A method of knowledgebase curation using RDF Knowledge Graph and SPARQL for a knowledge-based clinical decision support system

Xavierlal J Mattam and Ravi Lourdusamy

Sacred Heart College(Autonomous), Tirupattur, Tamil Nadu, India

Abstract

Clinical decisions are considered crucial and lifesaving. At times, healthcare workers are overworked and there could be lapses in judgements or decisions that could lead to tragic consequences. The clinical decision support systems are very important to assist heath workers. But in spite of a lot of effort in building a perfect system for clinical decision support, such a system is yet to see the light of day. Any clinical decision support system is as good as its knowledgebase. So, the knowledgebase should be consistently maintained with updated knowledge available in medical literature. The challenge in doing it lies in the fact that there is huge amount of data in the web in varied format. A method of knowledgebase curation is proposed in the article using RDF Knowledge Graph and SPARQL queries.

Keywords

Clinical Decision Support System, RDF Knowledge Graph, Knowledgebase Curation.

1. Introduction

Decision Support has been a crucial part of a healthcare unit. In every area of a health care facility, critical and urgent decisions have to be made. In such extreme situations, leaving lives at stake totally to mere human knowledge and memory is a very big risk. It can often lead to untold misery to the stakeholders and disaster to such facilities. In 2009 when Health Information Technology for Economic and Clinical Health (HITECH) was promulgated in the United States of America, monetary aid was disbursed for success in the implementation of Clinical Decision Support System(CDSS). It was because CDSS, although being far from a perfect system, was found to be better than mere human decisions. Since then, a lot of study and research is being done to perfect the CDSS.

One of the enlightening issues that came to the forefront during the recent pan-demic outbreak was the lack of widespread knowledge and awareness in the diagnosis and treatment of the disease. Although there were many breakthroughs published in medical literature globally, down-to-earth use of any of them were slow and far-between. It would have not been the case had there been CDSS that was capable of automatically acquiring reliable knowledge from authenticated medical literature. Such CDSS could alter heath workers with an all-round advanced knowledge at the moment of crucial decisions.

In the article, some aspects of the recent advances in the technology used in CDSS are described together with related works carried out in the development of knowledge-based CDSS before the proposed method of knowledgebase curation in CDSS is explained. In the section 2 that follows, a brief background is given into re-cent developments in knowledge-based CDSS. In section 3, some recent works on possible methods of knowledge base curation are mentioned. Then the proposed method is explained in section 4 and that is followed by a brief discussion on the proposed method in section 5. Finally, in section 6, a summarized conclusion is made.

2. Background

ISIC 2021, February 25-27,2021, New Delhi, India
EMAIL:xaviermattam@gmail.com (X. Mattam);
ravi@shctpt.edu(R. Lourdusamy);
ORCID: 0000-0002-5182-3627(X. Mattam);

CDSS has evolved gradually with the ample technological developments that has happened in the past few decades. The system is essentially centered on high adaption and effective use of constantly updated knowledge. With the evolution of CDSS over the years, there has also been a consistent evolution in its definition from a mere use of information technology for data entry to a hi-tech complex system that provides individual specific, intelligently filtered and efficiently presented knowledge for clinicians, staff, patients, or other individuals [1, 2]

CDSS can be broadly classified as knowledge-based CDSS and non-knowledge-based CDSS. The knowledge-based CDSS are designed to mimic the knowledge processed by a human expert. Such systems were earlier termed as the expert systems. Non-knowledge-based systems, on the other hand, rely on statistical data that is available to help in decision making. These systems make full use of the machine learning and neural network algorithms to predict possible outcomes [3, 4, 5, 6].

2.1. Knowledge-based CDSS

Knowledge-based systems evolved from expert systems. While the expert systems were built on the knowledge of human experts, the knowledge-based systems have the capability to acquire knowledge from different sources and build upon it. So, while the expert system could be ranked according to the knowledge of the expert, the knowledge-based systems had the capacity of greater knowledge [7, 8].

The knowledgebase of the knowledge-based CDSS ultimately determines the effectiveness of the CDSS [9]. The acquisition, representation and the integration of knowledge base in the workflow is vital for the success of CDSS [10]. The process of selecting, organizing, and looking after the knowledge in the knowledge base makes the knowledgebase efficient and the CDSS successful. The two important facets of the curation process is the method of knowledge acquisition and knowledge representation [11].

One way to build a cost effective knowledge-based CDSS is to use commercial knowledgebases that are available. It could reduce cost of the development time of the CDSS and also because of the common availability of such knowledgebases, it could also be cost effective in terms of price [12]. But such knowledge acquisition might lead to the knowledge-acquisition bottleneck as certain standards and formalization will have to be maintained for the knowledge portability. Such a knowledge-acquisition bottleneck could harm the effectiveness of the CDSS and freeing the CDSS of the bottleneck makes the knowledge acquisition process complex and difficult [12], [13]. Another bottleneck in the curation of knowledgebase lies in the maintenance of the knowledgebase [14]. Together with creation of knowledgebase, its verification and constant updating is equally important. The verification and validation of the knowledgebase involves transparency, updatability, adaptability, and learnability [15].

A knowledgebase is judged by its accuracy, completeness and the quality of its data. So, the construction of the knowledgebase is done in such a manner that these three factors are enhanced to the maximum. The methods of constructing knowledgebases can be classified into four main groups. There are closed methods in which the knowledgebase is are manually fixed by experts, open methods in which knowledgebases are curated by volunteers, automated semi-structured methods in which the knowledgebases are procured from semi-structured texts automatically by using rules that are programmed into the system and the automated unstructured methods that use artificial intelligence algorithms to extract knowledge from unstructured texts [16].

2.2. Knowledge Graphs for knowledge base

Knowledge graphs are knowledgebases in which knowledge is expressed in a graph structure having nodes to represent the concepts or entities and edges between the nodes to represent the relationship between those entities or concepts [16, 17, 18, 19, 20, 21, 22, 23]. There are diverse definitions for knowledge graphs varying according to the purpose for which the knowledge graph is created or by the knowledge graph model [24]. Although Google is credited for the popularity of knowledge graphs from 2012 [24, 25, 26], the term knowledge graph was used in a report in 1973 with a very similar meaning [27] and later in 1982 the term was used to represent textual concepts using graphs. There has been decades of study in representing knowledge using graphs [28].

