Interactive Domain-Specific Knowledge Graphs from Text: A Covid-19 Implementation

Vinícius Melquíades de Sousa(✉) and Vinícius Medina Kern

Universidade Federal de Santa Catarina, Florianópolis, Brazil
v.m.kern@ufsc.br

Abstract. Information creation runs at a higher rate than information assimilation, creating an information gap for domain specialists that usual information frameworks such as search engines are unable to bridge. Knowledge graphs have been used to summarize large amounts of textual data, therefore facilitating information retrieval, but they require programming and machine learning skills not usually available to domains specialists. To bridge this gap, this work proposes a framework, KG4All (Knowledge Graphs for All), to allow for domain specialists to build and interact with a knowledge graph created from their own chosen corpus. In order to build the knowledge graph, a transition-based system model is used to extract and link medical entities, with tokens represented as embeddings from the prefix, suffix, shape and lemmatized features of individual words. We used abstracts from the COVID-19 Open Research Dataset Challenge (CORD-19) as corpus to test the framework. The results include an online prototype and correspondent source code. Preliminary results show that it is possible to automate the extraction of entity relations from medical text and to build an interactive user knowledge graph without programming background.

Keywords: Knowledge graphs · COVID-19 · Information retrieval software · Natural language processing · Personalized analytics

1 Introduction

Shannon's Mathematical Theory of Communication [19] is understood as the Information Science debut [4]. Ever since Shannon's work the field has evolved into a number of sub-fields, following the advances in society. One of such fields is Information Retrieval, which was considered to be the Information Science main core [17]. It started in the 1970's and its focus was on the creation of retrieval indexes and the physical allocation of information. As the technological development took place the focus shifted towards information processing and

Funded by CNPq, Research Productivity Grant 314140/2018-2.

E. Bisset Álvarez (Ed.): DIONE 2021, LNICST 378, pp. 240–253, 2021.
https://doi.org/10.1007/978-3-030-77417-2_18

efficient information retrieval, digitally speaking [5]. The use of knowledge graphs to represent human knowledge, and therefore as a way into information retrieval, has been receiving attention both from academia and industry. A knowledge graph can be defined as a structured representations of facts, in the form of entities and relations and its semantic description [10]. A knowledge graph is composed by triplets in the form (head entity, relation, tail entity), Fig. 1 depicts an example of a knowledge graph where on the left-side it is presented the triplets and on the right-side the representation in a graph form.

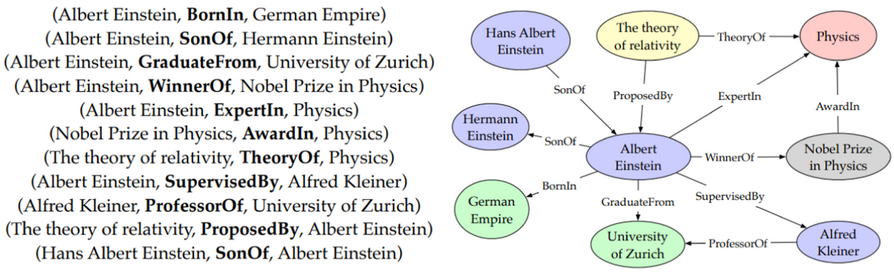


Fig. 1. Example of knowledge graph. Extracted from [10].

The construction of knowledge graphs can be classified into two main groups: (i) manually/curated or (ii) automatic/semi-automatic. The first group consists of allocating domain specialists to annotate, in accordance with a set of rules, the entities, relations and descriptions [22]. Manually constructed knowledge graphs are time consuming and tends to advance at a slower pace than information development. On the other hand, automatic/semi-automatic knowledge graphs are built upon a workflow, usually starting from a text corpus, from which entities and relations are inferred. Automatic/semi-automatic constructed knowledge graphs are able to keep up with the information creation, at the cost of (i) quality, that is, the entities and relations are not as accurate as when the knowledge graph is manually annotated [9] and (ii) having to deal with engineering challenges, such as data acquisition and storage, text parsing, information extraction, etc. While companies such as Google and Microsoft have the necessary resources to solve these challenges, smaller organizations and independent researchers are required to have programming skills in order to be able to use the advances of research in the information retrieval through knowledge graphs [18]. In other terms, the use of machine learning in information retrieval through knowledge graphs results in an increase on the complexity demanded to make use of such advances. The higher the complexity, the more limited is the number of people capable of making use of the gains allowed by those advances [8,14].

