Disease Ontology Application: An Exploration of Workflows, User Interfaces and The Future of Real-Time Disease Ontologies

Arunit Gupta, Luke Joseph McDuff, Manoj Prabhakar Ejjirothu, Marmikkumar Navinchandra Patel

University of Missouri- Kansas City

***Abstract*- Knowledge of health and disease information is a vast domain where knowledge is typically highly specified into a particular field or area. In the future, intelligent systems will provide information about semantic data that will help workers in the health and disease domain process and infer information from the vast quantities of data available in other domains. Our research explores this theme by focusing on creating an intelligent question and answering system that is built around the core of health and disease information. The main contributions of this research is in the development of our NLP workflow for building a disease ontology and developing an accessible user application. In this paper we report on our experience in building a question and answering system for our disease ontology. We describe our development process and our implementation, evaluate our results and suggest areas of our implementation that could be improved in future research.**

1. INTRODUCTION

Imagine an outbreak in the United States of a particular strain of flu occurs. The disease spreads quickly causing large problems for a huge majority of the population. Scientists from all domains quickly begin to mobilize, performing research into the disease and the effects on people in different areas. However, in this scenario scientists must conduct their research in a traditional sense, combining through vast quantities of data, slowly analyzing and understanding the situation. Scouring vast quantities of documentation is inefficient and wastes valuable time in a situation that demands time sensitivity. For situations such as these, having access to a disease ontology that is accessible could help scientists and other researchers perform rapid research into particular diseases they may not be familiar with. While previous work by [2] has helped contribute to the growth in the field of disease ontologies, the ability for users to easily query and approach health information has been lacking. Our research focus to specifically fill the gap of user accessibility to disease ontologies. Also, previous research has not taken advantage of the new technologies for creating ontologies available in the Spark libraries, so we define a new workflow based on those innovative and accessible technologies.

The foundational work by [2] has provided the research field of disease ontology with a model of performance, providing a mapping of over 8,000 diseases. This disease ontology has led to other various methods such as [4, 5] that have expanded on disease ontologies to incorporate them with gene based ontologies. Gene study has proven to be one of the largest areas within the research of disease ontologies. Specifically, researchers are interested in the human genome [3, 8] while other researchers are interested in understanding the animal genome [6]. The field of disease ontologies has also expanded out to focus in on narrower fields of research.

Primary example of this is the work done by [9] who specifies in Parkinson's disease ontology. Another example of specification is [11] who has developed an ontology on Polycystic Ovary syndrome. Various disease ontology researchers have tried to piece together different ontologies to create a unified knowledge base to benefit particular applications. Specifically, ontologies have been merged to attempt to create knowledge

bases that can support clinic operations [13] [15]. Our work is different from the previous work completed in the field, by the fact that we emphasize a precise NLP workflow of building the ontology and incorporate the ability for an easy to use user interface. All reviewed previous work has focused solely on the design of the ontology alone.

Section 2 details our proposed solution in describing the details of our approach to building the QA system. Section 3 describes the implementation of our system focusing on initial research, building the ontology and engineering the user application. Section 4 provides insight to our initial results and critiques of our current work. Finally, sections 6 and 7 offer our conclusive feedback and suggest the direction and insight of the future direction of research.

2. PROPOSED SOLUTION

Online articles, blogs, Wikipedia, twitter tweets were selected as online corpus for the system.

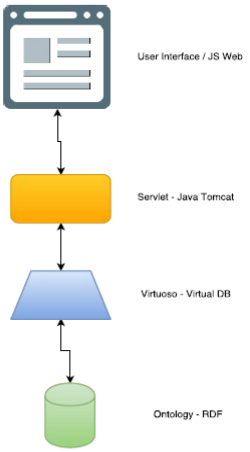

Figure 1: System Architecture

questions in natural language [16] were constructed in sentences format based on knowledge base. System is designed in a way when user needs information about a specific disease, symptoms and precaution etc. through designed interface. System is designed to work on datasets from wide Virtual DB which is a modern range of sources across the web or stored unstructured data, which could be processed to answer a question. First step includes the processing of our unstructured big data to a structured form which we chose as a RDF (Resource Description framework) [17] which gives a method for modelling this information using various syntax and notations. These syntaxes are known triples which models sentences in form of subject-predicate-object which further are used to build an ontology.

