Formalities on the Lab

Credits

Most of this material is based on the coursework of the City University, London, INM713 course of the 2020/21 academic year and reused here with kind permission by Ernesto Jiménez-Ruiz.

Deadline

You are not required to submit lab results. The lab material serves to help you getting familiar with the hands-on tools and questions. It makes sense to work on them in the week the respective topic is handled in class. Lab results will be discussed in the following week.

Working together

If you intend to work together as a team, that is fine. This can be the same team you use for the project, but that is not mandatory.

Links for the Lab

Main Page

https://github.com/turing-knowledge-graphs/teaching/tree/main/city

Lab 1

https://github.com/turing-knowledge-graphs/teaching/blob/main/city/2020-2021/INM713_Lab_Session1.pdf

Lab 2

https://github.com/turing-knowledge-graphs/teaching/blob/main/city/2020-2021/INM713_Lab_Session2.pdf

Lab 3: SparQL

https://github.com/turing-knowledge-graphs/teaching/blob/main/city/2020-2021/INM713_Lab_Session3.pdf

Lab 4: RDFS und Reasoning

Bei Interesse angucken, Reasoning behandeln wir aber erst später

https://github.com/turing-knowledge-graphs/teaching/blob/main/city/2020-2021/INM713_Lab_Session4.pdf

Lab 5: OWL
https://github.com/turing-knowledge-graphs/teaching/blob/main/city/2020-2021/INM713_Lab_Session5.pdf

Lab 6: KG Creation from Tables
https://github.com/turing-knowledge-graphs/teaching/blob/main/city/2020-2021/INM713_Lab_Session6.pdf

Formalities and Submission

Credits
Most of this material is based on the coursework of the City University, London, INM713 course of the 2020/21 academic year and reused here with kind permission by Ernesto Jiménez-Ruiz.

Deadline
The most important thing first: Deadline for submission of the coursework is **Sunday, July 16 or a week prior to your exam, whatever comes first**.

Working together
If you intend to work together as a team of 2 to 3 people, please indicate that at the beginning of the semester. If you have not named a team, we expect you to work on your own.

Submission
All coursework shall be submitted in your gitlab repository. We will grade based on the version in there at the time of the deadline. Coursework includes the software you have developed and a short report.

More precisely: *Caution: This may be extended once all tasks have been defined.*

- A README file detailing where the components below can be found.
- A report where you discuss the choices and results of the different tasks. In PDF format at most 10 pages, appendix if needed.
- All your code and a readme file on how to execute the scripts.
- Ontology in turtle format
- RDF data in turtle format
- Extended RDF data after reasoning in turtle format
- CSV file created in the SPARQL subtask

- Ontology alignment in turtle format
- Ontology embeddings in both binary and textual format.

Grading
The individual parts of the coursework will be weighted as follows for the grade:

- Report: 5%
- Basics: Creating an Ontology: 10%
- KG Creation: Modelling: 10%
- KG Creation: Tabular Data: 25%
- KG Usage: Querying and Reasoning: 15%
- KG Improvement: Alignment: 15%
- KG Improvement: Evolution: 10%
- KG Improvement: Embeddings: 10%

Dataset
We mostly rely on the dataset provided by City Uni. This is based on the kaggle dataset about Pizza Restaurants which in turn is based on a dataset provided by Datafiniti's Business Database. (https://www.kaggle.com/datafiniti/pizza-restaurants-and-the-pizza-they-sell). The dataset is linked below.

Coding
You can use Java or Python as programming language. You can reuse material from the lab sessions and any external library/resource as long as you cite them properly in your report.

Basics: Creating an Ontology

Credit
This assignment is identical to the one in the City uni course.

Ontology Modelling (Task OWL)
Create a small ontology that models the domain of Pizza_data. Use Protégé to develop the ontology. Save the ontology into turtle format (e.g., .ttl) and rdf/xml format (e.g., .owl).

Guidelines:
Subtask OWL.1: Discussion and motivation of the different modelling choices in the report (30%).

Subtask OWL.2: Change namespace for the ontology
e.g., http://www.city.ac.uk/ds/inm713/your_name/ (5%).