The information accessibility and availability for possible users is one of the tasks that Information Science is responsible for [15], as the general view of the information process, from creation to utilization, is a core activity of the area [2]. Domain specialists is a particular group of users, with real needs, that could benefit from using knowledge graphs. They are not usually proficient in programming/machine learning optimization skills and, at the same time, their information needs are not fulfilled by regular knowledge frameworks, such as google [11]. Therefore, if: (i) domain specialists cannot assimilate, through human cognition, the information in the same pace that the information is created [9]; (ii) regular knowledge frameworks are not sufficient to fulfill the domain specialists information needs; and (iii) domain specialists do not have the technical skills in order to make use of algorithms that would allow them to process and interact with a large amount of information. Then, it can be stated that a framework that allowed domain specialists to create and interact with their own knowledge graphs without requiring programming skills would be a step towards narrowing the information creation and assimilation gap. The present work depicts the preliminary results towards a framework that aims to assert the previously stated problem. In other words, it is presented the preliminary work of a framework that aims to allow domain specialist to make use of the advantages of Knowledge Graphs research by creating its own knowledge graph.

The work is organized as follows. Section 2 presents the used methods in order to achieve the results shown in Sect. 3. A discussion about the results is found in Sect. 4 and, finally, Sect. 5 concludes the present work.

2 Methods

This section presents the methods that were used in order to create the presented results. The Subsect. 2.1 depicts the search result for similar works, followed by the Subsect. 2.2 that presents the general overview of the proposed framework. Subsect. 2.3 explains the NLP technique that was used to build the knowledge graph. And finally, Subsect. 2.4 is responsible for justifying the use of network visualization.

2.1 Similar Works

In order to execute a search for similar works at least three search parameters have to be defined: (i) Scientific Bases; (ii) Keywords; and (iii) inclusion and exclusion criteria. Such definitions are as follows. (i) Searched Scientific Bases are: Web of Science, Scopus, IEEE Xplore and Association for Computing Machinery Digital Library (ACM). (ii) Chosen keywords: Knowledge Graph, text OR corpus and Graphical Interface OR Web Application. (iii) The inclusion criteria is listed:

1. Present a framework to build a knowledge graph from text corpus;
2. Present a form of interacting with the knowledge graph;
3. Make the source code or the framework available for use.

And, finally, the exclusion criteria:

1. Not Present a framework to build a knowledge graph from text corpus;

2. Not Present a form of interacting with the knowledge graph;
3. Not Make the source code or the framework available for use;
4. Not being an scientific paper;
5. Not being in English or Portuguese

The search resulted in seventy-two (72) retrieved papers, after removing duplicate papers a total of sixty-nine (69) paper abstracts were read by the authors. For each abstract it was attributed the inclusion and exclusion criteria. Figure 2 depicts the distribution count of paper for each criteria combination. The papers placed within the black rectangle refers to the papers to which were attributed at least one inclusion criteria and none exclusion criteria. That is, these are the works considered to be similar to the present one. The work myDIG: Personalized illicit domain-specific knowledge discovery with no programming [11] was the only one that was classified as a similar work by the criteria defined.