Architecture

Our Architecture is shown in figure 1. Our objective is to create a system which can understand question and answer it after applying techniques. System architecture works on building an ontology upon RDF, for storing this ontology we need a database which was used as Virtuoso-

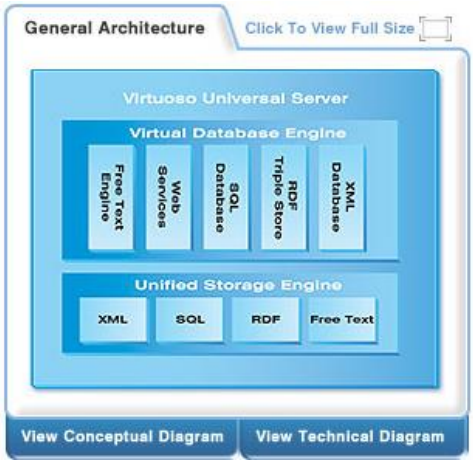

Figure 2: Virtuoso

enterprise grade solution for data access, integration and relational database management which are like RDF based Property/predicate graphs and SQL tables. It has a Unique architecture which offer distinct server functionality such as Relational tables management, content management, web application server.

Five star linked open data deployment, web file services and graph data management. The services defined run on tomcat server. User are provided with user interface developed using Ionic which allows end user to use the application through mobile or web site.

Domain Research

Domain chosen was specific to Healthcare which was narrowed down to diseases in terms of implementation which could be spanned over broader domain in future. In terms of dataset which we chose to have big data which can come from various sources of information. It could be raw information from text files, health blogs and articles, newsletters, Wikipedia, twitter streaming data, social networking sites where people post about recent activities. This particular project was implemented on datasets which were collected from health blogs like center for disease control [18] which has detailed information about disease, symptoms, prevention and statistics; also data was taken from Wikipedia, news articles, web data base and tweets collected from twitter API which provides dynamic data which gets updated quite frequently. All these datasets are further used as a knowledge base where all natural language techniques to convert all unstructured data and semi structured data to structured data from corpus which is tokenized, lemmatized, topic discovery and go through machine learning algorithm which is used to create ontology based on RDF graphs which can be used to query and retrieve meaningful answers.

3. IMPLEMENTATION

Ontology-Development

The Scope of the Topics was narrowed to particular categories in order to build a feasible ontology which can be easily visualized. The Topics covered in this Ontology were different Disease Types (Viral, Physical, cancer). Each category further had three different topics as Symptoms, effected Body parts and Causes for those diseases. A source Ontology was created which depicts the above mentioned structure (through JAVA). The below Figure 3 Shows the built ontology for reference. The actual Ontology for the training the model was built using the NLP pipe line method. The data was initially highly filtered in order to remove the junk from the documents. Then through the NLP methods (Tokenization, lemmatization, Name Entity Relation) the data was completely categorized into words. The TF –IDF words generated were mostly belonged to categories like Disease and Body parts as expected. As the Datasets were huge the execution of these methods on these datasets took a considerable amount of time. Triplets were also generated using the OPEN – IE from the entire Dataset. The triplets generated included the subject predicate and object. The triplets were then combined through the Name Entity. Relations found through the NLP Method. Protégé and Web owl were used for the visualization of the ontology. Now the ontology was generated based on the type of the queries which were designed to run on this ontology.