Subtask OWL.3: Create a prefix (e.g., your initials ejr:) for the defined namespace (5%).

Subtask OWL.4: The ontology should abstract knowledge about the domain, valid for the data at hand but potentially valid for other datasets (e.g., tables). The ontology should, in principle, only contain concepts, object and data properties. Tip: you should difference between the general concept Margherita_Pizza and the margherita pizza served at "Pizzeria da Mario" (10%).

Subtask OWL.5: Organize classes into a hierarchy (15%).

Subtask OWL.6: Create local scope property restrictions or global scope ones (i.e., domain and range) as necessary (15%).

Subtask OWL.7: Create appropriate property characteristics (10%).

Subtask OWL.8: Create a label (e.g., name or synonym) and a comment (e.g., meaning of the concept) for each created entity (5%).

Subtask OWL.9: Create or reuse an annotation property creator and annotate/indicate the ontology has been created by you (5%).

KG Creation: Modelling

Conceptual Modeling (Task Model)
Revise the ontology you previously created and apply the basic steps of LOT and evaluate your results using the OntoClean.

Guidelines:
Subtask Model.1: Motivate your ontology with a set of competency questions or statements and document those in the report. (30%)

Subtask Model.2: From the competency questions and statements, derive terms and properties that your ontology needs to cover. Add any missing classes or properties. (30%)

Subtask Model.3: Verify the rigidity, identity, and unity of your classes. If changes to your class-hierarchy and/or new properties become necessary, add them to the ontology. (20%)

Subtask Model.4: Document all steps and resulting changes to your ontology in the report. (20%)

KG Creation: Tabular Data

Credit
This assignment is identical to the one in the City uni course.

Tabular Data to Knowledge Graph (Task RDF)
Transform Pizza_data into RDF triples using your favorite programming language. Please document your code. Save the RDF data into turtle format (.ttl).

Subtask RDF.1 Discuss in the report the different transformation choices and how entity resolution was treated (30%). As we saw in the module, there is not a unique way to transform the elements in a table into elements of a knowledge graph (i.e., classes, object properties, data properties or instances). For example the column "menu item" contains the name of the pizza but it may also include additional information about toppings and type of pizza. Tip1: You will need to apply some basic text processing and entity recognition based on the ontology vocabulary. Tip2: After processing the data, one may also need to extend the ontology with new elements.

Subtask RDF.2 Create RDF triples (30%). Use the created ontology to guide the transformation e.g., to link the data to the ontology (e.g., via rdf:type) and to use the defined ontology properties (e.g., has_topping) and concepts (Margherita_Pizza).

Subtask RDF.3 For the cells in the columns city, country and state; instead of creating new URIs (e.g., new individuals) for the information in the table cells, reuse an entity URI from DBPedia, Wikidata or Google's Knowledge Graph (http://dbpedia.org/resource/Los_Angeles). Tip: communicate with their respective look-up services as we saw in the lab session (25%).

Subtask RDF.4 Correctness of the code and code documentation (15%).

Various data sets are used in the labs. Here are the links:

- Pizza Ontology
- SPARQL Playground
- Nobel Prize Data
- World cities dataset
- DBPedia
- OAEI datasets

KG Usage: Querying and Reasoning

Credit

This assignment is identical to the one in the City uni course.

SPARQL and Reasoning (Task SPARQL)

Write SPARQL queries, according to the requirements in the following subtasks, and execute them over the created ontology and the generated data.

Subtask SPARQL.1 ~~Perform reasoning with the created ontology and the generated data. Save the extended graph in turtle format (.ttl)~~ (10%).

Subtask SPARQL.2 ~~Return all the details of the restaurants that sell pizzas without tomate (i.e., pizza bianca). Return the results as a CSV file~~ (20%).

Subtask SPARQL.3 ~~Return the average prize of a Margherita pizza (20%).~~

Subtask SPARQL.4 ~~Return number of restaurants by city, sorted by state and number of restaurants (20%).~~

Subtask SPARQL.5 ~~Return the list of restaurants with missing postcode (20%).~~

Subtask SPARQL.6 Correctness of the queries and code, and documentation of the created SPARQL queries in the report (10%).