can also use community detection algorithms to find communities of countries of interest in our knowledge graph, such as the "Strongly Connected Components" algorithm to find clusters of disjoint communities of countries to identify counties that could have cases in the future.

In the future, we will apply spatial and temporal reasoning to further analyze the spread of the virus according to the geographic position of the countries and the temporal aspect of the dissemination (like flight date and increase of the number of cases on another date). There is also the possibility of trying to predict future cases of the virus in countries that never had it before by trying to predict an edge between an event and a country. We can achieve this by using a group of methods called Link Prediction [24]. These methods tend to have more accurate predictions as the data increases. It would be interesting to experiment with Link Prediction utilizing more months of COVID-19 data.

Many Link Prediction methods can only be applied to floating number vector representations of knowledge graphs, also known as embeddings. There are many embedding algorithms, but they can be separated into two categories: homogeneous embeddings that consider only one class and only one relationship [24] and heterogeneous embeddings that consider multiple classes and relationships [25]. In future works, it would also be interesting to analyze the results of these different embeddings in our knowledge graph and the impact of both on the accuracy of the Link Prediction methods.

Finally, this research is part of a framework for a larger research project. In this project, we propose using knowledge graph models to try to detect the emergence of new health risks by means of social media, biomedical literature, and news media mining. Similar work was done in the field of social politics [26]. However, our methodology considers biomedical data and can extract viruses' information directly from texts from different sources. This approach could give us data about possible new viruses, and by using the methodology shown in this paper, we would be able to manage this data in order to analyze useful indicators about the spread of emergent viruses.

## References

- [1] Singer, Benjamin J., Robin N. Thompson, et Michael B. Bonsall. "The effect of the definition of 'pandemic' on quantitative assessments of infectious disease outbreak risk". *Scientific Reports* 11, number 1 (28 janvier 2021): 2547. <https://doi.org/10.1038/s41598-021-81814-3>.
- [2] Babcock, Shane, et al. "The Infectious Disease Ontology in the age of COVID-19". *Journal of Biomedical Semantics*, vol. 12, no 1, juillet 2021, p. 13. *BioMed Central*, <https://doi.org/10.1186/s13326-021-00245-1>.
- [3] R. Studer, R. Benjamins, and D. Fensel. *Knowledge engineering: Principles and methods*. *Data & Knowledge Engineering*, 25(1–2):161–198, 1998.

- [4] Horridge, M., Knublauch, H., Rector, A., Stevens, R. & Wroe, C. (2004). *A Practical Guide To Building OWL Ontologies With The Protege-OWL Plugin* University of Manchester.
- [5] Janev, Valentina & Graux, Damien & Jabeen, Hajira & Sallinger, Emanuel. (2020). *Knowledge Graphs and Big Data Processing*. 10.1007/978-3-030-53199-7.
- [6] Hogan, Aidan, et al. "Knowledge Graphs". *Synthesis Lectures on Data, Semantics, and Knowledge*, vol. 12, no 2, novembre 2021, p. 1-257. DOI.org (Crossref), <https://doi.org/10.2200/S01125ED1V01Y202109DSK022>.
- [7] Xiaojun Chen, Shengbin Jia, Yang Xiang, A review: Knowledge reasoning over knowledge graph, *Expert Systems with Applications*, Volume 141, 2020, 112948, ISSN 0957-4174, <https://doi.org/10.1016/j.eswa.2019.112948>.
- [8] Ehrlinger, L., & Wöß, W. (2016). Towards a definition of knowledge graphs. *SEMANTiCS (Posters, Demos, SuCCESS)*, 48(1-4), 2.
- [9] HOGAN, Aidan, BLOMQUIST, Eva, COCHEZ, Michael, et al. *Knowledge graphs*. *Synthesis Lectures on Data, Semantics, and Knowledge*, 2021, vol. 12, no 2, p. 1-257.
- [10] Singhal, Amit (May 16, 2012). "Introducing the Knowledge Graph: Things, Not String". *Google Official Blog*. Retrieved September 6, 2014.
- [11] Wu J. Construct a Knowledge Graph for China Coronavirus (COVID-19) Patient Information Tracking. *Risk Manag Healthc Policy*. 2021;14:4321-4337 <https://doi.org/10.2147/RMHPS309732>
- [12] Saxena, Akрати & Iyengar, Sudarshan. (2020). *Centrality Measures in Complex Networks: A Survey*.
- [13] Gérard, Ligozat & Mitra, Debasis & Condotta, Jean-François. (2004). *Spatial and temporal reasoning: Beyond Allen's calculus*. *AI Commun.* 17. 223-233.
- [14] John McCarthy. 1993. *Notes on Formalizing Context*. In *Proceedings of the 13th International Joint Conference on Artificial Intelligence*. Chambéry, France, August 28 - September 3, 1993, Ruzena Bajcsy (Ed.). Morgan Kaufmann, 555–562
- [15] C. Gutierrez, C. A. Hurtado and A. Vaisman, "Introducing Time into RDF," in *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, no. 2, pp. 207-218, Feb. 2007, doi: 10.1109/TKDE.2007.34.
- [16] James F. Allen.: *Maintaining knowledge about temporal intervals*. In *Readings in qualitative reasoning about physical systems*, pp. 361-372. Elsevier (1990)
- [17] [https://en.wikipedia.org/wiki/Betweenness\\_centrality](https://en.wikipedia.org/wiki/Betweenness_centrality), last accessed on April 19, 2022
- [18] Javaid, Adeel. (2013). *Understanding Dijkstra Algorithm*. SSRN Electronic Journal. 10.2139/ssrn.2340905.
- [19] Roachat, Y. (2009). *Closeness Centrality Extended to Unconnected Graphs: the Harmonic Centrality Index*.
- [20] Vincent Levorato, Coralie Petermann. *Detection of Communities in Directed Networks based on Strongly p-Connected Components*. (CASoN), Oct 2011, Salamanca, Spain. pp.211-216. hal- 00634309v2
- [21] Xavier Olive, Martin Strohmeier, & Jannis Lübke. (2022). *Crowdsourced air traffic data from The OpenSky Network 2020 [CC-BY] (v21.12) [Data set]*. Zenodo. <https://doi.org/10.5281/zenodo.5815448>
- [22] WHO COVID-19 Dashboard. Geneva: World Health Organization, 2020. Available online: <https://COVID19.who.int/> (last cited: 2022-03-22).
- [23] Pokorný, Jaroslav. "Graph Databases: Their Power and Limitations". *Computer Information Systems and Industrial Management*, édité par Khalid Saeed et Wladyslaw Homenda, vol. 9339, Springer International Publishing, 2015, p. 58-69. DOI.org (Crossref), [https://doi.org/10.1007/978-3-319-24369-6\\_5](https://doi.org/10.1007/978-3-319-24369-6_5).
- [24] Wang, Meihong, et al. "A Survey on Knowledge Graph Embeddings for Link Prediction". *Symmetry*, vol. 13, no 3, mars 2021, p. 485. DOI.org (Crossref), <https://doi.org/10.3390/sym13030485>.
- [25] Xiao Wang, Deyu Bo, Chuan Shi, Shaohua Fan, Yanfang Ye, Philip S. Yu: *A Survey on Heterogeneous Graph Embedding: Methods, Techniques, Applications and Sources*. *CoRR abs/2011.14867* (2020)
- [26] Abu-Salih, Bilal & Al-Tawil, Marwan & Aljarah, Ibrahim & Faris, Hossam & Wongthongtham, Pornpit. (2020). *Relational Learning Analysis of Social Politics using Knowledge Graph Embedding*.