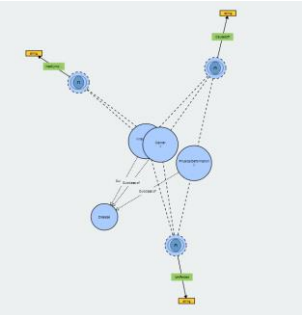

Figure: 3 Reference-Ontology

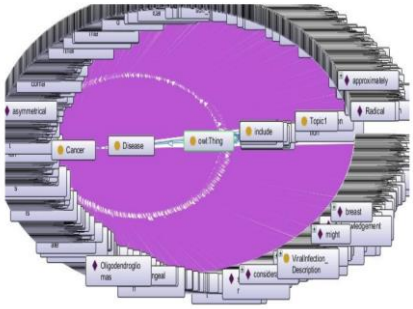

Figure:4 Complete Ontology

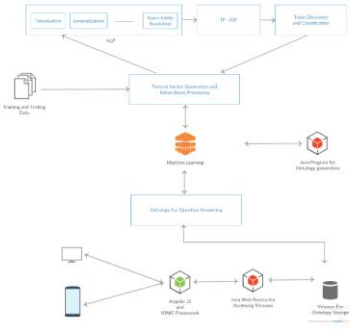

Figure: 5 Workflow

The SQWRL queries were designed in order to limit the scope of the ontology and to make the execution of the queries on the ontology more feasible. The SQWRL queries were designed based of the format like “what “, ”which” and the subjects like “types of”, “body parts affected” and “Symptoms of”. The predicate was chosen as the answers for the built queries. The ontology properties were defined based on the above mentioned subjects and all the classes were related through these properties.

The Work flow diagram (Figure 5) which was shows the Feature Vector Generation, where the topics were generated based on the different categories like diseases, causes and body parts were give as the topics based on which the topics were classified and the feature Vector was generated. Through the Naïve Bayes approach, 76% accuracy was achieved for the classification of the topics. **User Interface** The user Interface was built using the Virtuoso, Angular JS and the Java web services.

The Virtuoso was used to run the queries and extract the results which are accessed through the Java Web services using Virtuoso API. Through the API, it is easy to access virtuoso, run the SQWRL queries on any ontology and extract the results for any applications. The ontology can be stored in the Virtuoso In the form of OWL file. The front end is developed using the Angular JS. The below figure shows the sample User Interface. Once the user Selects respective options and submits for the results, the respective query generated is displayed on the screen. The Query is run on the background in Virtuoso and the results are sent back through the web service and displayed on the screen. The entire application is deployed in any public platform based services and it runs on the local machine.

For access to current implementation (Git) and more info regarding implementation see [19] [20].

4. RESULT AND EVALUATION

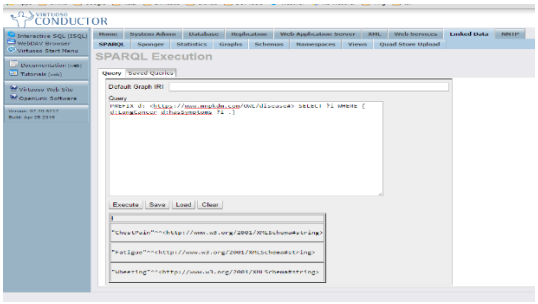

Figure:6 Query execution in Virtuoso

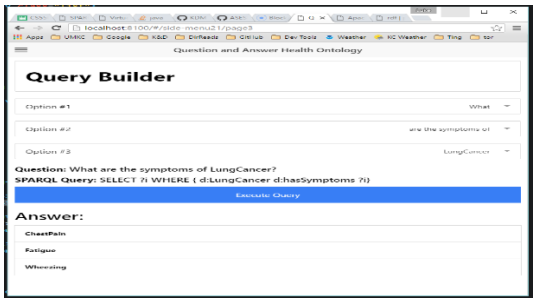

Figure:7 User Interface

Platform: Intellij

Programming Language: Scala (For Generating Ontology), Java (For Developing Web Service)

Ontology Language: OWL (Web Ontology Language)

Database: Virtuoso

Framework: ionic (For generating mobile and web application)

Testing and Training Model: Naïve Bayes

Preprocessing on Data: coreNLP, TF-IDF, Topic Discovery and Feature Vector

After training and testing of model for generating ontology, we have tried to evaluate the generated ontology using several SPARQL queries. The tools and technologies used for evaluation is summarize above.

