Concordia University

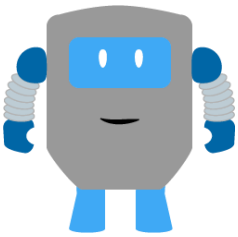
Engineering and Computer Science

COMP 6741

Intelligent Systems

Project-1 Report

Unibot



Submitted To: Prof. Dr. René Witte

Date : 21 March, 2021

Team **FL_G_03**

First Name	Last Name	Student ID	Email ID
Chirag	Patel	40160656	chiragpatel2965@gmail.com
Harshil	Patel	40163431	harshilpatel1903@gmail.com

Table of Contents

ABSTRACT:

The overall goal of this project is to build Unibot, an intelligent agent that can answer university course- and student-related questions, using a knowledge graph and natural language processing.
The first phase of our project was to gather information on all Concordia University courses. This information was used to generate a Knowledge Graph using existed and newly created vocabularies. A series of competency questions are written and transformed into queries to test the functionalities of generated Knowledge Base.

Competency Questions

1. How many courses are offered at Concordia?
2. Which topics are covered in COMP 6741 lectures?
3. Which topics is Arihant is competent in?
4. Which courses at Concordia teaches deep learning?
5. Where can I get more information about Comp6721?
6. What are grades of Brendon Rihards?
7. Which COMP courses are taught at Concordia?
8. Which lecture of Comp6741 courses knowledge graph?
9. Does ACCO230 and ACCO350 cover similar topic?
10. How many students have taken COMP6721 courses?

Vocabulary:

Description how you modeled the schema for your knowledge base, including the vocabularies you reused, any vocabulary
extensions you developed, etc. Give brief justifications where appropriate (e.g., choice of existing vocabulary).

Table 1 : Vocabulary

RDF	Type	For defining classes and properties
	Property	
RDFS	Class	For defining classes
	subClassOf	For extending class definitions
	domain	For defining properties
	range	
FOAF	Person	A student is a person

	topic	Property used to relate a topic to a university course
	mbox	Mail info of the student
DBO	document	For defining our transcripts document seemed a fitting concept.
	type	For use of the Public_university type
DBR	Course_(education)	Suitable for broadly defining the courses offered at universities
	Public_university	Suitable definition for Universities
XSD	string	Literal strings serve as the object of several properties we have generated.
DCTERMS	relation	A related resource.
DCMITYPE	event	A non-persistent, time-based occurrence.

We used the focu schema <http://focu.io/schema#> and the ex schema <http://example.org/> to contain instances of the remaining classes and properties. Their definitions and details concerning their usage are given below. Each class and property created for our vocabulary contains an rdfs:label and rdfs:comment property as per project requirement.

1) **focu:Student**

```
a rdfs:Class;
rdfs:subClassOf foaf:Person;
```

Class to describe a student at the university

2) **focu:Lecture**

```
a rdfs:Class;
rdfs:subClassOf dcmitype:Event;
```

Class to describe lecture as a Dublin Core Event

3) **focu:Course**

```
a rdfs:Class;
rdfs:subClassOf  dbr:Course_(education);
```

Class to describe Course at the university in the combined form of Subject + Catalog

4) **focu:courseTaken**

```
a rdf:property ;
rdfs:domain <Student> ;
rdfs:range <Course>;
```

Property to describe what course has been taken by a Student

5) **focu:hasContent**

```
a rdf:property ;
rdfs:subClassOf dcterms:relation ;
```

```
rdfs:domain <Lecture> ;
rdfs:range xsd:string;
```

Property to describe content for a specific lecture

6) **focu:competencies**

```
a rdf:Property;
rdfs:domain <Student>;
rdfs:range xsd:string;
```

Property to describe topics in which a Student is competent in

7) **focu:Record**

```
a dbo:document;
```

Property to describe information regarding course and grade achieved by a Student

8) **focu:hasRecord**

```
a rdf:Property;
rdfs:domain <Student>;
rdfs:range <Record>;
```

Property to describe what records a Student has

9) **focu:subject**

```
a rdf:Property;
rdfs:domain <course>;
rdfs:range xsd:string;
```

Property to describe course subject (“COMP”, “SOEN”)

10) **focu:catalog**

```
a rdf:Property;
rdfs:domain <course>;
rdfs:range xsd:string;
```

