Knowledge Representation and Reasoning Project Fall 2025

Dr. Khurram Jadoon

November 24, 2025

Summary

In this course project you will achieve the following:

1. Select a non-RDF structured/unstructured dataset.
2. Design/Re-use an appropriate Ontology.
3. Convert it to RDF/OWL using well defined RDF/OWL conversion guidelines.
4. Publish it as linked data.
5. Link it to other open linked data sets.
6. Show the utility of your dataset by making federated SPARQL Queries.
7. **Bonus**: Demo of an application built using your dataset.

Details

1. Form a Team

- You will be working in teams of 2-3 on this project.
- Submit your project team information here: `https://forms.cloud.microsoft/r/E8sRmLyMUL`
- Last date to submit your team information is given on **MS Teams**.

2. Selection of a domain and dataset

- You are free to select any domain/dataset of your choice. `https://www.kaggle.com/datasets` has a huge list.
- Keep in mind the following when selecting your dataset:
  - You must be able to show evidence that the dataset has NOT already been published or available as linked data.
  - Provide some motivation and context why you have selected this dataset.
  - Examine its potential for interlinking with other data sets.

3. Competency Questions

- For the selected domain & dataset, create a list of **competency questions** that you will be using to design the conceptual model for your knowledge base.
- Later these competency questions will be used to validate your results via SPARQL Queries.

4. Conceptual Modelling for your Knowledge Base

- Create a conceptual model for your knowledge base, which is a diagrammatic representation of your ontology.
- It must contain the T-Box of your KB schema and may contain an example of how the A-Box schema will emerge from the T-Box.
- The classes and relationships must be meaningfully represented.

5. Design an Ontology/Re-use Appropriate Vocabularies

- On your conceptual model you must mention the appropriate class names and predicates reused from other vocabularies.
- Design suitable classes with restrictions/axioms.
- Perform reasoning/classification.

Suggested Guidelines for Ontology Design

Ontology must follow these rules:

- **Classes**
  - At least 20 classes in total.
  - At least one class defined as enumeration of its individuals.
  - At least one class defined using property cardinality restrictions.
  - At least one class defined using property range restrictions.
  - At least one class defined as a union of classes.
  - At least one class defined as an intersection of classes.
  - At least one class defined as a complement of some class.
- **Properties**
  - At least 7 object properties in total.
  - At least one object property should be functional.
  - At least one object property should be inverse functional.
  - At least 3 object properties should have some range restriction.
  - At least 7 datatype properties in total.
- Make sure to check the consistency of the ontology.

Note: If you have tried and some of the rules above don't apply to the context in your domain, then mention clearly and also the justification. The above rules are generic enough to be applied to any domain. Take inspiration from PizzaOntology Tutorial if you are stuck.

Submission:

- Submit two separate ontology files, one without and one with individuals.
- Annotate at least 10 individuals in a separate file. You may hand annotate them in protege to test your defined classes, classifications and any reasoning.

6. Map and Convert your dataset to RDF/OWL using the Ontology

- Convert your dataset to RDF using an appropriate tool e.g. OpenLink Virtuoso or using an RDF API such as Jena/OWLAPI/RDFLib or any other tool/solution available.
- Be careful to devise an appropriate and meaningful URI mechanism for your dataset.
- Tip: Work with a small part of data, some test cases first before mapping to the entire dataset.

7. Publish & Link with other Datasets on the Linked Data cloud

- Publish as Linked Data and expose your RDF data through a SPARQL endpoint.
- Attempt to make your data 5* by linking to other datasets e.g. DBPedia, WikiData or any other.
- You may write your own code to do this or use an existing tool.

8. Validate

- Validate your dataset by writing SPARQL queries against your competency questions.

9. Visualize

- Use an appropriate tool such as VOWL or GraphDB or any other to visualize your RDF Data.
- Create interesting visualization scenarios and include them in your report and demo.

10. Reflect

- Reflect on your learning experience. What added potential were you able to achieve and demonstrate by converting a non-RDF data into linked data?

Bonus

- Create some reasoning scenarios, explore the use of named and defined classes, and perform some reasoning and consistency checking if possible.
- Explore the use of SWRL or other rule based languages for RDF.

Some relevant tools and resources

You may use any relevant tools for the said tasks:

- Jena/ARQ API
- OWLReady/RDFLib
- OpenLink Virtuoso/GraphDB

Note: This is a suggestive project outline. If you have a specific research/application idea in mind using semantic technologies/knowledge graphs you may discuss with the instructor.

Submission

There are three parts to the submission and evaluation process:

1. Project Report
2. Demo/Viva

Project Milestones: Refer to MS Teams for Deadlines

M1: Team Formation, Domain and Dataset Selection

- Submit your project team information (See Section 1).

M2: Competency Questions and Conceptual Model

Submit the following:

- Domain Description
- Competency Questions
- Conceptual Model with identified Vocabularies you will use or design on your own
- Description of the Dataset
- External Linked Data sets identified for Linking, mention particular entities (Also draw them on conceptual model)

Submission Instructions:

- Create a word doc and submit the link to the word doc using this form:
- Name the google doc as: **KRR-Fall2025-Project-YourTeamNo**
- Submit a copy of the document at this tab.
- Incomplete submissions and those that do not follow the instructions will not be graded.

M3: Ontology File, Converted RDF/OWL Data Files

This is intermediary project deliverable to make sure that you are making steady progress towards your project, and not leaving work till last minute. All submissions are graded.
You will submit at-least the following:

- Ontology file, if you have designed your own, and your data converted/mapped to RDF/OWL
- Include your modeling decisions
- Perform some reasoning, and the significance of the reasoning performed

M4: Complete Final Project Submission

- Submit the complete project report and artifacts.

Evaluation Rubric

The final project will be evaluated based on the following criteria:

#	**Criteria**	**Marks**
1	Introduction to the Domain	
2	Motivation and Target Application Use case	
3	Conceptual Model	
4	Ontology Design as per the Conceptual Model	
5	Ontology Visualization in Protege/OWLVis or other tool	
6	Graph Generation and Validation in Protege	
7	Graph Generation using Python	
8	Graph Validation and Visualization using GraphDB/Any other Visualization Tool	
9	Reasoning Scenarios and Validation (Using Protege)	
10	Reasoning in Code via Python	
11	Graph Published via a suitable tool such as GraphDB/Virtuoso with SPARQL Endpoint	
12	SPARQL Queries via Graph DB	
13	SPARQL Queries via Code	
14	**Demo of End User Application with Graph BackEnd [BONUS]**	