We have used predefined format for asking question, so that we can improve the user experience and also results which are extracted are relevant to query. We have used two types of question what and which. We have prepared several question which are related to data and for which answers can be inferred from the ontology. Some of the example questions are as follow:

What are the symptoms of <name of Disease>?

Which body parts are affected by <name of Disease>?

What are the types of <name of Disease>?

We have prepared SPARQL queries which provides expected results for the above question. Queries used for extracting the result from the ontology are given below.

SELECT ?i where { d:LungCancer d:hasSymptoms ?i} (for symptoms of lung cancer)

SELECT ?i where { d:LungCancer d:isAffected ?i } (for affected part by lung cancer)

SELECT ?i where { ?i rdf:type d:Cancer } (for types of cancer)

The results for one of this query in protégé is shown in figures. Figure 6 gives the result on protégé.

We have also tried to show the query result in web application using ionic framework so the user can have nice and interactive interface for question and answering. Figure 7 provides the screenshot of interface. From Figure 7 we can see that, it is pretty convenient for the user to prepared the query and we can accumulate more queries in future.

To evaluate the accuracy of our classification, we have used Naïve Bayes classification algorithm and we have found accuracy of 76%. We also tried to evaluate the performance based on the running time. The timing for the preprocessing of data is about 56ms which includes the applying the operations like tokenization, lemmatization, name-entity resolution, TF-IDF and topic discovery. This all operations are applied in order to increase the accuracy of operations. While training time for LDA model is about 375 seconds and the training set size is nearly 5400 documents.

3. CONCLUSION

From the evaluation and result, it is clear that the model we have used is good in accuracy, but we have tried it on relatively small dataset. The accuracy may decrease with respect to size of the datasets and there are still lot of optimization and accuracy operations which can be applied on the model. Some these operations are word2vec to

find the similar words from the documents and openIE to match the synonyms from the large external dataset. Our approach is still semi-automatic where we have to define the partial structure of ontology, so we can use to machine learning to make it completely automatic. So, in terms of accuracy and user experience, application developed by us perform quite well but there still long way to make it accurate and automatic.

4. FUTURE WORK

Future research would seek to improve a variety of different aspects of the current system. Beginning with the ontology we would like to spend considerable more time researching the domain and understanding more how current disease ontologies are built. Learning the relations and the schema would be very useful in improving the overall quality of our own ontology. After reviewing current design in greater detail having our designs reviewed by an expert would provide even more refinement to the process and could enable a better ontology. Once the design of the ontology was improved then we begin training it with a large quantity of disease related health data. However, our group would be particularly interested in using real-time information that could be mined from Twitter tweets. This information would provide a very interesting real time component of the semantic discussion around diseases. In particular, it could provide relevant real time information of disease outbreak and epidemics in cities across the world. In today's world where information is very time sensitive it would be interesting to generate new understandings of diseases based off this vast pool of unstructured semantic data. This is another area in the research on disease ontologies that is currently not being explored by the field of research. Our initial research into the field of study has not proven that other researchers are taking this unique approach, which would provide a new method in the development and use of disease ontologies.

Another important aspect of our design that was missing was the ability for users to specify their own natural language queries to the ontology. This feature would allow the user to extract the most meaningful and personal information to them. However, implementation would mean an extensive increase in time spent training and building the ontology to allow for greater exploration of the knowledge base as well as performing natural language processing on the query itself. And finally, the last improvement to the UI would be the ability to display answers from the ontology in more than just natural language answers. For example, using Twitter data could provide use of the Twitter meta-data that contains information about the tweet geographic location. When dealing with tweets about outbreaks, knowing the locations of the tweets could allow for some very interesting information to be built and generated from the data. In particular, creating a variety of visual data representation such as heat maps, scatter plots or graphs based on semantic data could create a new way of understating data. We would like to build and develop this semantic data representation engine to be used with our ontologies and possibly be incorporated into the NLP pipeline.