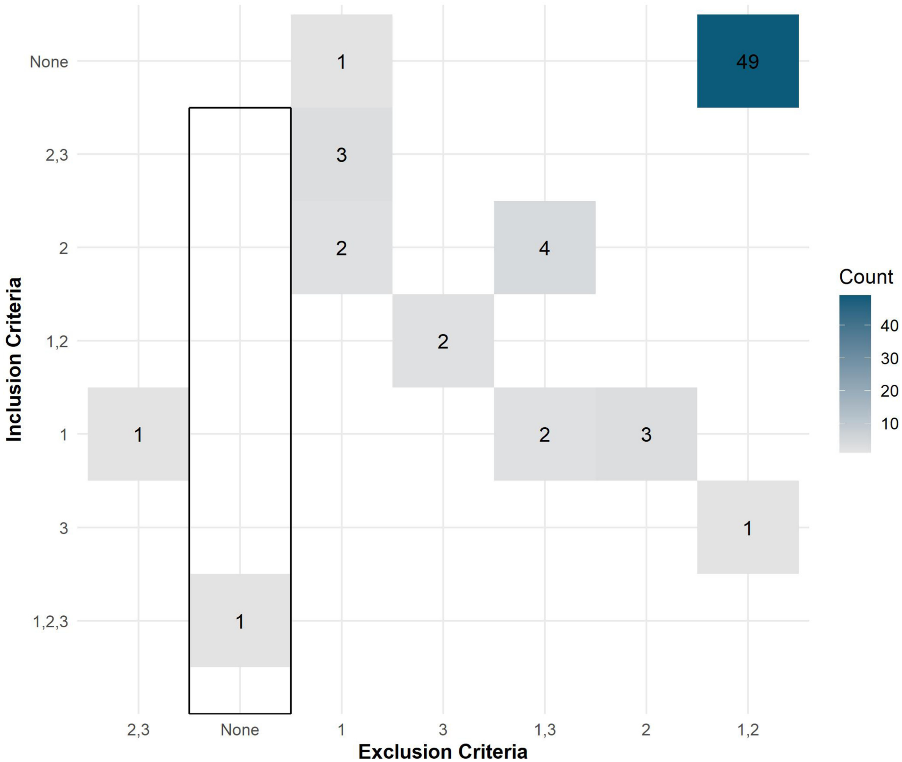


Fig. 2. Inclusion and exclusion criteria count

The work was developed at the Information Sciences Institute of University of Southern California, and presents a framework that allows investigative domain specialists to build and interact with their own knowledge graphs from the web pages. As one would expect, there are similarities and differences between myDIG and the present work.

The main similarity is found in the problem to be solved. Both works acknowledges that domain specialists struggle to keep up with the information creation. At the same time, the advances of data processing with Machine Learning, that would allow a way to narrow the gap between information creation and assimilation, requires programming and machine learning skills, that is not commonly found in domain specialists, restricting the number of domain specialists that can make use of such advances.

On the other hand, the main difference is found in the user profile. Both works have in mind domain specialists. However, while myDIG is focused in a case where the user has a well defined idea of what she is looking for, the present work focuses on the step where the domain specialist needs to have an overview of the knowledge relation in his corpus, that is, an easy to assimilate and interactive content summarization. Another difference is found in the input data, myDIG uses web pages while the present paper works is build upon natural language text. One final difference worth mentioning is related to the availability of the framework. The myDIG paper indicated a GitHub repository with the framework code, and therefore is was not attributed to it the third exclusion criteria. However, when the authors of the present work read the full myDIG paper it was explained that the engine that transform web pages into a knowledge graph is maintained by a private company and its not available and therefore it was not explained how it worked. This work on the other hand was built upon open source technologies and is also completely available[1].

Next sub-section presents the proposed framework that aims at allowing domain specialists with no programming skills to benefit from machine learning advances.

2.2 Proposed Framework - KG4All

Figure 3 presents the proposed framework, named as KG4All, that stands for Knowledge Graphs for All. The image can be read starting from the left side user icon and following the lines direction. The user uploads a corpus to a web application. This web application then sends the text from the corpus to a back-end. This stage is where the machine learning algorithms are used in order to build a knowledge graph from the texts. Once the Knowledge Graph is created the web application makes use of interactive tools, allowing the user to interact with the knowledge contained in the corpus that was uploaded.

This work, as mentioned in the last paragraph of Sect. 1, presents preliminary results of the process of building the KG4All. Specifically, it presents the first

[1] Web interface code: https://github.com/viniciusmsousa/kg4all. Data Processing workflow: https://github.com/viniciusmsousa/KG4All-data-processing-explained. At the current stage these components are not connected in the application, as explained in Sect. 3.

results of the elements inside the black rectangle. That is, the *corpus* element, highlighted with red in Fig. 3, has not been implemented yet.

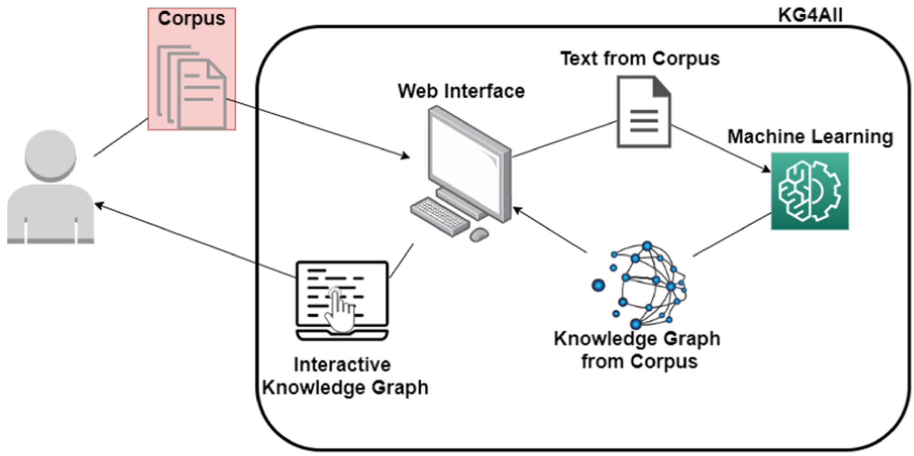


Fig. 3. Proposed framework (Color figure online)

