Building Natural Language Query System on Unstructured Data Using Knowledge Graph

Data Science Institute
COLUMBIA UNIVERSITY

Alex Kita, Arnav Saxena, Hyo Won (Elin) Kim, Ridwan Olawin, Shashwat Singh

Data Science Capstone Project with Accenture

Problem Statement

Knowledge based industries such as consulting and finance are heavily dependent on extracting information from unstructured financial text data for generating value. However, they mostly rely on manually parsing through this data to achieve the goal which clearly is inefficient given the ever increasing volume of such data. To deal with this problem, our team worked on building a knowledge graph (KG) based question answering system focused on cryptocurrency domain. Professionals can simply query this KG in natural language and get their answers easily.

End-to-End Solution

We gathered 208 documents (research papers) and, through pre-processing, broke each down into cleaned sentences. We then feed the sentences to the models and post-process. The outputs are triplets (e.g., [subject, verb, object]), and the post-process filters out unclean outputs. Then, the cleaned triplets are fed into the ontology building which filter out triplets that are not related to the specific domain. We load the final triplets to the Knowledge Graph Database. Finally, the user can input query in natural language through the interface which will automatically convert the query to SPARQL in the background and will fetch results from the database.

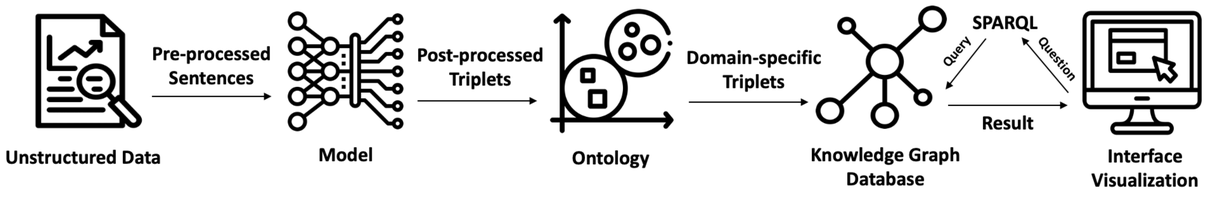


Figure 1: End-to-End Solution Workflow

Models Explored

We mainly explored four triplet extraction models namely - AllenNLP, StanfordOpenIE, KnowGL, and REBEL. We tuned these models and tested them against the custom ground truth triplets as expected in our case. The best models considering all the metrics were AllenNLP and StanfordOpenIE as shown in Table 1.

Models	Avg Precision	Avg Recall	F1 score
KnowGL	0.18	0.07	0.1
REBEL	0.29	0.21	0.22
Stanford OpenIE	0.29	0.29	0.25
AllenNLP	0.55	0.5	0.49

Table 1 : Comparison of different models for Triplet Extraction

Thus, we finally selected the outputs of these two models for Ontology generation and building Knowledge Graphs. Therefore, we have two Knowledge Graphs, one associated with the outputs from StanfordOpenIE and another one associated with the outputs of AllenNLP.

Ontology Generation

We generated both the relation ontology and entity ontology in a semi-automatic format which can be used to develop an ontology for different domains given different key terms. We used triplets extracted from the models as the input and passed them through FinBert Embeddings to generate embeddings for every unique relation and entity. We then applied Agglomerative Hierarchical Clustering on these embeddings to identify and group similar relations/entities. We also cross-referenced the clusters with common domain-specific financial relationships and entities along with NER to get domain-specific ontology. The flowchart for this process is shown below.

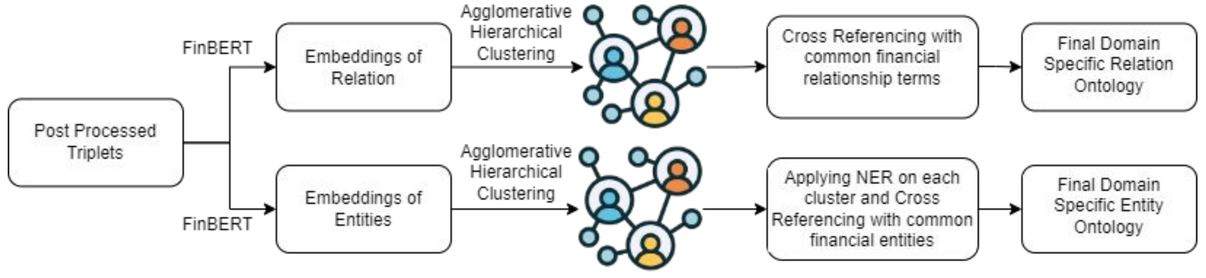
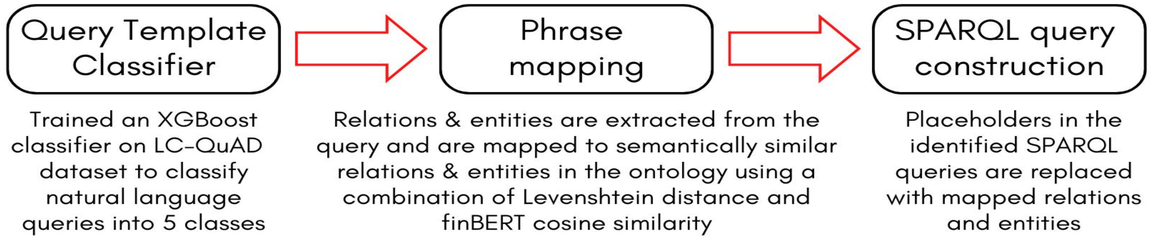


Figure 2: Semi- Automatic Ontology Generation

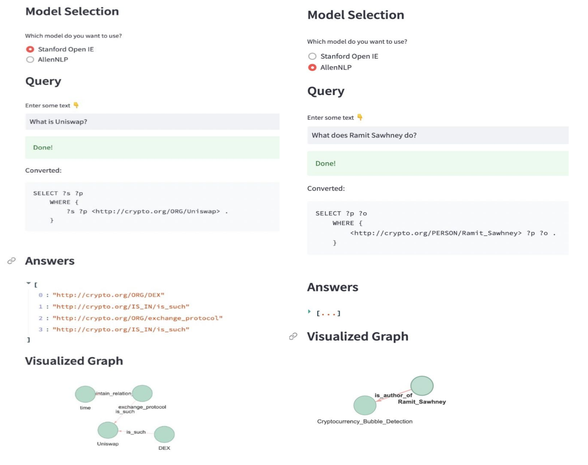
Natural Language to SPARQL query conversion



Results

As output of ontology, 43 relation classes are identified for Stanford OpenIE and 19 for AllenNLP. Also, 14 entity classes are identified for the former and 8 for the latter. Every input text is mapped to those classes in each model.

Through the interface (as shown right), user can input query in natural language and results will be displayed and visualized.



Acknowledgments

We would like to express our greatest gratitude for Mr. Satish Banka and Mr. Rajkumar Subramanian at Accenture for their guidance throughout the entire process. Also, we thank Professor Sining Chen and Aayush Verma for their support throughout.