CSE 5320/7320

Final Exam

Due: Wed May 8

You are working for a national security agency and a message has been intercepted. It reads:

`Jimmy works at Google in California. He was seen whispering to Marla.`

There are several questions for which you are asked to provide answers:

- What are the names of all the Persons mentioned?
- What are the names and locations of all Organizations mentioned?
- What organization is Jimmy affiliated with?
- Does Jimmy know Marla?

As a human you can deduce the following facts:

- `Jimmy and Marla are Persons`
- `Jimmy has a givenName of "Jimmy"`
- `Marla has a givenName of "Marla"`
- `Google is an Organization`
- `Google has a name of "Google"`
- `California is a Place`
- `Google is located in California`
- `California is a Place`
- `Jimmy is affiliated with Google`
- `Jimmy knows Marla`

Your first task is to represent these facts as N3/Turtle triples using only schema.org terms and your namespace. In this exercise we will not use anonymous nodes.

Use the following `Schema.org items:`

- `Person`
- `Organization`
- `Place`
- `knows`
- `memberOf`
- `affiliation`
- `givenName`
- `name`
- `location`

`For your namespace use: http://s2.smu.edu/~yourid/# . Use it only if you cannot find a relevant term in schema.org`

Part A.

- Write N3/Turtle triples that capture the above facts using schema.org and your namespace.
- Write a SPARQL Queries that provides answers to the above questions and capture the results.

Part B.

Now that you know the end goal – a collection of triples and SPARQL queries, write code to automatically generate your N3 using Python and NLTK.

To accomplish this, you should use a data flow pipeline and take advantage of the fact that information extracted in one step can be used in subsequent steps.

Suggested set of steps:

Step 1: Find the Entities – the things talked about and generate N3 from them.

To do Entity Recognition, use the ne-chunker to identify all PERSONs and ORGANIZATIONs and GEs (locations) in the text

For example, for the sentence: **Jimmy works at Google in California.**

the **ne_chunker** will return a syntax tree :

```
(S
  (PERSON Jimmy/NNP)
  works/VBZ
  at/IN
  (ORGANIZATION Google/NNP)
  in/IN
  (GPE California/NNP)
  ./.)
```

We see that the Syntax tree contains:

```
(PERSON Jimmy/NNP)
```

which tells us the class of the entity (PERSON) and a name to use to refer to it. From this you should be able to generate the two triples:

- `:Jimmy rdf:type schema:Person .`
- `:Jimmy schema:givenName "Jimmy" .`

Do this for all the entities you find and generate the following N3

- `:Google rdf:type schema:Organization .`
- `:Google schema:name "Google" .`
- `:California rdf:type schema:Location .`

- :California schema:name "California" .
- :Marla rdf:type schema:Person .

Step 2: Find the Organization Jimmy is affiliated with.

Hint: make a list of verbs that indicate someone is affiliated with some organization. Include 'works' in the list. If you find a verb in the list then you can conclude that there is a affiliation relations

Step 3: Find whether Jimmy knows Marla.

Hint: Replace the 'He' in a sentence with the subject of the previous sentence. This should give you:

Jimmy was seen whispering to Marla

Process this new sentence using Chunking and Chinking so that you end up with something like:

```
(S
  (NP Jimmy/NNP)
  (VP was/VBD seen/VBN whispering/VBG)
  to/TO
  (NP Marla/NNP)
  ./.)
```

Define a set of 5-10 verbs (including whispering) that imply A :knows B.

Examine your VP for different verbs. If there is a match to your list add the triple: A :knows B where A and B are the identities of the NPs

Part C.

Group all you're the triples generated by your code. The triples should match the triples you created in Part A.

Run your SPARQL Query from step 1 against your generated triples.

Submit: a PDF

- a well-organized answer showing your code and output for the parts and steps as a PDF
- all code should be in Courier font of size between 10 -12, single spaced but with spacing and comments for readability.