A few practical considerations should be made. The choice to build a web application was made as the result of the following reasoning: The utilization of digital a tool is necessary mainly due to the fact that large amount of data processing is only possible through computers. Therefore, the real decision to be made is whether to build an web application or a smartphones app. The authors have chosen to build the web application for the following reasons. First, given the authors background build an web application presented less technical challenges. And secondly, people tend to be more productive on personal computers when compared to smartphones [1]. In order to build the presented framework the authors used the Shiny R package [7], which is a framework to build web application using the statistical programming language R [16]. Examples of others apps built with the framework can be found in the maintainer official gallery web page[2]. The main advantage of the framework is that it allows the creation of fully functional web application with in a relatively simple structure. The *en_core_sci_sm* model from the SciSpacy [13] python package was the choice to build the NLP tasks, that are explained in the Subsect. 2.3. Finally, as explained in Subsect. 3.1, the implementation was made using the metadata file from the COVID-19 Open Research Dataset Challenge (CORD-19) [3] and the raw data used for the results presented can be found in the link[3].

With that in mind the rest of this section presents the steps taken in order to achieve the preliminary result, shown in Sect. 3.

[2] https://shiny.rstudio.com/gallery/.

[3] https://drive.google.com/drive/folders/1YAHpv4-93rqMy94CyP830fRzN81Cwk9_?usp=sharing.

2.3 NLP

The objective is to allow the user to upload it's own corpus into KG4All, then from this corpus a knowledge graph is built. This section presents the text processing tasks that are responsible to create the triplets set from the texts. In other words, the *Machine Learning* block in the Fig. 3. The general task, *i.e.*, extract triplets from natural language text, can be splited into two sub-tasks: (i) Name Entity Recognition and (ii) Entity Linking.

Name Entity Recognition (NER) labels sequences of words in a text which are the names of things, such as person, company, etc. [21]. For example take the following natural language statement:

$$\text{Armstrong landed on the moon.}$$

After a NER processing this statement could be annotated as follows:

$$\text{Armstrong}_{\text{person}} \text{ landed on the moon}_{\text{location}}.$$

Since KG4All is implemented in the medical domain it is needed a source to get medical entities definitions. The Unified Language Medical System (UMLS) [6] provides just that. A few examples are shown in Table 1 and the full database with the definitions and relations from UMLS used in this work can be found in this link[4].

Table 1. Examples of medical entities from the UMLS.

Entity type	Entity name
Intellectual Product	Clinical Trial Objective
Virus	Avipoxvirus
Cell Component	Azurophilic granules
Temporal Concept	Priority
Bird	Aves
Intellectual Product	Report (document)
Population Group	Donor person

Therefore, an example of a NER annotated medical text could look like:

$$\text{The report}_{\text{Intellectual Product}} \text{ on the Avipoxvirus}_{\text{Virus}} \text{ is the current priority}_{\text{Temporal Concept}}.$$

The second sub-task is called Entity Linking which aims at finding a relation between two entities [21]. For example, by reading the statement **Armstrong landed on the moon** the human cognition interprets the semantic meaning

[4] https://drive.google.com/drive/folders/1kEw1_rJA7pI5VycmaXBVwbN0XMWUMsST?usp=sharing.

and concludes that there is a link between the entities **Armstrong** and **moon**. And this link is **landed_on**. Finally, this knowledge can be represented in a triplet form as:

$$(\text{Armstrong}, \text{landed on}, \text{moon})$$

Entity linking aims at using algorithms to detect these relations. The algorithms usually integrate three steps to link entities [21]:

1. Entity mention spotting: Detects mentions in the text of multiple entities;
2. Entity mention mapping: Lists the possibles entities from a formal knowledge base;
3. Candidate Selection: Selects, based on a criteria, which candidates are indeed linked with the mentioned entity.

