# Ontology Learning from Linked Data

TP_JAN2021_slide

PhD. Candidate: Ziwei XU
Date: 19/01/2021

# Q1. How to discover knowledge from text?

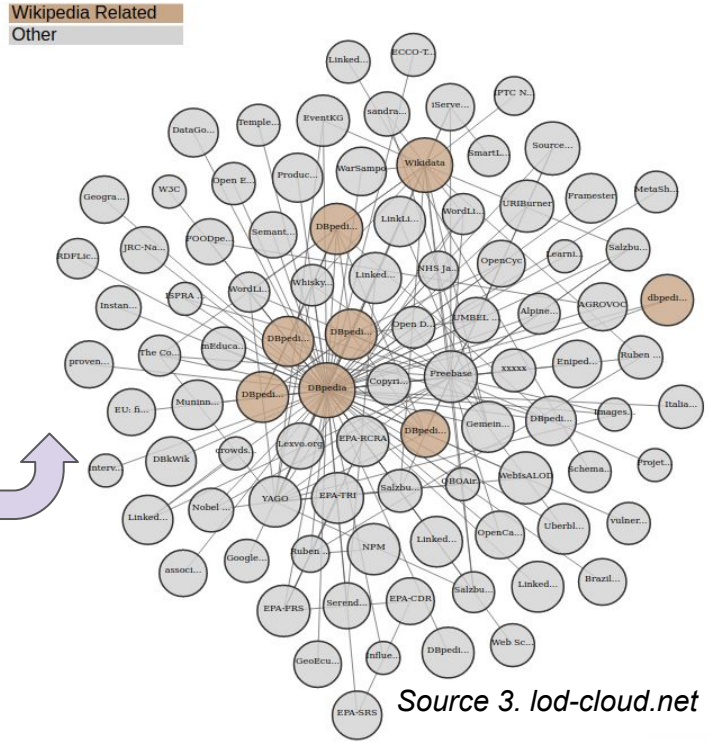
Google was founded in September 1998 by Larry Page and Sergey Brin while they were Ph.D. students at Stanford University in California.

*Source 1. Wikipedia Google*


*Source 2. Entity Visualizer*

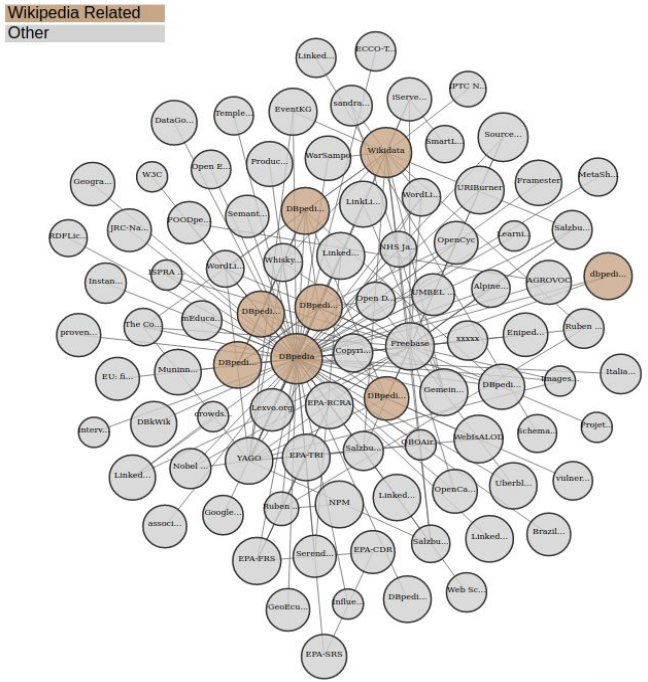
# Q2. How to link the knowledge from text with the knowledge of web?

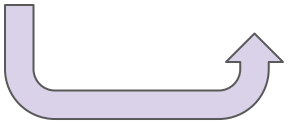
Google ORG was founded in September 1998 DATE by Larry Page PERSON and Sergey Brin PERSON while they were Ph.D. PERSON students at Stanford University ORG in California GPE .

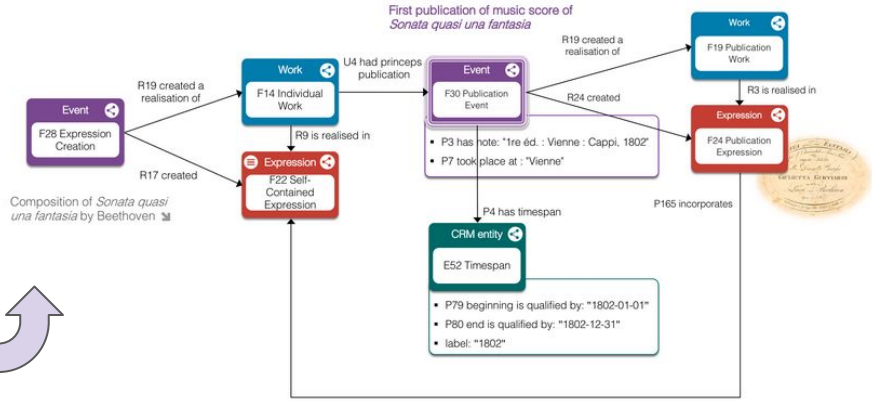



*Source 3. lod-cloud.net*

# Q3. How to build domain ontology from them?






*Source 4. data.doremus.org*

# The Goal of this TP

Q1. How to discover knowledge from text?