The maintenance of knowledge graphs has the processes of creation, hosting, curation and deployment. The process of creation can be manual as in the case of expert systems or semi-automatic or automatic. Apart from these, there is also a method of annotation by mapping the knowledgebase entities to the source without actually keeping the entities in the knowledgebase. Hosting or storage processes use various methods of keeping knowledge in the knowledgebase. The curation processes involve three steps, namely, the assessment of new knowledge, its cleaning and its enrichment by detecting the source of the knowledge, integrating it with existing knowledgebase, detecting duplication and correcting entity relations. Once the knowledgebase is ready, it is deployed in appropriate application [29].

There are various sources of knowledge that can be utilized for the creation of knowledge base. Textual knowledge that can be in the form of newspapers, books, scientific articles, social media, emails, web crawls, and so on is a very rich source of knowledge for building a knowledge graph for CDSS. However, the process of extracting knowledge from text is complex and involves the application of Natural Language Processing(NLP) and Information Extraction(IE) techniques. Curation of the knowledgebase using these techniques may follow a combination of five stages. In the pre-processing stage, the text is analyzed for atomic terms and symbols. Some of the techniques used in the pre-processing stage are Tokenization, Part-of-Speech(POS)tagging, Dependency Parsing and Word Sense Disambiguation(WSD). After the pre-processing stage is the Named Entity Recognition (NER) stage in which the various concepts or entities that forms the nodes of the graph are identified. The NER is followed by the Entity Linking (EL) stage in which an association is made between the entities that are identified in the text and the entities in the existing knowledge graphs so that the similar entities could be placed side-by-side. During the Relation Extraction (RE) stage, the relation between the various entities taken from the text are considered using a various RE techniques. Finally the extracted relation is joint to the entities in the last stage of the text processing [22].

2.3. RDF Knowledge Graphs

Resource Description Framework(RDF) is a World Wide Web Consortium (W3C) specification to represent knowledge in the form of triples (subject, predicate, object) containing references, literals or blank [30], [31]. RDF can be modelled as directed label graphs in which the subject and object are represented by the vertices or nodes and the corresponding predicate are represented by the labelled edges [32, 33, 34, 35]. RDF graphs are widely used to represent knowledge graphs like in the cases of Freebase, Yago, and Linked Data since the billions of triples scattered across the web can be captured and integrated with the existing knowledge using powerful abstraction for representing heterogeneous, partial, scant, and potentially noisy knowledge graphs [36], [37]. Unlike property graphs that is also quite popular representation of knowledge graphs due to its property and value representation for its edges, the use of metadata in RDF knowledge graphs allows the convenient distributed storage of knowledge. That also makes RDF graphs more flexible than property graphs [37, 38].

RDF knowledge graphs are stored as triples in a Triple store or RDF stores. The flexibility of RDF stores is its greatest advantage. Since the RDF knowledge graph has the ability to link any number of entities with their relations, the RDF stores are also flexible enough to store them without restriction on size. Moreover any kind of knowledge can be expressed and stored using RDF knowledge graphs that allows its extraction and reuse by different applications [39].

In the case of textual knowledge, RDF knowledge graphs are helpful in finding the Thematic Scope or the Topic Model of a text. The topic category or the semantic entities in a set of documents is abstracted using the Thematic Scope or the Topic Model [38, 40]. Since the RDF knowledge graphs of a document is represented as a set of triples in which each triple is considered a word or an entity, there is the possibility of detecting word and phrase patterns automatically by clustering word groups that best characterize a document. Some of the methods of the Topic Modelling of RDF knowledge graphs are Latent Semantic Analysis (LSA), Probabilistic Latent Semantic Indexing (pLSI) and Latent Dirichlet

Allocation (LDA). The challenges faced in Topic Modelling include sparseness of the entities, a lack of context especially when words used have multiple meanings especially when the entities are sparce and the use of unnatural language like the use of special characters or unusual casing of letters which normally are removed while pre-processing the text. Normally, these challenges are overcome by supplementing the text or modifying the method of Topic Modelling [40, 41, 42]. Entity summarization which is the best way of summarize an entity by identifying a limited number of ordered RDF triples is one of the problems that is solved using Topic Models of RDF knowledge graphs [41, 42]. Entity summarization has many applications like search engines and is useful for research activities. The existing entity summarization techniques can be classified into the generic techniques that apply to a wide range of domains, applications and users and the specific techniques that make use of external resources or factors that are effective only in specific domains or applications. While the generic techniques make use of universal features like frequency and centrality, informativeness, and diversity and coverage, the specific techniques make use of domain knowledge, context awareness and personalization [43].

3. Related works

Extracting knowledge from unstructured textual sources has been a challenge. Several studies have been done in order to solve the problem of retrieving meaningful and relevant knowledge from literature since it is crucial for decision support in systems like the CDSS. Some relevant techniques have been delt with in the earlier sections on knowledge graphs and RDF knowledge graphs. As part of the proposed method, certain other related techniques have to be explained in order to have a complete picture of the complexity of the problem of curating the knowledgebase for CDSS.

3.1. Question Answering using SPARQL

Question-answering (QA) is a process of retrieving knowledge from different sources using a part or the whole expression of a question in natural language. The question in the natural language can be interrogative in which case it will be a factoid type of question and its answer will be a fact from the knowledge source or the question could be statement in which case the answer will be in the form of either a list or a definition or hypothetical statement or a causal remark or a relationship description or procedural explanation or just a confirmation. The sources of knowledge are normally unstructured and is a set of documents, video clips, audio clips, or text files that are given as input to the systems [44]. QA systems make use of Information Retrieval, Information Extraction and Natural Language Processing(NLP) techniques [45].

QA systems have a long history of development starting with the earliest popular system BASEBALL in 1961 and LUNAR made in 1972. The Text REtrieval Conference (TREC) that began in 1992 for large scale information retrieval accelerated interest and growth in QA systems. Other forums and campaigns such as the Cross Language Evaluation Forum (CLEF) and NIITest Collections for IR systems (NTCIR) campaigns also enhanced the QA systems. Noteworthy QA systems include IBM Watson, and the other commercial personal assistant software like Apple's Siri, Amazon's Alexa, Google's Assist and Microsoft's Cortana [44, 45, 46, 47, 48, 49].