Therefore, by completing this two sub-task it is possible to build a knowledge graph from text. An example of the data processing workflow developed by authors to create a KG from Medical text can be found on this github page (prepared by the authors)[5]. It presents the use of the SciSpacy [13] which is a open source python [20] framework dedicated to dealing with scientific texts from medical domain. The framework allows a large range of tasks, but the purpose of this research it was focused on the Named Entity Recognition and Entity Linking.

Once the data processing workflow is completed it is possible to create tools to allow the user to interact with the knowledge graph without having to program anything, this is the topic of the Subsect. 2.4.

2.4 Knowledge Graph Visualization

Having the extracted triplets from the corpus the next step towards the proposed framework is to allow a user to interact with the knowledge graph. As shown in Sect. 1, a knowledge graph has a network structure, *i.e.*, nodes (the entities) connected by edges (the relations).

Producing and examining a network plot is often one of the first step in a network analysis, since its overall purpose is to allow a better understanding of the underling structure in the data [12]. Figure 1 is an example of how a network can be visualized in order to reveal the underlying structure of the data. The use of aesthetics can enhance certain feature from the data in a better visual form. Color the nodes to indicate different types of nodes and change the edge size to depict the relation strength or count are two ways to do so. Therefore, the first interaction element implemented in the KG4All is the tool that allows the user to view a network graph from the knowledge graph extracted form the corpus, by selecting a document of interest.

In summary, this section started demonstrating, in Subsect. 2.1, the research gap in allowing domain specialists to benefit from the advances in the machine

[5] https://github.com/viniciusmsousa/KG4All-data-processing-explained/blob/main/01DataProcessingExplained.ipynb.

learning and natural language processing research in order to interact with a large number of documents. Second, Subsect. 2.2, presented and explained in a high level a possible way towards fulfilling the gap previously mentioned with the KG4All framework. Thirdly, Subsect. 2.3 explained the tasks of entity recognition and entity linking, which are the tasks that the presented work relies upon. And finally, the current sub section justified the choice of using network graphs to create an interactive knowledge graph as kick start to the web application. The next session presents the results obtained as this research evolves.

3 Results

This section presents the functional prototype of the KG4All framework. As stated before the KG4All source code is open source, currently it is present in two github repositories[6] due to the fact that the Web Application is not integrated with the Machine Learning back end yet. And the application can be accessed throught the link viniciusmsousa.shinyapps.io/KG4All[7]. The prototype current main features are: (i) detects the relations within the abstracts from the COVID-19 Open Research Dataset Challenge (CORD-19) [3] and (ii) connect these relations to the UMLS relations mapping. The following of this presents the domain implementation and test corpus in Subsect. 3.1. Next, the web interface components in Subsect. 3.2. The triplets display component in Subsect. 3.3 and finally the interactive graph in Subsect. 3.4.

3.1 Domain Implementation

The Covid-19 pandemic breakout in early 2020 and changed most people's life. The subject became an important topic in the international organizations agendas. Due to the fact that this research was taking place during the pandemic peak in Brazil the authors decided to implement the proposed framework in the Medical Domain. Specifically, it was chosen to develop a data processing workflow that works with medical texts, based in the Unified Medical Language System [6] and tested it to generate the results using the abstracts of papers related to the coronavirus extracted from the COVID-19 Open Research Dataset Challenge (CORD-19) [3], which is the result of a response coordinated by the White House to make available the scientific publications related to the coronavirus.

3.2 Web Interface Components

Figure 4 presents the KG4All Web Interface, *i.e.*, the interface that the domain specialist interacts with. It is composed mainly by three components marked in the figure. Component 1 is the search bar that allows the user to search for a

[6] Web application: https://github.com/viniciusmsousa/kg4all. Data Processing: https://github.com/viniciusmsousa/KG4All-data-processing-explained.

[7] https://viniciusmsousa.shinyapps.io/KG4All.

desired document using words. Component 2 presents the triplets extracted from the abstract of the document selected in component 1. And finally, component 3 presents the interactive graph visualization of the relations.

The Subsect. 3.3 explains the KG4All's triplets component.