Q2. How to link the knowledge from text with the knowledge of web?

Q3. How to build domain ontology from them?

- Learn about ontology building steps
- Practice with SPARQL queries to prepare for ontology building
- Experiment with machine learning techniques for knowledge discovery

# The Introduction of Web Ontologies

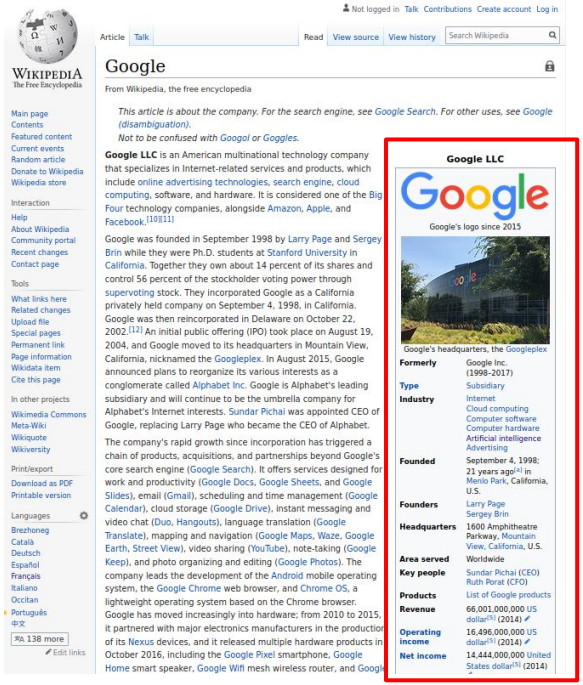
**DBpedia**

Background: DBpedia is the most popular and prominent web ontology. Since the first public release in 2007, DBpedia is up-dated roughly each year.

Knowledge Acquisition Method: DBpedia is created from automatically-extracted structured information contained in Wikipedia, such as from infobox tables, categorization information, geo-coordinates, and external links.

Utilities: DBpedia is used extensively for research, but is also relevant in commercial settings: companies use it to organize their content, such as the BBC and the New York Times.

Hint: A classes visualizer of DBpedia:
<http://mappings.dbpedia.org/server/ontology/classes >




*Source 1. Wikipedia Google*

# The Introduction of Web Ontologies

**Freebase**

Background: Freebase is announced in 2007 and was acquired by Google in 2010 and shut down its services in 2015.

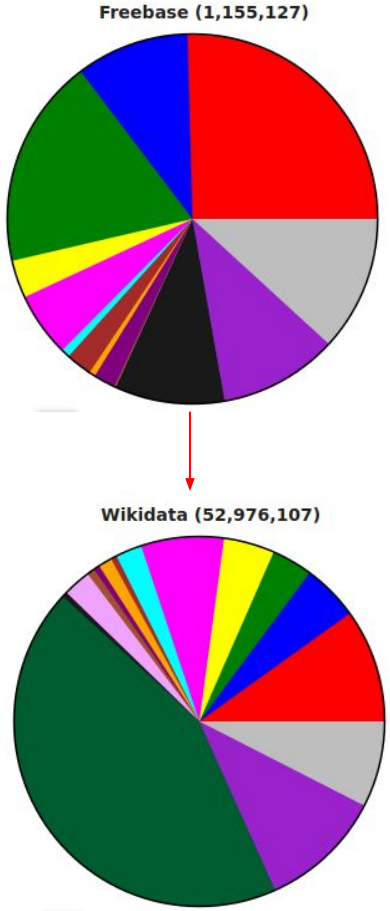
Knowledge Acquisition Method: In contrast to DBpedia, Freebase had provided an interface that allowed end-users to contribute by editing structured data.

Utilities: “Freebase is closed source.” Their data was integrated into Wikidata sep by step.

**WikiData**

Background: Wikidata is a free, collaborative, multilingual, secondary database, collecting structured data.

Utilities: It can be the connection of many web ontologies.




