Knowledge Engineering report

# Introduction

In this assignment we aimed to combine two different datasets to produce a set of queries that make use of data from both datasets when outputting their results. The first dataset we made use of is the GeoHive dataset; a linked data dataset containing Irish geospatial information such as boundary data per-county, city and even by parishes.

The second dataset is the “Crimes at Garda Stations Level 2010-2016” dataset, a CSV formatted dataset specifying a list of different Garda-Stations, their Divisions and a set of different crimes that occurred within that station’s jurisdiction over the past year.

<https://data.gov.ie/dataset/crimes-at-garda-stations-level-2010-2016>

# Approach to Ontology Modelling

The chosen dataset contains information about all the different Garda-Stations in Ireland, the name in both English and Gaelic, the Division that the station falls under, a list of different crimes committed within that stations jurisdiction within a time range and even the geographical Cartesian coordinates for each Station. This location information per station will then be used to link our dataset to GeoHive, by determining in which county each station is.

## Assumptions Made

Due to the large volume of data within our dataset, we decided to cut down the statistics across the years, focusing instead on just the 2015 data – the most recent year with complete statistics across the whole year.

Ontology modelling assumptions;

1. A County can have many stations
2. A County contains geospatial data
3. A Station can have multiple different crime counts
4. A Station has a location – X & Y coordinates

## References to sources used/reused

When designing the ontology, the documentation for both Protégé (<https://protegewiki.stanford.edu/wiki/ProtegeDesktopUserDocs>) and Jena (<https://jena.apache.org/documentation/>) were used.

Before implementation, Stanford University Protégé tutorials were also used to fully understand the theory behind ontology modelling. (<https://protege.stanford.edu/publications/ontology_development/ontology101.pdf>)

When picking the ontologies, we were inspired by the works done by the Maynooth University, providing a GUI that allows users to interact with a map of the Republic of Ireland, showing multiple different statistics based on the users choices. (<http://airo.maynoothuniversity.ie/external-content/recorded-crime-monitoring-tool>)

## Data conversion process

In this project, 2 different datasets are combined to get the expected outputs, GeoHive dataset and the Irish Crime dataset. The GeoHive dataset will provide the information about a county, such as polygonal coordinates. Through this information, an area can be calculated from this polygonal data. The GeoHive dataset is provided in RDF format while the Crime dataset is provided as a CSV document. This dataset is to be uplifted to RDF or equivalent format for combining and querying the data.

In this project, we selected a single year (2015) crime data to avoid the ambiguity and complexity of having multiple similar values for different years. Another reason for this was that feeding the entire corpus for conversion threw an error due to incorrect values or invalid cells in the CSV File.

For converting the CSV document to TTL/RDF format Apache Jena is being used. The CSV file is read, and each row is parsed removing the extra columns in the process. The empty values are replaced with an empty string and saved to an intermediate CSV file. This Intermediate file is then converted to a turtle file (.ttl). Prefixes such as foaf, owl, RDF etc. are added to the converted model using setNsPrefix() function. The resulting model is written to a TTL file so that it can be used in protégé and Jena for ontology modelling.