SPARQL is a query language in which a pattern in a query is matched with that in a graph from different sources. The matching is done in three stages. It begins with the pattern matching involving features like optional parts, union of patterns, nesting, filtering or constraining values of matches, and selecting the data source to be matched by a pattern. Once these features are applied, the output is computed using standard operators like projection, distinct, order, limit, and offset to modify the values that is got from pattern matching. Finally, the result of the SPARQL query is given in one of the many forms like Boolean answers, the pattern matching values, new triples from the values, and resources description. Working with RDF knowledge graphs require the use of SPARQL [50, 51]. In order to apply NLP techniques used in the QA systems over SPARQL queries, query builders such as QUaTRO2, OptiqueVQS, NITELIGHT, QueryVOWL, Smeagol, SPARQL Assist language-neutral query composer, XSPARQL-Viz, Ontology Based Graphical Query Language, NL-Graphs and so on are used [52]. Some of the challenges in the use of SPARQL for QA systems include lexical gap, ambiguity, multilingualism, use of complex operators, distributed knowledge and in the use of procedural, temporal, spatial templates [53].

3.2. Ranked RDF Triples and Federated Search

Ranking SPARQL query results is an important process for applications involving searches, QA and entity summarization techniques. Ranking of RDF triples can be over resources, properties, or triples as a whole but a combined ranking of both the triples and its entities are important for RDF knowledge graphs for faster and efficient searches in the knowledgebase [54]. Ranking can be done on structured data using structured queries that results in a structured graph. Such rankings are structure-based ranking and mostly use an extension of ranking algebra that was earlier used in relational database. Content-based ranking on the other hand tries to rank the content of structure or unstructured data. In content-based ranking, the ranking is done according to the match between the pattern of the query and its holistic match in the knowledgebase. Further classification of query ranking can be as keyword queries on unstructured data like documents, structured queries on structured data, keyword queries on structured data, and keyword-augmented structured queries on structured data [55]. Ranking can also be based on the relevance or importance of the SPARQL query results with the topic on which the query is made. The relevance ranking requires the subject of the query to be clearly defined so that the results of the query can be ranked according its relevance to the subject. The importance ranking on the other hand specifies the importance that is given to the query result. In the importance ranking factors such as authoritative, trustworthy, and so on are placed for the ranking purpose for which human cognitive results are taken for consideration [56]. The ranking is placed along with the triple using tokens in an Extended Knowledge Graphs [57] or more prevalently using graph embeddings [58, 59].

Knowledge graphs can be centralized or distributed. Both the centralized and distributed knowledge graphs have their advantages and disadvantages [60]. When it comes to distributed knowledge graphs, federated query processing is used in which the result of the query is computed from different data source. The federated query processing accesses different autonomous, distributed, and heterogeneous data sources to without having any control over the sources. Federated query processing is more complex than the centralized system because of the many parameters involved in the query processing. Federated query processing makes use of federated query engine to search for the results over distributed sources [61]. The federated SPARQL query processing can be done either over different SPARQL endpoints or over linked data or over Distributed Hash Tables. The federated SPARQL query processing can also be classified either as catalog or index assisted processing or as catalog or index free processing or a combination of both [62].

3.3. Open Information Extraction

Information Extraction(IE) is an automated process of collecting a set of corresponding information of interest from a given sequences of unstructured data. IE has many applications such as part-of-speech tagging, named entity recognition, shallow parsing, table extraction, contact information extraction and so on. Methods used for IE can be classified as rule learning based extraction methods, classification based extraction methods, and sequential Labeling based extraction methods [63]. Open Information Extraction(OIE) is a text IE paradigm that enables relations discovery independent of the domain that is readily scalable to the variations in size and content of the web. OIE is technically capable of meeting the challenges of the IE, namely, automation of the process, heterogeneity of the web corpus and efficiency in extraction of information [64]. The OIE like Text Runner, Clause IE, OLLIE, and the like were data based using training data that were represented either by dependency parsing or parts-of-speech tagged text. The Rule-based OIE were manually programmed using the dependency trees or parts-of-speech tagged text. Two examples of Rule-based OIE are clauseIE and ExtrHech [65]. Another method of OIE is by linguistic analysis that shows the canonical ways in which verbs in a text is used to express relationship between entities. RE-VERB, ARGLEARNER and R2A2(combination of REVERB and ARGLEARNER) are examples of the linguistic analysis method. The earlier methods were based on the label, learn and extract stages of IE. The main drawback of the three-stage process was that the information extracted were either incoherent or uninformative and therefore they were of little use to applications. The linguistic-statistical analysis for extractions on the other hand identifying a more meaningful and informative relation phrase [66].

The biggest advantage of OIE is its ability to extract relationship between entities allowing queries like "(?, kill, bacteria)" or "(Bill Gates, ?, Microsoft)" to extract resultant missing relationships from a text corpus. Moreover, OIE will result in a compressed data for a knowledgebase [67]. Other than populating knowledgebases, OIE is also used for question answering, semantic indexing, semantic search and such target applications. Converting OIE triples into RDF knowledge graph is possible since the longer sentences are broken into triples with entities and relationship between entities leaving out the determiners and propositions. Knowledgebase populating using OIE has been a very useful application domain [68, 69]. Integrating the OIE triples with the exiting knowledgebase has been a research challenge and there have been many solutions proposed for it like the predicate or attribute level schema where similarity on names, types, descriptions, instances, and so on are mapped and universal schema to apply inferences got from OIE and the existing knowledge mapped at the instance-level. One of the problems of using universal schema is that the process ignores unseen entities and entity pairs and tries to overfit the space entities to large number of parameters. Rowless Universal Schema attempts to find inferences between predicated and relations so that the problem of unseen entities and entity pairs are solved. But it tends to completely ignore the existence of entities and thus it functions like the predicate or attribute level schema [69].