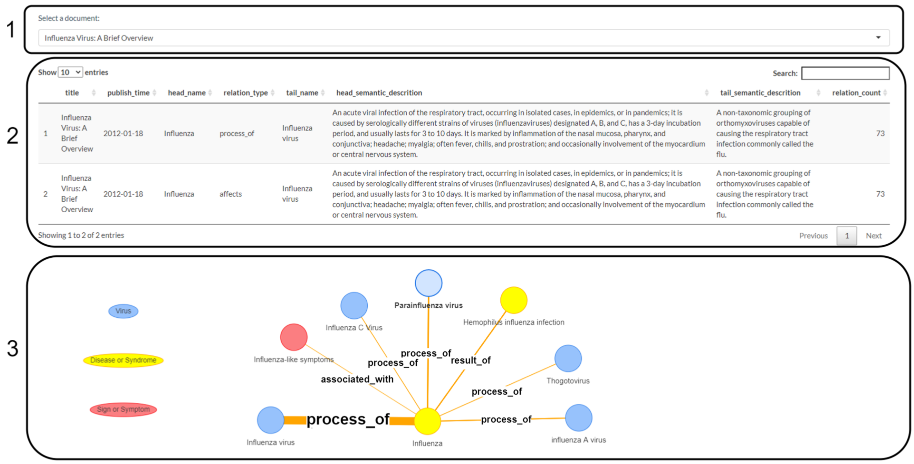
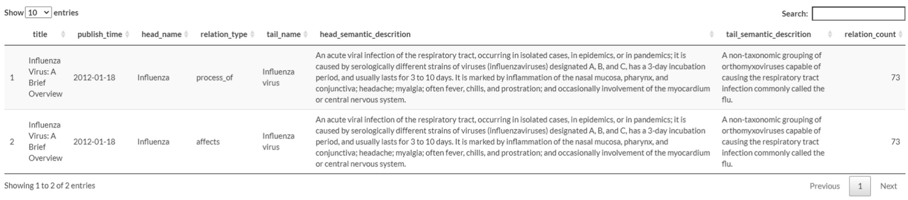


Fig. 4. KG4All Web Interface ('Influenza Virus: A Brief Overview')

3.3 Triplets Component

The triplets are shown in component 2 of the KG4All. Figure 5 is a zoom in component 2. Each row each represents on relation found in the text. The relation in itself is in the columns *head_name*, *relation_type* and *tail_name*. The other columns of the table presents additional information about the relation. We highlight that the column entitled *relation_count* depicts the number of times that the relation occurred in the whole corpus.



Show 10 entries — Search:

	title	publish_time	head_name	relation_type	tail_name	head_semantic_descrition	tail_semantic_descrition	relation_count
1	Influenza Virus: A Brief Overview	2012-01-18	Influenza	process_of	Influenza virus	An acute viral infection of the respiratory tract, occurring in isolated cases, in epidemics, or in pandemics; it is caused by serologically different strains of viruses (influenzaviruses) designated A, B, and C, has a 3-day incubation period, and usually lasts for 3 to 10 days. It is marked by inflammation of the nasal mucosa, pharynx, and conjunctiva; headache; myalgia; often fever, chills, and prostration; and occasionally involvement of the myocardium or central nervous system.	A non-taxonomic grouping of orthomyxoviruses capable of causing the respiratory tract infection commonly called the flu.	73
2	Influenza Virus: A Brief Overview	2012-01-18	Influenza	affects	Influenza virus	An acute viral infection of the respiratory tract, occurring in isolated cases, in epidemics, or in pandemics; it is caused by serologically different strains of viruses (influenzaviruses) designated A, B, and C, has a 3-day incubation period, and usually lasts for 3 to 10 days. It is marked by inflammation of the nasal mucosa, pharynx, and conjunctiva; headache; myalgia; often fever, chills, and prostration; and occasionally involvement of the myocardium or central nervous system.	A non-taxonomic grouping of orthomyxoviruses capable of causing the respiratory tract infection commonly called the flu.	73

Showing 1 to 2 of 2 entries — Previous 1 Next

Fig. 5. Triplets component ('Influenza Virus: A Brief Overview')

For example, from the first line of the figure it can be seen that the entity **Influenza** (head entity) is a **process of** (relation) the **Influenza Virus** (tail entity). Next step in to connect this entities with other entities found in the corpus, through the UMLS. This is presented in the Subsect. 3.4.

3.4 Interactive Graph

The last implemented component is the interactive knowledge graph, presented in Fig. 6. One might note that there more relations in the graph than in the triplets table. This is due to the fact that the graph shows the relations found in the abstract with the relations found in the whole corpus that involves the entities from the selected text. This allows the domain specialist to have a general view of how the selected text is related with the whole corpus. It is worth mentioning that even though it is not implemented, the authors are studying ways to explicit the document from which the additional relations are from.