Property to describe course catalog number (“474”, “6741”)

11) **focu:credits**

```
a rdf:Property;
rdfs:domain <course>;
rdfs:range xsd:int;
```

Property to describe number of credits a course contains

12) **focu:grade**

```
a rdf:Property;
rdfs:domain <Record>;
rdfs:range xsd:string;
```

Property to describe what grade is scored by a student in particular course

13) **focu:provenance**

```
a rdf:Property;
rdfs:domain foaf:topic;
rdfs:range xsd:string;
```

Property to describe the source of topic identified for the course

14) **focu:firstName**

```
a rdf:Property;
rdfs:domain <Student>;
rdfs:range xsd:string;
```

Property to describe first name of the student

15) **focu:lastName**

```
a rdf:Property;
rdfs:domain <Student>;
rdfs:range xsd:string;
```

Property to describe last name of the student

16) **focu:offeredAt**

```
a rdf:Property;
rdfs:domain focu:Course;
rdfs:range dbr:Public_university;
```

Property to describe what course is offered at what university

Schema Design :


http://www.w3.org/1999/02/22-rdf-syntax-ns#type
http://www.w3.org/1999/02/22-rdf-syntax-ns#type
http://www.w3.org/1999/02/22-rdf-syntax-ns#type
http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
http://focu.io/schema#competencies
http://www.w3.org/1999/02/22-rdf-syntax-ns#type
http://www.w3.org/2000/01/rdf-schema#label
Competencies
http://www.w3.org/1999/02/22-rdf-syntax-ns#type
http://www.w3.org/1999/02/22-rdf-syntax-ns#type
http://www.w3.org/1999/02/22-rdf-syntax-ns#type
Competencies, defined as a set of topics, based on the courses a student success ...
http://www.w3.org/2000/01/rdf-schema#comment
An integer number which uniquely identifies a course within a particular discipl ...
http://focu.io/schema#catalog
http://www.w3.org/2000/01/rdf-schema#comment
http://www.w3.org/2000/01/rdf-schema#label
Course number
http://www.w3.org/2000/01/rdf-schema#domain
http://example.org/course
http://focu.io/schema#subject
http://www.w3.org/2000/01/rdf-schema#domain
http://www.w3.org/2000/01/rdf-schema#comment
An abbreviation for the discipline or department under which an academic course ...
http://www.w3.org/2000/01/rdf-schema#domain
Discipline of course
http://www.w3.org/2000/01/rdf-schema#label
http://www.w3.org/2000/01/rdf-schema#label
Letter grade
http://focu.io/schema#credits
http://www.w3.org/2000/01/rdf-schema#range
http://www.w3.org/2001/XMLSchema#int
http://www.w3.org/2000/01/rdf-schema#comment
A single letter measuring the performance in the course of the associated Record ...
http://www.w3.org/2000/01/rdf-schema#comment
A university course is offered at the associated university
http://www.w3.org/2000/01/rdf-schema#comment
An abbreviated course displicine and course number which combined uniquely ident ...
http://www.w3.org/2000/01/rdf-schema#label
Offered At
http://www.w3.org/2000/01/rdf-schema#domain
http://focu.io/schema#offeredAt
http://www.w3.org/2000/01/rdf-schema#label
Unique identifier for a course
http://www.w3.org/2000/01/rdf-schema#range
http://focu.io/schema#Course
http://www.w3.org/2000/01/rdf-schema#subClassOf
http://www.dbpedia.org/resource/Course_(education)
http://www.w3.org/1999/02/22-rdf-syntax-ns#type
http://www.w3.org/2000/01/rdf-schema#Class
http://www.dbpedia.org/resource/Public_university
http://www.w3.org/1999/02/22-rdf-syntax-ns#type
http://www.w3.org/2000/01/rdf-schema#comment
Information about lectures in a course
http://www.dbpedia.org/ontology/type
http://www.w3.org/1999/02/22-rdf-syntax-ns#type
http://focu.io/schema#Lecture
http://www.w3.org/2000/01/rdf-schema#label
Lecture
http://www.w3.org/2000/01/rdf-schema#subClassOf
http://purl.org/dc/dcmitype/Event
http://www.w3.org/2000/01/rdf-schema#label
Student
http://focu.io/schema#Student
http://www.w3.org/2000/01/rdf-schema#subClassOf
http://www.w3.org/2000/01/rdf-schema#comment
http://xmlns.com/foaf/0.1/Person
http://xmlns.com/foaf/0.1/name
Concordia University
http://www.w3.org/2000/01/rdf-schema#seeAlso
http://example.org/Concordia
http://www.dbpedia.org/resource/Concordia_University
A person enrolled in one or more courses at a academic institution or other plac ...