4. Proposed Method of Knowledgebase Curation

A CDSS is as efficient as its knowledgebase is. If the knowledgebase of the CDSS is highly adaptive to automatically and constantly update itself reflecting recent advances and local practice, then that will be a robust CDSS. The flexibility of the knowledgebase to accept knowledge from diverse sources and portability of the knowledgebase for various practice settings will make the knowledgebase more effective [9, 70]

4.1. Motivation

CDSS requires quick and reliable knowledge. Therefore, a centralized knowledgebase will be better than a federated search. Since knowledge on most of the advances found in medical literature, the knowledge extracted from the literature has to be found in the knowledgebase. The information extraction from medical literature can be done using OIE. If the user interface of the CDSS allows natural language questions to be asked, the questions can be converted through a QA system as a SPARQL query linked to through the knowledge graph. If answers are not found in the existing knowledgebase of the CDSS, it can than be passed through the OIE to relevant medical literature and the resultant knowledge can be integrated to the existing knowledgebase. The knowledgebase is so enhanced that most answer to query will be found in it and the updating will be done automatically once new knowledge is found in any form on the web.

There are many advantages of such centralized knowledge graphs. Centralized knowledge graphs can be controlled by a single entity when it comes to strategic issues such as symptoms for diagnostic systems of the CDSS. Such control increases the survivability and robustness of the CDSS. The uniform use of terms in centralized knowledgebase make it more stable. The fixed curation method of the knowledgebase of the centralized systems make the knowledge consistent and improves its quality. Moreover, the fixed schema of the centralized knowledge graphs allows uniform usage. The knowledge graphs allows the use of application programming interface (API) for knowledge retrieval and query processing [60].

4.2. Proposed Approach

The proposed approach of knowledgebase curation for CDSS has three stages. In the first stage, new knowledge is extracted from a medical literature using an OIE application. The OIE application is for knowledge extraction since it will result in triples that can be integrated to the existing knowledge graph of the CDSS. The triples got from the OIE application on the medical literature is queried using keywords from the CDSS interface for relevant knowledge using a QA system. If the query results in new knowledge being found, then those triples in RDF from are added to the existing knowledge graph of the CDSS. A table is maintained with the list of medical literature already checked for knowledge so that they need not be looked for new knowledge again. The algorithm for the proposed system is as shown in Algorithm 1.

Algorithm 1.
Curating the CDSS knowledge base using RDF Knowledge Graph

URI (Uniform Resource Identifier) table U = {u_1, u_2, …, u_n}
RDF Knowledge Graph G ={V,E} where V Є {v_1,v_2,….,v_n} and E Є {e_1,e_2,…,e_n}
RDF triple S = {s, p, o} (subject(s), predicate(p), object(o))
Keyword K = {k_1, k_2,…k_n} taken from the QA system of CDSS
1 Read u_i //URI of a new document
2 If u_i Є U GOTO Step 12
3 Else use OpenIE to create G
4 For every K
5 use QA system with SPARQL query to find k_i in G
6 For every S found
7 If S not in CDSS knowledge base
8 Append S to CDSS knowledge base
9 END For loop
10 END For loop
11 END Else
12 If another document exists GOTO Step 1
13 Else STOP

4.3. Evaluation of the Proposed Method

For the evaluation of the proposed algorithm, the precision and recall method are used as it is the typical form of evaluation metrics used in information extraction. During the process of information extraction or retrieval, there could be two types of knowledge that is obtained from the knowledge source. There is knowledge that can be considered important to the application and there is knowledge relevant to the query. In the proposed approach, since the query is based on keywords from the QA system, only knowledge relevant to the query is selected rather than all knowledge that is deemed important from the knowledge source. Therefore, the results of the OIE is restricted to just the knowledge relevant to the query. The application of the evaluation metrics is also bound by the consideration that only the sum total of the relevant knowledge found by the system proportionate to all the relevant knowledge that can be manually counted on the same medical literature is calculated rather that taking in consideration all the important knowledge present in the literature that is used in the test. The precision evaluation metric is given by the formula in equation (1)

$$\text{Precision} = \frac{\text{relevant RDF triples} \cap \text{retrived RDF triples}}{\text{retrived RDF triples}} \quad (1)$$

A contingency matrix can be formed using the relevance of the RDF triple as shown in table 1. If the triple retrieved by the system is relevant to the query than it is true positive otherwise it is false positive. So also, if a relevant triple in the knowledge source is not retrieved by the system then it is false negative and if a triple that is not relevant and is ignored by the system it is true negative.

Table 1
Contingency matrix according to the relevance of RDF triples

	Relevant	Not Relevant
RDF triple retrieved	True positive	False positive
RDF triple not retrieved	False negative	True negative

The formula to calculate the precision of the system using the contingency matrix can be given as in equation (2)

$$\text{Precision} = \frac{\text{total number of true positives}}{\text{total number of true positives and false positives}} \quad (2)$$

The percentage of the precision can also be calculated as in equation (3)

$$\text{Precision }\%=\frac{\text{total number of true positives}}{\text{total number of true positives and false positives}}\times 100 \quad (3)$$

For recall we take into consideration the proportion of the retrieved triples to the total relevant triples as in equation (4)

$$\text{Recall}=\frac{\text{relevant RDF triples} \cap \text{retrived RDF triples}}{\text{relevant RDF triples}} \quad (4)$$

Using the contingency matrix, the formula to calculate recall is as in equation (5)

$$\text{Recall} = \frac{\text{total number of true positives}}{\text{total number of true positives and false negatives}} \quad (5)$$

5. Discussion for Further Study and Development

The proposed method of knowledgebase curation using RDF Knowledge Graph and SPARQL for a knowledge-based CDSS is pretty straightforward and simple. Its efficiency depends on the underlying OIE method chosen for extracting knowledge. The system provides the automatic updating of the knowledgebase and in turn offers reliability to the CDSS. Being a centralized system, the fixed curation method of the knowledgebase will consistently improve its quality and make room for its usefulness in decision making processes.

However, there is a lot of improvement possibilities that can make the system much more efficient and robust as a perfect system. One of the improvements that can be worked into the system is to use ranked RDF triples which can serve in two ways. First of all, it can give weightage to the decision suggestion and secondly, it can help in removing redundant triple from the knowledgebase allowing the CDSS to work faster. The ranked triples can be evaluated using the precision and recall curves that can give a better appraisal of the system. Another approach to removing redundant knowledge is to formalize forgetting. Formalizing forgetting in knowledge graphs implies a method of removing either the redundant entities or the redundant relations. Entities may not exist without relations. So, by removing relations would mean new updated relations replacing old relations.