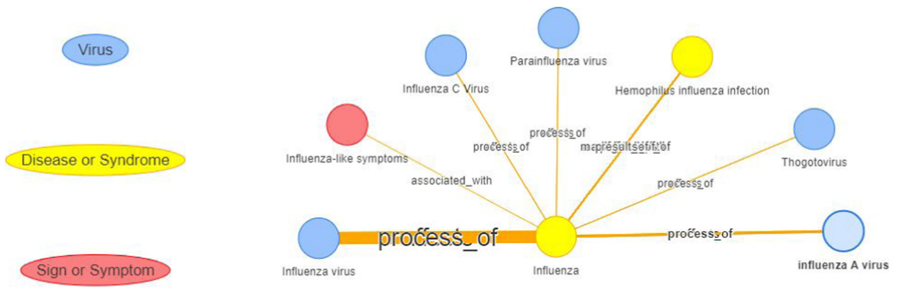


Fig. 6. Interactive knowledge graph ('Influenza Virus: A Brief Overview')

As is expected in network visualizations KG4All use some aesthetics to add more information to the relations. The nodes color demonstrates to which group each node belong to. For example, **Influenza** is classified by the UMLS as a **Disease or Syndrome**. On the other hand **influenza A virus** is classified as a **virus**. Another aesthetics used in the interactive graph is the edge (or link) size. It proportional to the number of times that the relation was present in the corpus.

Section 4 presents considerations about the results, improving directions that are in the authors workflow.

4 Discussion

A few considerations about KG4All itself. First, the current implementation uses a selected dataset to create the interactive knowledge graph, however this is a temporary implementation. Once the upload interface and the integration between the web interface and the back end model is done, KG4All will have an upload interface where the users will be allowed to upload their own medical corpus. Second, there are two factors that impacts KG4All computing cost: (i) The model used to detect the medical entities and (ii) the size of the corpus that is submitted to the data processing pipeline. The model that is current being used is the *en_core_sci_sm* [13] and once it is loaded it uses 132 MiB of memory. And the dataset used to create the prototype, with 81.354 medical abstracts, used around 10 GB while running on win10 with intel i7. It is worth noting that in practical use the authors expect smaller corpus, for instance, the result OS a search in scientific articles database. Third, the use of machine learning algorithms to extracted the triplets cannot guarantee that all the entities relations present in the text will be extracted. How ever, as shown in the SciSpacy paper [13] the amount of relations detected are not insignificant, providing a reasonable summarizing of the knowledge present in the corpus. And, finally, there are both some implementations as well as corrections to be made on the current state. For example, a way to explicit from each document the entity was extracted, when it is a relations that is not in the selected document is a implementation to be made. In some cases there are overlapping of the edges name, which is a correction in the back log.

Besides the practical differences from the myDIG [11] work, explained in Subsect. 2.1, the authors believes that KG4All complements the myDIG, in the sense that the same issue, gap between domain specialists information assimilation and creation, is being addressed. And contributing for a different group of information users by focusing in an open source tool for knowledge graph creation and interaction.

5 Conclusion

The present work has argued that there is a gap between information creation and assimilation. This gap impacts the domain specialists, a group that the traditional information tools does not satisfies their information necessities. It has also been argued that the research advances in the field of information retrieval through knowledge graph using machine learning algorithms is evolving and provides ways to narrow the information gap. However, such advances are relied on a high level of computational and mathematical complexity. This high complexity results in the need of programming and machine learning skills in order to make use of the advances. Such skills are not commonly found in domain specialists. It was presented the KG4All prototype, which is a framework that will allow users to upload their own corpus and interact with a knowledge graph created from the corpus without the need of programming skills. The domain that the KG4All is implemented is the medical one, due to the fact that the Covid-19 pandemic took place while this research was taking place. There are other works proposing solutions to the same problem however with a differences in the target domain specialists user profile, and therefore, the present work contributes to the research on how to make the advances in knowledge graph through machine learning usage.

