COMP 474/6741 Intelligent Systems (Winter 2021)

Worksheet #1: Knowledge Graphs

Task 1. We start by modeling some university-related knowledge in form of a *graph*: (1) Joe studies at Concordia University; (2) Joe knows Jane; (3) Jane studies at McGill University. Draw the graph:

Task 2. Let's add some additional knowledge: (4) Joe's email address is `joe@example.com`; (5) Concordia is located in Montreal; (6) McGill is located in Montreal. Add these to the graph above.

Task 3. Graphs can be represented as *triples* (and vice versa), consisting of

`<subject> <predicate> <object>`

Write the triples corresponding to the first three statements above:

1. .....
2. .....
3. .....

Task 4. Here's another triple: (*Joe, is a, Person*). Add it to the graph you drew in Task 1.

Task 5. So far, we defined everything in terms of natural language. That's not very useful for a knowledge base to be used in an intelligent system. Rather than writing *"Concordia"*, we will use a URL that points to a machine-readable description in the RDF *(Resource Description Framework)* format. Using your phone or laptop, look up (our) Concordia University in the open knowledge base *DBpedia* (http://dbpedia.org):[1]

- URL: .....

[1] What you see in your browser is actually a human-readable web page obtained through a HTTP 303 redirect, not the raw RDF data that would be consumed by a program.

Task 6. Ok, now let's go back to the graph from Task 1. Replace the string "Concordia" with the URL you obtained in the previous step in your graph above. *Note:* To obtain a complete RDF graph, you'd have to continue replacing all subjects, predicates and objects (except literals) with URLs.

Task 7. DBpedia contains (among other information) the link to Concordia's *homepage*. Find the information and write it in form of a triple:

- .....

Now add this triple to the graph from Task 1.

Task 8. What about our Joe and Jane? Unlike some famous persons, we will most likely not find them in DBpedia (or any other public knowledge graph). However, an organization might have them in their own graph (for example, in Facebook, LinkedIn, or Concordia's student database). For this example, we simply name them as `http://example.org/joe#me` (likewise for Jane). How do we model the predicate that Joe *knows* Jane in a machine-readable way? The details will be the topic of the next lecture; for now simply use the URI `http://xmlns.com/foaf/0.1/knows`. Re-write the triple (2) from Task 1 using three URIs:

- .....

Observe how an intelligent agent can now learn more about each part of this triple simply by resolving the URI and reading the RDF data retrieved through it.

Task 9. Writing these full URIs (technically IRIs) is tiring (and uses up storage space). Using the following prefixes:

```
PREFIX foaf:  <http://xmlns.com/foaf/0.1/>
PREFIX user:  <http://example.org/>
```

re-write the triple from Task 8:

- .....

Task 10. Given the knowledge graph you constructed, can an intelligent agent answer the question "*Which city is Joe studying in?*"? If yes, how? If no, why not?

To learn more about the technical details, see the references provided in the lecture, in particular https://www.w3.org/TR/rdf11-primer/