6. Conclusion

CDSS has been considered a very important system in the healthcare sector. That is the reason for the numerous studies that has been done on developing a perfect system that is highly efficient while at the same time reliable. Since the CDSS requires a very quick response to queries, a centralized system is to be considered. At the same time, the knowledgebase of such a system requires being maintained with constant and consistent updating from various sources of medical literature. Knowledge graphs have proved to be a very formidable approach to represent huge amount of knowledge that is now available in the web. RDF triples are reliable storage method for knowledge graphs in the form of RDF knowledge graphs. Curation of the RDF knowledge graph can be done through a QA system that converts natural language questions into SPARQL queries which when matched with RDF triples from an OIE process can enhance the knowledgebase. Therefore, a method is proposed to curate knowledge base of a CDSS using RDF Knowledge graph. It is possible to evaluate the system using precision and recall methods and give an appraisal of its efficiency in acquiring knowledge from various sources.

References

[1] B. Middleton, D. F. Sittig, A. Wright, Clinical Decision Support: a 25 Year Retrospective and a 25 Year Vision, Yearbook of Medical Infor- matics 25 (2016) S103–S116. URL: http://www.thieme-connect.de/ DOI/DOI?10.15265/IYS-2016-s034. doi:10.15265/IYS-2016-s034.

[2] J. A. Osheroff, J. M. Teich, B. Mid- dleton, E. B. Steen, A. Wright, D. E. Detmer, A Roadmap for National Action on Clinical Decision Support, Journal of the American Medical Informatics Association 14 (2007) 141–145. URL: https://academic.oup.com/ jamia/article-lookup/doi/10.1197/jamia. M2334. doi:10.1197/jamia.M2334.

[3] A. M. Shahsavarani, Abadi, E. A. Marz, M. H. Kalkhoran2, S. Jafari, S. Qaranli, Clinical Decision Support Systems (CDSSs): State of the art Review

of Literature, International Journal of Medical Reviews 2 (2015) 299–308. URL: http://www.ijmedrev.com/article{_}68717.html.

[4] R. T. Sutton, D. Pincock, D. C. Baumgart, D. C. Sadowski, R. N. Fedorak, K. I. Kroeker, An Overview of Clinical Decision Support Systems: Benefits, Risks, and Strategies for Success, npj Digital Medicine 3 (2020) 17. URL: http://www.nature.com/articles/s41746-020-0221-y. doi:10.1038/s41746-020-0221-y.

[5] R. A. Greenes, Definition, Scope, and Challenges, in: R. A. Greenes (Ed.), Clinical Decision Support : The Road to Broad Adoption, sec- ond ed., Elsevier, London, 2014, pp. 3–47. URL: https://linkinghub.elsevier.com/retrieve/pii/C20120003043. doi:10.1016/C2012-0-00304-3.

[6] E. S. Berner, T. J. La Lande, Overview of Clinical Decision Support Sys- tems, in: E. S. Berner (Ed.), Clinical decision support systems, third ed., Springer, Cham, Cham, 2016, pp. 1–17. URL: http://link.springer.com/10.1007/ 978-3-319-31913-1{_}1. doi:10.1007/978-3-319-31913-1_1.

[7] D. Graham, Introduction to Knowledge-Based Systems, in: Knowledge-Based Image Processing Systems, Springer London, London, 1997, pp. 3–13. URL: http://link.springer.com/10.1007/ 978-1-4471-0635-7{_}1. doi:10.1007/978-1-4471-0635-7_1.

[8] D. Dinevski, U. Bele, T. Sarenac, U. Rajkovic, O. Sustersic, Clinical Decision Support Systems, in: G. Graschew, S. Rakowsky (Eds.), Telemedicine Techniques and Applications, In- techOpen, 2011, pp. 185–210. URL: http://www.intechopen.com/books/telemedicine-techniques-and-applications/clinical-decision-support-systems. doi:10.5772/25399.

[9] G. P. Purcell, What makes a good clinical decision support sys- tem, BMJ (Clinical research ed.) 330 (2005) 740–741. URL: https://europepmc.org/articles/PMC555864. doi:10.1136/bmj.330.7494.740.

[10] R. A. Greenes, D. W. Bates, K. Kawamoto, B. Middleton, J. Osheroff, Y. Shahar, Clinical Decision Support Models and Frameworks: Seeking to Address Research Issues Underlying Implementation Successes and Failures, Journal of Biomedical Informatics 78 (2018) 134–143. URL: https://linkinghub.elsevier.com/retrieve/pii/S1532046417302757. doi:10.1016/j.jbi.2017.12.005.

[11] G. Kong, D.-L. Xu, J.-B. Yang, Clinical Decision Support Systems: A Review on Knowledge Representation and Infer- ence Under Uncertainties, International Journal of Computational Intelli- gence Systems 1(2008) 159–167. URL: https://doi.org/10.1080/18756891.2008.9727613. doi:10.1080/18756891.2008.9727613.

[12] A. Al-Badareen, M. Selamat, M. Samat, Y. Nazira, O. Akkanat, A Review on Clinical Decision Support Systems in Healthcare, in: Journal of Convergence Information Technology(JCIT), volume 9, 2014, pp. 125–135.

[13] M. A. Musen, J. van der Lei, Knowledge engineering for clinical consultation programs: modeling the application area., Methods of information in medicine 28 (1989) 28–35. URL: http://www.ncbi.nlm.nih.gov/pubmed/2649771.

[14] S. A. Spooner, Mathematical Foundations of Decision Support Systems, Springer International Publishing, Cham, 2016, pp. 19–43. URL: https://doi.org/10.1007/978-3-319-31913-1{_}2. doi:10.1007/978-3-319-31913-1_ 2.

[15] K. von Michalik, M. Kwiatkowska, K. Kielan, Application of Knowledge-Engineering Methods in Medical Knowledge Management, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 205–214. URL: https://doi.org/10.1007/978-3-642-36527-0{_}14. doi:10.1007/978-3-642-36527-0_ 14.

[16] M. Nickel, K. Murphy, V. Tresp, E. Gabrilovich, A Review of Relational Machine Learning for Knowl- edge Graphs, Proceedings of the IEEE 104 (2016) 11–33. URL: https://ieeexplore.ieee.org/document/7358050/. doi:10.1109/JPROC.2015.2483592.

[17] K. Balog, Meet the Data, in: Entity-Oriented Search, first ed., Springer, Cham, Cham, 2018, pp. 25–53. URL: http://link.springer.com/10.1007/978-3-319-93935-3{_}2. doi:10.1007/978-3-

319-93935-3_2.