Overview of Design

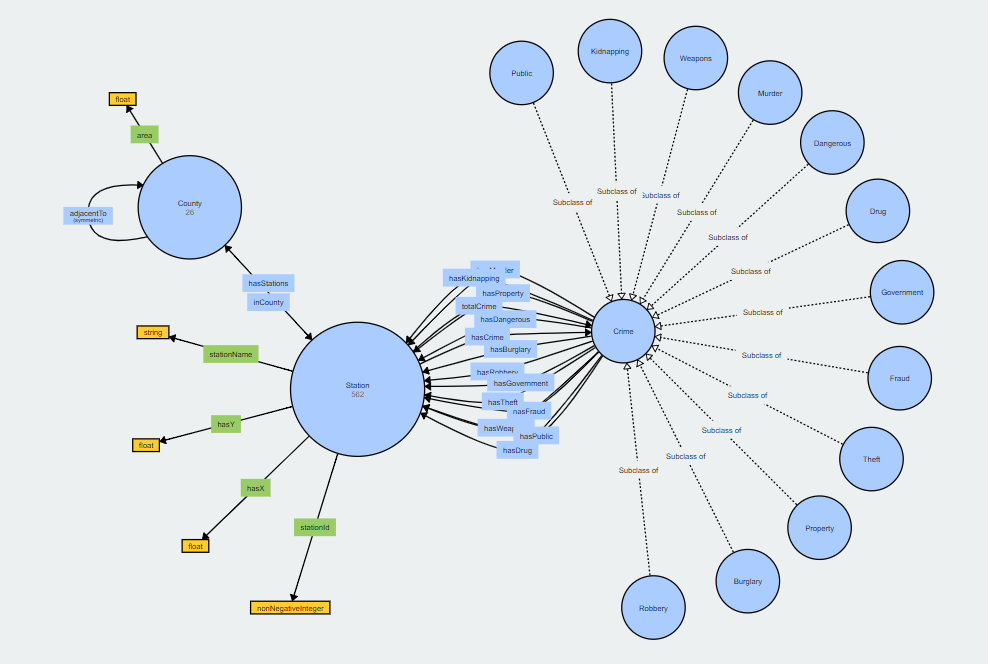
1. Description of query interface

Once the ontology and corresponding owl file was created, the next task was to query for results. The queries are again executed using the Apache Jena library.

The created ontology is loaded using the OntModel class. Once the ontology is loaded the query string is created and the querying is performed using the QueryExecutionFactory class – loading the Result into ResultSet.

A stand-alone Java Swing app was created, displaying the possible queries in the question panel. When the user selects a specific query and clicks on the submit button the query is automatically triggered and the output is displayed on the output pane. This user interface was designed with simplicity in mind, acting as an intermediate between user and ontology; allowing any user to run the queries in a very simple manner.

1. Description of Queries
2. VOWL Representation



Challenges faces while ontology modelling or creating queries

* Deciding whether a class/property was necessary
* Deciding whether the types of crimes should be listed as classes or individuals
* Testing properties like symmetricity and transitivity of properties
* Finalizing which tool to use when developing the ontology (Protégé or Jena)
* Getting the reasoner to work properly and execute in reasonable time
* Deciphering the errors thrown by the reasoner
* Unable to find a relationship between division and county. They were initially assumed to be the same
* Finding the link between the two datasets
* Introducing transitive and symmetric properties

# Conclusion

Our ontology model could successfully link the two different chosen datasets through the use of station coordinates. Using these coordinates, we could then identify in which County each station was and extrapolate that data in our queries to determine which county was the most dangerous in 2015.

While inverse and symmetric properties were defined and used in our ontology, no transitive properties were used. Our initial proposal was to link counties to the divisions they each contain and then link divisions to all the stations within them. Using this approach, the transitive property could have easily been used to infer stations within each county. However, the dataset only provides location information per station, making all the provided Division information essentially useless for us. The dataset explains that Divisions are the region in which different stations are, but since only stations have location information, we had to use stations to link directly to counties. For this reason we also decided to remove Divisions from our ontology since there was no real benefit of having this extra information.

# Self-Reflecting Report

## ANIRBAN BHATTACHARJEE

Implemented several queries in SPARQL and contributed in finalizing the dataset, helped in refining the ontology to some extent. However, there are limitations with this ontology and it could be enhanced by linking it to more than one dataset. Cardinalities could have been well thought of at the beginning of the project.

LOVISH

NEERAJ

## NICHOLAS BONELLO

Most of my focus was on understanding how to create the ontology, modelling the different classes, defining properties per class and evaluating how to interlink the two datasets through different object properties. Once the initial design was modelled and agreed upon with the rest of the group members, I began creating the ontology by making use of the Jena toolkit in Java, updating the ontology throughout the duration of the project with suggestions from the team along the way.

Once the ontology was complete and successfully evaluated by the reasoner, my next task was to use the uplifted Crime and GeoHive datasets to add all the individuals into the newly formed ontology – ensuring that all object properties were correctly added for each individual.

TOMIN BIJAIMON AZHAKATHU

Working with my group in Knowledge and Data Engineering group assignment helps me to learn the importance of group work and documentation. The well-organized team works help us to solve the problem in a quick and efficient manner.

I’m responsible for uplifting or converting the CSV File to RDF File. In this method we planned to eliminate many columns with unwanted data for our ontology and replace all empty cells with the empty string so that there won’t be any parsing error will occur while uplifting which was occurred while trying to uplift the entire dataset as it is. This helps me to learn the core features of Apache Jena Like Model Class, RDF Class etc.

Apart from Uplifting I also contributed to GUI creation for the Query so that anybody can run the defined query without any extra effort. The GUI was created using Java Swing Library. In backend Apache Jena is used for querying from the ontology/ owl file. This helps me study more on Ontology class, Querying, and representation of the result.

Being in the team working with the team members in KDE assignment helps me to learn more deeper in Linked Data, its creation, functionality and representation. It helps me in improving my knowledge in uplifting, creating ontology, creating queries and analyzing results.