References

1. Adepu, S., Adler, R.F.: A comparison of performance and preference on mobile devices vs. desktop computers. In: 2016 IEEE 7th Annual Ubiquitous Computing, Electronics Mobile Communication Conference (UEMCON), pp. 1–7 (2016). https://ieeexplore.ieee.org/document/7777808
2. Agarwal, R., Dhar, V.: Editorial-big data, data science, and analytics: the opportunity and challenge for IS research. Inf. Syst. Res. **25**(3), 443–448 (2014). https://doi.org/10.1287/isre.2014.0546
3. AI, A.I.F. https://www.kaggle.com/allen-institute-for-ai/CORD-19-research-challenge
4. Araújo, C.A.A.: Correntes teóricas da ciência da informação. Ciência da informação **38**, 192–204 (2009). http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0100-19652009000300013&nrm=iso
5. Araújo, C.A.A.: Fundamentos da Ciência da Informação: correntes teóricas e o conceito de informação. Perspectivas em Gestão Conhecimento **4**(1), 57–79 (2014). https://periodicos.ufpb.br/ojs2/index.php/pgc/article/view/19120
6. Bodenreider, O.: The unified medical language system (UMLS): integrating biomedical terminology. Nucleic Acids Res. **32**(1), D267–D270 (2004)
7. Chang, W., Cheng, J., Allaire, J., Xie, Y., McPherson, J.: shiny: Web Application Framework for R (2020). https://CRAN.R-project.org/package=shiny, R package version 1.5.0
8. Elbashir, M., Collier, P., Davern, M.: Measuring the effects of business intelligence systems: the relationship between business process and organizational performance. Int. J. Accounting Inf. Syst. **9**(3), 135–153 (2008). https://www.scopus.com/inward/record.uri?eid=2-s2.0-51249116446&doi=10.1016%2fj.accinf.2008.03.001&partnerID=40&md5=f6748444fd6918d43aa33b5de2c118d3, cited By 254
9. Hoyt, C.T., et al.: Re-curation and rational enrichment of knowledge graphs in Biological Expression Language. Database 2019 (2019). https://academic.oup.com/database/article/doi/10.1093/database/baz068/5521414
10. Ji, S., Pan, S., Cambria, E., Marttinen, P., Yu, P.S.: A Survey on Knowledge Graphs: Representation, Acquisition and Applications (2020). https://arxiv.org/abs/2002.00388
11. Kejriwal, M., Szekely, P.: myDIG: personalized illicit domain-specific knowledge discovery with no programming. Future Internet **11**(3) (2019). https://www.mdpi.com/1999-5903/11/3/59
12. Luke, D.A.: A User's Guide to Network Analysis in R. UR, Springer, Cham (2015). https://doi.org/10.1007/978-3-319-23883-8
13. Neumann, M., King, D., Beltagy, I., Ammar, W.: ScispaCy: fast and robust models for biomedical natural language processing. In: Proceedings of the 18th BioNLP Workshop and Shared Task, Florence, pp. 319–327. Association for Computational Linguistics (2019). https://www.aclweb.org/anthology/W19-5034
14. Olszak, C., Ziemba, E.: Approach to building and implementing Business Intelligence systems. Interdisciplinary J. Inf. Knowl. Manag. **2**, 135–148 (2007). https://www.scopus.com/inward/record.uri?eid=2-s2.0-77749242597&partnerID=40&md5=fd70fbb98a2ddee0b6daf68f28050db5, cited By 81
15. Pinto, A.L., Silva, A.M., Sena, P.M.B.: Ontologias baseadas na visualização da informação das redes sociais. Prisma.com (Portugal) **24**(13), 5–24 (2010). https://www.brapci.inf.br/index.php/res/v/68060

16. R Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna (2020). https://www.R-project.org/
17. Saracevic, T.: Ciência da informação: origem, evolução e relações. Perspectivas em Ciência da Informação **1**(1) (1996). http://portaldeperiodicos.eci.ufmg.br/index.php/pci/article/view/235
18. Sen, S., Li, T.J., Team, W., Hecht, B.: WikiBrain: Democratizing Computation on Wikipedia (2014). https://doi.org/10.1145/2641580.2641615
19. Shannon, C.E.: A mathematical theory of communication. Bell Syst. Tech. J. **27**(3), 379–423 (1948). https://onlinelibrary.wiley.com/doi/abs/10.1002/j.1538-7305.1948.tb01338.x
20. Van Rossum, G., Drake, F.L.: Python 3 Reference Manual. CreateSpace, Scotts Valley, CA (2009)
21. Waitelonis, J.: Linked Data Supported Information Retrieval. Ph.D. thesis, Karlsruher Institut für Technologie (2018)
22. Yuan, J., et al.: Constructing biomedical domain-specific knowledge graph with minimum supervision. Knowl. Inf. Syst. **62**(1), 317–336 (2020). https://link.springer.com/article/10.1007/s10115-019-01351-4