[18] J. Yan, C. Wang, W. Cheng, M. Gao, A. Zhou, A retrospective of knowledge graphs, Frontiers of Computer Science 12 (2018) 55–74. URL: http://link.springer.com/10.1007/s11704-016-5228-9. doi:10.1007/s11704-016-5228-9.

[19] P. A. Bonatti, S. Decker, A. Polleres, V. Presutti, Knowledge Graphs: New Directions for Knowledge Representation on the Semantic Web, Dagstuhl Reports 8 (2019) 29–111. URL: https://drops. dagstuhl.de/opus/volltexte/2019/10328/. doi:10.4230/DagRep.8.9.29.

[20] R. Popping, Text Analysis for Knowledge Graphs, Qual- ity & Quantity 41 (2007) 691–709. URL: http://link.springer. com/10.1007/s11135-006-9020-z. doi:10.1007/s11135-006-9020-z.

[21] R. Popping, Knowledge Graphs and Network Text Analysis, Social Science Information 42 (2003) 91–106. URL: http://journals.sagepub.com/doi/10.1177/ 0539018403042001798. doi:10.1177/0539018403042001798.

[22] A. Hogan, E. Blomqvist, M. Cochez, C. D'Amato, G. de Melo, C. Gutier- rez, J. E. L. Gayo, S. Kirrane, S. Neu- maier, A. Polleres, R. Navigli, A. C. N. Ngomo, S. M. Rashid, A. Rula, L. Schmelzeisen, J. Sequeda, S. Staab, A. Zimmermann, Knowledge Graphs (2020). URL: http://arxiv.org/abs/2003. 02320. arXiv:2003.02320.

[23] S. Seufert, P. Ernst, S. J. Bedathur, S. K. Kondreddi, K. Berberich, G. Weikum, Instant Espresso: Interactive Analysis of Relationships in Knowledge Graphs, in: Proceedings of the 25th International Conference Companion on World Wide Web - WWW '16 Companion, ACM Press, New York, New York, USA, 2016, pp. 251–254. URL: http://dl.acm.org/ citation.cfm?doid=2872518.2890528. doi:10.1145/2872518.2890528.

[24] L. Ehrlinger, W. Wöß, Towards a Definition of Knowledge Graphs, in: M. Martin, M. Cuquet, E. Folmer (Eds.), Joint Proceedings of the Postersand De- mos Track of the 12th International Conference on Semantic Systems - SE-MANTiCS2016 and the 1st Internation-alWorkshop on Semantic Change & Evolving Semantics (SuCCESS'16) co-located with the 12th International Con-feren, Sun SITE Central Europe (CEUR), Leipzig, Germany, 2016.

[25] M. Färber, F. Bartscherer, C. Menne, A. Rettinger, Linked data quality of DBpedia, Freebase, OpenCyc, Wikidata, and YAGO, Semantic Web 9 (2018) 77– 129. doi:10.3233/SW-170275.

[26] D. Fensel, U. Şimşek, K. Angele, E. Hua-man, E. Kärle, O. Panasiuk, I. Toma, J. Umbrich, A. Wahler, How to Build a Knowledge Graph, in: Knowl- edge Graphs, Springer International Publishing, Cham, 2020, pp. 11–68. URL: http://link.springer.com/10.1007/ 978-3-030-37439-6{_}2. doi:10.1007/ 978-3-030-37439-6_2.

[27] E. W. Schneider, Course Modularization Applied: The Interface System and its Implications for Sequence Control and Data Analysis, Technical Report, Human Resources Research Organization (Hum-RRO), Virginia, 1973. URL: https://files. eric.ed.gov/fulltext/ED088424.pdf.

[28] S. S. Nurdiati, C. Hoede, 25 years de-velopment of knowledge graph theory: the results and the challenge, Technical Report, Department of Applied Mathematics, University of Twente, Enschede, 2008. URL: https://research.utwente.nl/en/publications

[29] D. Fensel, U. Şimşek, K. Angele, E. Hua-man, E. Kärle, O. Panasiuk, I. Toma, J. Umbrich, A. Wahler, Introduc- tion: What Is a Knowledge Graph?, in: Knowledge Graphs, Springer In-ternational Publishing, Cham, 2020, pp. 1–10. URL: http://link.springer.com/ 10.1007/978-3-030-37439-6{_}1. doi:10.1007/978-3-030-37439-6_1.

[30] A. Harth, S. Decker, Optimized In- dex Structures for Querying RDF from the Web, in: Proceedings of the Third Latin American Web Congress, LA-WEB '05, IEEE Computer Society, USA, 2005, p. 71. URL:https://doi.org/10.1109/LAWEB.20 05.25. doi:10.1109/LAWEB.2005.25.

[31] B. Villazon-Terrazas, N. Garcia-Santa, Y. Ren, A. Faraotti, H. Wu, Y. Zhao, G. Vetere, J. Z. Pan, Knowledge Graph Foundations, Springer Inter- national Publishing, Cham, 2017, pp. 17–55. URL: https://doi.org/10.1007/ 978-3-319-

45654-6{_}2. doi:10.1007/ 978-3-319-45654-6_2.

[32] W. Zheng, L. Zou, W. Peng, X. Yan, S. Song, D. Zhao, Semantic SPARQL Similarity Search over RDF Knowl- edge Graphs, Proc. VLDB Endow. 9 (2016) 840–851. URL: https://doi.org/10. 14778/2983200.2983201. doi:10.14778/ 2983200.2983201.

[33] T. Yang, J. Chen, X. Wang, Y. Chen, X. Du, Efficient SPARQL Query Evaluation via Automatic Data Partitioning, in: W. Meng, L. Feng, S. Bressan, W. Winiwarter, W. Song (Eds.), Database Systems for Advanced Applications DASFAA 2013, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 244–258.

[34] O. Curé, G. Blin,RDF and the Semantic Web Stack, in: O. Curé, G. Blin (Eds.), RDF Database Sys- tems, first ed., Elsevier, Waltham, MA, 2015, pp. 41–80.URL:https://linkinghub.elsevier.com/retrieve/pii/ B9780127999579000031. doi:10.1016/ B978-0-12-799957-9.00003-1.