Knowledge Base Construction:

Describe (a) your dataset and (b) your process and developed tools for populating the knowledge base from the dataset.
Describe how to run the tools to create the knowledge base. Explain your process for linking entities to DBpedia

Knowledge Base was build using Python and following libraries:

Pandas - for reading and pre-processing csv files
Spacy - to perform DBPedia Spotlight annotation on course descriptions
Rdflib – To create knowledge graph using **Graph** and use existing namespaces provided by rdflib

Database description :

Database was generated using files obtains from **Concordia opendata** datasets, namely course catalog and course description files were used. Using pandas dataframe, both the files were merged into one :

```
df = pd.merge(a, b, on = "Course ID", how="inner")
```

Unwanted columns were then dropped. The final output was saved to "**dataset.csv**".

The file contains 7050 datapoints each of which contains the following values :

```
Course ID, Subject, Number, Name, Credits, Description
```

There are 2 additional files "**lecture_data.csv**" and "**content_data.csv**" which contains following values respectively:

```
CourseId, Identifier, Title, seeAlso, Topic
CourseId, Identifier, ContentType, Content
```

Additionally, a "**students.csv**" file contains information about randomly generated students using **Students.py** :

```
studentID, firstName, lastName, Email, Subjects, SubjectsGrades, Competencies
```

Generating Knowledge Graph :

Knowledge Graph was generated using **main.py**.

All the csv files from dataset folder were loaded using pandas. For each data point in the dataset.csv, following values were added to the graph :

```
course_id, course_subject, course_number, course_title, course_credit,
course_description, course_topics, course_link
```

Using Spacy and it's connection to DBPedia Spotlight, the course title and course descriptions were annotated with a confidence score or 0.4 and were stored in course topics list.

```
course_topics = list(set([(ent.text, ent.kb_id_) for ent in topic_ents]))
```

Each topic was then added to the graph along with it's course.

Similarly, lecture data and content of lecture matching to the course were split into list and added as triples to the graph.

After this, randomly generated information of 100 students were created and appropriate triples for them were added to the graph.

Finally the graph was stored in a serialize format of both N-triple and Turtle namely **knowledge_base_n3.nt** and **knowledge_base_ttl.ttl**

Generated Knowledge Base:

Triples – 111284
Course – 7025
Lectures – 26
Contents – 118
Students – 100

Queries :

⇨ The following queries have been used to test the graph. The queries can be found in the accompanying queries.sparql file. In the report, we will limit to 10 queries, but more are defined in the queries.sparql file

- Question 1
  How many courses are offered at Concordia?

  Query

```
SELECT (count(?courseId) as ?CourseCount)
WHERE {
       ?courseId a focu:Course.}
```

  Result

```
"7009"^^<http://www.w3.org/2001/XMLSchema#integer>
```

- Question 2
  Which topics are covered in COMP 6741 lectures?

  Query

```
SELECT ?topics  WHERE {
?sub focu:subject "COMP" .
?sub focu:catalog "6741".
?sub rdfs:comment ?topics .
}
```

  Result

```
Knowledge representation and reasoning. Uncertainty and conflict
resolution. Design of intelligent systems. Grammar-based,
rule-based, and blackboard architectures.
```

- Question 3
  Which topics is Arihant competent in?

  Query

```
SELECT ?Competencies  WHERE {
       ?sudent a focu:Student.
       ?sudent focu:firstName "Arihant".
       ?sudent focu:competencies ?Competencies.
}
```

  Result

  1 automated reasoning
  2 audit
  3 A.I.
  4 CGA
  5 heuristic
  6 recurrent neural networks
  7 knowledge representation
  8 Artificial Intelligence
  9 ACCO
  10 assurance services

- Question 4
  Which courses at Concordia teaches deep learning?