ACKNOWLEDGEMENTS

Authors of this paper would like to thanks Dr. Yugyung Lee for valuable suggestions and comments to improve this paper, CSEE Department, and University of Missouri – Kansas City for providing opportunity to learn this course.

REFERENES

[1] Jensen M, Cox AP, Chaudhry N, et al. The neurological disease ontology. *Journal of biomedical semantics*. 2013; 4:42-42.

[2] Schriml LM, Arze C, Nadendla S, et al. Disease Ontology: a backbone for disease semantic integration. *Nucleic Acids Research*. 2012; 40: D940-D946.

[3] Osborne JD, Flatow J, Holko M, et al. Annotating the human genome with Disease Ontology. *BMC genomics*. 2009;10 Suppl 1: S6-S6.

[4] Du P, Feng G, Flatow J, et al. From disease ontology to disease-ontology lite: statistical methods to adapt a general-purpose ontology for the test of gene-ontology associations. *Bioinformatics*. 2009; 25: i63-i68.

[5] Kibbe WA, Arze C, Felix V, et al. Disease Ontology 2015 update: an expanded and updated database of human diseases for linking biomedical knowledge through disease data. *Nucleic Acids Research*. 2015; 43: D1071-D1078.

[6] Hayman GT, Laulederkind SJF, Smith JR, et al. The Disease Portals, disease-gene annotation and the RGD disease ontology at the Rat Genome Database. *Database: the journal of biological databases and curation*. 2016; 2016.

[7] Adams N, Hoehndorf R, Gkoutos GV, Hansen G, Hennig C. PIDO: the primary immunodeficiency disease ontology. *Bioinformatics*. 2011; 27:3193-3199.

[8] Davis AP, Wiegers TC, King BL, et al. Generating Gene Ontology-Disease Inferences to Explore Mechanisms of Human Disease at the Comparative Toxicogenomics Database. *PLoS One*. 2016;11: e0155530.

[9] Younesi E, Malhotra A, Gündel M, et al. PDON: Parkinson's disease ontology for representation and modeling of the Parkinson's disease knowledge domain. *Theoretical biology & medical modelling*. 2015; 12:20.

[10] Lin Y, Xiang Z, He Y. Brucellosis Ontology (IDOBRU) as an extension of the Infectious Disease Ontology. *Journal of biomedical semantics*. 2011; 2:9-9.

[11] Joseph S, Barai RS, Bhujbalrao R, Idicula-Thomas S. PCOSKB: A KnowledgeBase on genes, diseases, ontology terms and biochemical pathways associated with PolyCystic Ovary Syndrome. *Nucleic acids research*. 2016;2015;44: D1032.

[12] Thirugnanam M, Ramaiah M, Pattabiraman V, Sivakumar R. Ontology Based Disease Information System. *Procedia Engineering*. 2012; 38:3235-3241.

[13] Pârvan A. Monistic dualism and the body electric: An ontology of disease, patient and clinician for person-centred healthcare. *Journal of Evaluation in Clinical Practice*. 2016; 22:530-538.

[14] Refolo LM, Snyder H, Liggins C, et al. Common Alzheimer's Disease Research Ontology: National Institute on Aging and Alzheimer's Association collaborative project. *Alzheimer's & dementia: the journal of the Alzheimer's Association*. 2012; 8:372.

[15] Timmermans S, Buchbinder M. Expanded newborn screening: articulating the ontology of diseases with bridging work in the clinic. *Sociology of Health & Illness*. 2012; 34:208220.

[16]http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3825096/

[17]https://en.wikipedia.org/wiki/Resource_Description_Framework

[18]https://www.cdc.gov/

[19]https://github.com/ljm7b2/KDM_Summer_2016_GroupX

[20] https://youtu.be/jCC86gkk_6Q