[35] K. Hose, R. Schenkel, RDF Stores, in: L. Liu, M. T. Özsu (Eds.), Encyclopedia of Database Systems, Springer New York, New York, NY, 2017, pp. 3100– 3106. URL:http://link.springer.com/10.1007/978-1-899-7993-3{_}80676-1. doi:10.1007/978-1-4899-7993-3_ 80676-1.

[36] F. Du, Y. Chen, X. Du, Partitioned Indexes for Entity Search over RDF Knowledge Bases, in: S.-g. Lee, Z. Peng, X. Zhou, Y.-S. Moon, R. Unland, J. Yoo (Eds.), Database Systems for Advanced Applications DASFAA 2012, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012, pp. 141–155.

[37] A. Mohamed, G. Abuoda, A. Ghanem, Z. Kaoudi, A. Aboulnaga, RDF- Frames: Knowledge Graph Access for Machine Learning Tools (2020). URL: http://arxiv.org/abs/2002.03614. arXiv:2002.03614.

[38] R. Denaux, Y. Ren, B. Villazon-Terrazas, P. Alexopoulos, A. Faraotti, H. Wu, Knowledge Architecture for Organisations, Springer International Publishing, Cham, 2017, pp. 57–84. URL: https://doi.org/10.1007/978-3-319-45654-6{_}3. doi:10.1007/978-3-319-45654-6_ 3.

[39] D. C. Faye, O. Curé, G. Blin, A sur- vey of RDF storage approaches, Revue Africaine de la Recherche en Informa- tique et Math{é}matiques Appliqu{é}es 15 (2012) 11–35. URL: https://hal.inria.fr/ hal-01299496.

[40] J. Sleeman, T. W. Finin, A. Joshi, Topic Modeling for RDF Graphs, in: A. L. Gentile, Z. Zhang, C. D'Amato, H. Paulheim (Eds.), Proceedings of the Third International Workshop on Linked Data for Information Extraction (LD4IE2015), CEUR Workshop Proceedings (CEUR-WS.org), Bethlehem, Pennsylvania, USA, 2015, pp. 48–62.

[41] S. Pouriyeh, M. Allahyari, K. Kochut, G. Cheng, H. R. Arabnia, Combin- ing Word Embedding and Knowledge- Based Topic Modeling for Entity Summarization, in: 2018 IEEE 12th International Conference on Semantic Computing (ICSC), Institute of Electrical and Electronics Engineers (IEEE), Laguna Hills, CA, USA, 2018, pp. 252–255. doi:10.1109/ICSC.2018.00044.

[42] S. Pouriyeh, M. Allahyari, K. Kochut, G. Cheng, H. R. Arabnia, ES-LDA: En- tity Summarization using Knowledge- based Topic Modeling, in: G. Kondrak, T. Watanab (Eds.), Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Asian Federation of Natu- ral Language Processing, Taipei, Taiwan, 2017, pp. 316–325. URL: https://www. aclweb.org/anthology/I17-1032.pdf.

[43] Q. Liu, G. Cheng, K. Gunaratna, Y. Qu, Entity Summarization: State of the Art and Future Challenges, ArXiv abs/1910.0 (2019). URL: http://arxiv.org/abs/1910. 08252. arXiv:1910.08252.

[44] O. Kolomiyets, M.F.Moens, A Survey on Question Answering Technology from an Information Retrieval Perspective, Information Sciences 181 (2011) 5412–5434.URL:http://www.sciencedirect.com/science/article/pii/S0020025511003860. doi:https://doi.org/10.1016/j.ins.2011.07.047.

[45] E. M. Nabil Alkholy, M. Hassan Haggag, A. Aboutabl, Question Answering Systems: Analysis and Survey, International Journal of Computer Science & Engineering Survey 09 (2018) 1–13. URL:http://aircconline.com/ijcses/V9N6/9618ijcses01.pdf.

doi:10.5121/ijcses.2018.9601.
[46] S. K. Dwivedi, V. Singh, Research and Reviews in Question Answering System, Procedia Technology 10 (2013) 417–424. URL:https://linkinghub.elsevier.com/retrieve/pii/S2212017313005409. doi:10.1016/j.protcy.2013.12.378.
[47] E. Dimitrakis, K. Sgontzos, Y. Tzitzikas, A Survey on Question Answering Systems Over Linked Data and Documents, Journal of Intelligent Information Systems 55 (2020) 233–259. URL: https://doi.org/10.1007/s10844-019-0584-7. doi:10.1007/s10844-019-00584-7.
[48] A. M. N. Allam, M. H. Haggag, The Question Answering Systems: A Survey, International Journal of Re- search and Reviews in Information Sciences (IJRRIS) 2 (2012) 211–221. URL: https://pdfs.semanticscholar.org/b294/4a85b9cb428a28e30bdd236471e712667e91.pdf.
[49] D. Diefenbach, V. Lopez, K. Singh, P. Maret, Core Techniques of Question Answering Systems over Knowledge Bases: A Survey, Knowledge and Information Systems 55 (2018) 529–569.URL: https://doi.org/10.1007/s10115-017-1100-y. doi:10.1007/s10115-017-1100-y.
[50] J. Pérez, M. Arenas, C. Gutierrez, Semantics and Complexity of SPARQL, in: I. Cruz, S. Decker, D. Allemang, C. Preist, D. Schwabe, P. Mika, M. Uschold, L. M. Aroyo (Eds.), The Semantic Web - ISWC 2006, Springer Berlin Heidelberg, Berlin, Heidelberg, 2006, pp. 30–43.
[51] R. Angles, C. Gutierrez, The Expressive Power of SPARQL, in: A. Sheth, S. Staab, M. Dean, M. Paolucci, D. Maynard, T. Finin, K. Thirunarayan (Eds.), The Semantic Web - ISWC 2008, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, pp. 114–129.
[52] P. Grafkin, M. Mironov, M. Fellmann, B. Lantow, K. Sandkuhl, A. V. Smirnov, SPARQL Query Builders: Overview and Comparison, in: B. Johansson, F. Vencovský (Eds.), Joint Proceedings of the {BIR} 2016 Workshops and Doc- toral Consortium co-located with 15th International Conference on Perspec- tives in Business Informatics Research {(BIR} 2016), Prague, Czech Republic, September 14 - 16, 2016, volume 1684 of {CEUR} Workshop Proceedings, CEUR-WS.org, 2016, p. 12. URL: http://ceur-ws.org/Vol-1684/paper21.pdf.
[53] K. Höffner, S. Walter, E. Marx, R. Usbeck, J. Lehmann, A.-C. N. Ngomo, Survey on Challenges of Ques- tion Answering in the Semantic Web, Semantic Web 8 (2017) 895–920. URL:https://content.iospress.com/articles/semantic-web/sw247. doi:10.3233/SW-160247.
[54] Axel-Cyrille, N. Ngomo, M. Hoffmann, R. Usbeck, K. Jha, Holistic and Scal- able Ranking of RDF Data, in: 2017 IEEE International Conference on Big Data (Big Data), IEEE, Boston, MA, 2017, pp. 746–755. doi:10.1109/BigData.2017.8257990.
[55] S. Elbassuoni, M. Ramanath, R. Schenkel, M. Sydow, G. Weikum, Language-Model-Based Ranking for Queries on RDF-Graphs, in: Proceedings of the 18th ACM Conference on Infor- mation and Knowledge Management, CIKM ’09, Association for Comput- ing Machinery, New York, NY, USA, 2009, pp. 977–986. URL:https://doi.org/10.1145/1645953.1646078. doi:10.1145/ 1645953.1646078.
[56] A. Buikstra, H. Neth, L. Schooler, A. Ten Teije, F. Van Harmelen, Ranking Query Results from Linked Open Data Using a Simple Cognitive Heuristic, in: Proceedings of the 2011 International Confer- ence on Discovering Meaning On the Go in Large Heterogeneous Data, LHD’11, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2011, pp. 55–60.
[57] M. Yahya, D. Barbosa, K. Berberich, Q. Wang, G. Weikum, Relationship Queries on Extended Knowledge Graphs, in: Proceedings of the Ninth ACM International Conference on Web Search and Data Mining, WSDM ’16, Association for Com- puting Machinery, New York, NY, USA, 2016, pp. 605–614. URL: https://doi.org/10.1145/2835776.2835795. doi:10.1145/2835776.2835795.
[58] Q. Wang, Z. Mao, B. Wang, L. Guo, Knowledge Graph Embedding: A Survey of Approaches and Ap- plications, IEEE Transactions on Knowledge and Data Engineering 29 (2017) 2724–2743. URL: http://ieeexplore.ieee.org/document/8047276/. doi:10.1109/TKDE.2017.2754499.
[59] G. A. Gesese, R. Biswas, M. Alam,