  Query

```
SELECT ?subjects  WHERE {
  ?subjects foaf:topic dbr:deep_learning.
}
```

  Result

```
1 <http://example.org/CEPS1114EO>
2 <http://example.org/COEN432>
3 <http://example.org/COMP432>
```

- Question 5
  Where can I get more information about Comp6721?

  Query

```
SELECT ?information
WHERE {
       ?subject a focu:Course.
       ?subject focu:subject "COMP".
       ?subject focu:catalog "6721".
       ?subject rdfs:seeAlso ?information.
 }
```

  Result

```
1 ('A.I.', 'http://dbpedia.org/resource/Artificial_intelligence')
```

- Question 6
  What are grades of Brendon Rihards?

Query

```
SELECT ?course ?grade  WHERE {
  ?student a focu:Student.
  ?student focu:firstName "Brendon".
  ?student focu:lastName "Rihards".
  ?student focu:hasRecord ?record.
  ?record focu:courseTaken ?course.
  ?record focu:grade ?grade
}
```

Result

1 http://example.org/SOEN6441 A-
2 http://example.org/ACCO465 B+
3 http://example.org/COMP6741 B

- Question 7
  Which COMP courses are taught at Concordia?

Query

```
SELECT ?subjects
WHERE {
   ?subjects a focu:Course.
   ?subjects focu:subject "COMP".
     } LIMIT 5
```

Result

```
1 <http://example.org/COMP6651>
2 <http://example.org/COMP371>
3 <http://example.org/COMP6661>
4 <http://example.org/COMP6521>
5 <http://example.org/COMP492>
```

- Question 8
  Which lecture of Comp6741 courses knowledge graph?

Query

```
SELECT ?lecture
WHERE {
       ?subject a focu:Course.
       ?subject focu:subject "COMP".
       ?subject focu:catalog "6741".
       ?lecture dcterms:isPartOf ?subject.
       ?lecture foaf:topic "Knowledge_Graph".

 }
```

Result

1 <http://example.org/COMP6741_Lec2>

- Question 9
  Does ACCO230 and ACCO350 cover similar topic?

Query

```
SELECT ?topics
WHERE {
       ?subject1 a focu:Course.
```

```
        ?subject1 focu:subject "ACCO".
        ?subject1 focu:catalog "230".
        ?subject1 foaf:topic ?topics.
        ?subject2 a focu:Course.
        ?subject2 focu:subject "ACCO".
        ?subject2 focu:catalog "350".
        ?subject2 foaf:topic ?topics.

 }
```

Result
1 <http://www.dbpedia.org/resource/Accounting>

- Question 10
  How many students have taken COMP6721 courses?

  Query
```
SELECT ?firstName ?lastName  WHERE {
  ?student a focu:Student.
  ?student focu:firstName ?firstName  .
  ?student focu:lastName ?lastName.
  ?student focu:courseTaken ?course.
  ?course a focu:Course.
  ?course focu:subject "COMP".
  ?course focu:catalog "6721".
} LIMIT 5
```

  Result
```
1 Dante Walter
2 Bartlomiej Zubair
3 Bruin Thomas-Jay
4 Aaryn Struan
5 Ardeshir Madison
```

References

[1] RDF Schema 1.1
https://www.w3.org/TR/rdf-schema/

[2] FOAF Vocabulary
http://xmlns.com/foaf/spec/

[3] Dublin Core Metadata Initiative
https://www.dublincore.org/specifications/dublin-core/dcmi-terms/#http://purl.org/dc/dcmitype/Event

[4] Vivo Core Ontology
https://lov.linkeddata.es/dataset/lov/vocabs/vivo

[5] Vivo Tutorial by Shanshan Chen, Yuyin Sun, Ying Ding
https://info.sice.indiana.edu/~dingying/Teaching/S604/VIVO-tutorial-v1.pdf

[6] Merge 2 CSV files.
https://www.geeksforgeeks.org/how-to-merge-two-csv-files-by-specific-column-using-pandas-in-python/

[7] Write SPARQL query in Fuseki-Server
https://www.youtube.com/watch?v=5-UfFV5XmTI

[8] Remove blanks from text file
https://www.kite.com/python/answers/how-to-remove-empty-lines-from-a-string-in-python

[9] Spacy to annotate using DBPedia Spotlight
https://spacy.io/api