H. Sack, A Survey on Knowledge Graph Embeddings with Literals: Which model links better Literal-ly?, ArXiv abs/1910.1 (2019).

[60] P. Bonatti, S. Decker, A. Polleres, V. Presutti, Knowledge Graphs: New Directions for Knowledge Representation on the Semantic Web (Dagstuhl Seminar 18371), Dagstuhl Reports 8

(2018) 106–110. URL: https://drops.dagstuhl.de/opus/volltexte/2019/10328/. doi:10.4230/DagRep.8.9.29.

[61] K. M. Endris, M.-E. Vidal, D. Graux, Federated Query Processing, Springer International Publishing, Cham, 2020, pp. 73–86. URL: https://doi.org/10.1007/ 978-3-030-53199-7{_}5. doi:10.1007/978-3-030-53199-7_5.

[62] M. Saleem, Y. Khan, A. Hasnain, I. Ermilov, A.-C. N. Ngomo, A Fine-grained Evaluation of SPARQL Endpoint Federation Systems, Semantic Web 7 (2016) 493–518.

[63] J. Tang, M. Hong, D. L. Zhang, J. Li, Information Extraction: Methodologies and Applications, in: H. A. do Prad, E. Ferneda (Eds.), Emerging Technolo- gies of Text Mining, IGI Global, 2008, pp. 1–33. URL: http://services.igi-global.com/resolvedoi/resolve.aspx?doi=10.4018 /978-1-59904-373-9.ch001. doi:10.4018/978-1-59904-373-9.ch001.

[64] O. Etzioni, M. Banko, S. Soderland, D. S. Weld, Open Information Extrac- tion from the Web, Commun. ACM 51 (2008) 68–74. URL: https://doi.org/ 10.1145/1409360.1409378. doi:10.1145/ 1409360.1409378.

[65] S. Ali, H. Mousa, M. Hussien, A Review of Open Information Extrac- tion Techniques, IJCI. International Journal of Computers and Infor- mation 6 (2019) 20–28. URL:https://ijci.journals.ekb.eg/article{_} 35099. htmlhttps://ijci.journals.ekb.eg/ article{_}35099{_}2751a97dec8ca23f3e6 ca98f27cee4b6.pdf. doi:10.21608/ijci.2019.35099.

[66] O. Etzioni, A. Fader, J. Christensen, S. Soderland, M. Mausam, Open Information Extraction: The Second Generation, in: Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Vol- ume One, IJCAI'11, AAAI Press, 2011, pp. 3–10.

[67] Mausam, Open Information Extraction Systems and Downstream Applications, in: Proceedings of the Twenty-Fifth International Joint Conference on Artifi- cial Intelligence, IJCAI'16, AAAI Press, 2016, pp. 4074–4077.

[68] A. Zhila, E. Yagunova, O. Makarova, Bringing The Output of Open Information Extraction to The RDF / XML Format : A Case Study, 2015.

[69] G. Angeli, M. J. J. Premkumar, C. D. Manning, Leveraging Linguistic Structure For Open Domain Information Extraction, in: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Nat- ural Language Processing of the Asian Federation of Natural Language Processing, {ACL} 2015, July 26-31, 2015, Beijing, The Association for Computer Linguis- tics, 2015, pp. 344–354. URL: https://doi. org/10.3115/v1/p15-1034. doi:10.3115/ v1/p15-1034.

[70] I. Sim, P. Gorman, R. A. Greenes, R. B. Haynes, B. Kaplan, H. Lehmann, P. C. Tang, Clinical decision sup- port systems for the practice of evidence-based medicine, Journal of the American Medical Informatics Association: JAMIA 8(2001)527–534. URL:https://pubmed.ncbi.nlm. nih.gov/11687560https://www.ncbi. nlm.nih.gov/pmc/articles/PMC130063/. doi:10.1136/jamia.2001.0080527.