*Source 5. WikiProject Freebase*

# The Introduction of Web Ontologies

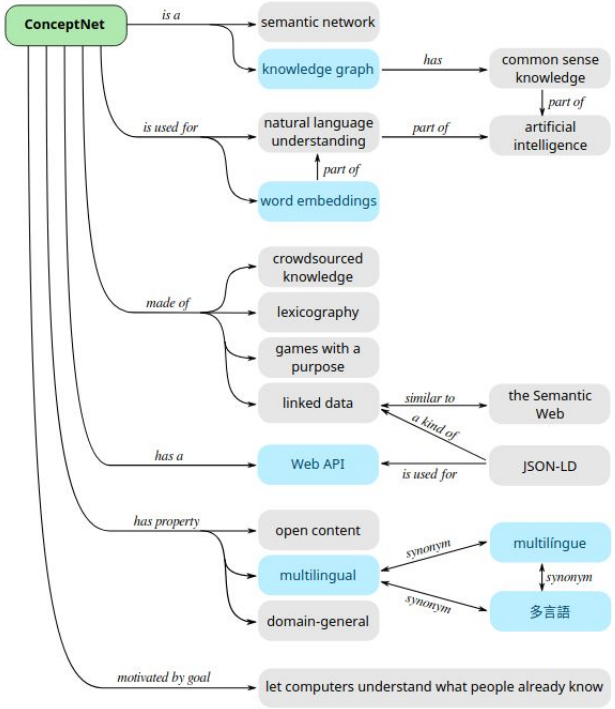
**ConceptNet**

Background: user-friendly knowledge base

Knowledge Acquisition Method: Initially built on top of commonsense sentences, but not from the wikipedia sentences

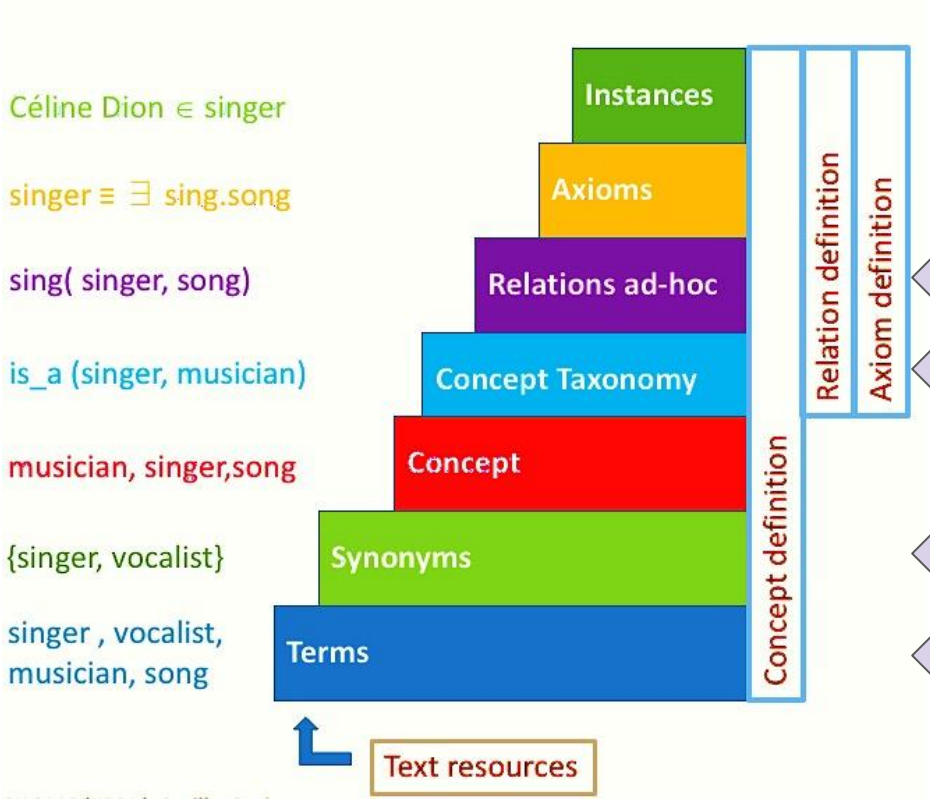
Utilities: use by API or their search engine interface

Hints: <http://www.conceptnet.io/>




*Source 6. Conceptnet.io*

# I. Learning the Ontology Building Steps




Ad-hoc relations discovery between terms

Hypernym relations discovery between terms

Synonym relations discovery between terms

Domain terms extraction

Ontology learning layer cake (Buitelar et al. 2005)

# II. Practice with SPARQL queries

## Practice 1

#1.1 (review) Please find all the resources of synonyms of ‘violin’ in DBpedia.

**Hints**

- DBpedia SPARQL servicer: http://dbpedia.org/sparql
- use property <owl:sameAs>

# II. Practice with SPARQL queries

**The SPARQL of Practice #1.1**

```
PREFIX dbr: <http://dbpedia.org/resource/>
PREFIX owl: <http://www.w3.org/2002/07/owl#>

SELECT ?y where {
  dbr:Violin owl:sameAs ?y .}
```

OR

```
PREFIX dbr: <http://dbpedia.org/resource/>
PREFIX dbp: <http://dbpedia.org/property/>
SELECT ?y where {
  dbr:Violin dbp:names ?y . }
```

| y |
|---|
| http://cs.dbpedia.org/resource/Housle |
| http://de.dbpedia.org/resource/Violine |
| http://es.dbpedia.org/resource/Violín |
| http://eu.dbpedia.org/resource/Biolin |
| http://fr.dbpedia.org/resource/Violon |
| http://id.dbpedia.org/resource/Biola |
| http://it.dbpedia.org/resource/Violino |
| http://ja.dbpedia.org/resource/ヴァイオリン |
| http://ko.dbpedia.org/resource/바이올린 |
| http://nl.dbpedia.org/resource/Viool |
| http://pl.dbpedia.org/resource/Skrzypce |
| http://pt.dbpedia.org/resource/Violino |
| http://wikidata.dbpedia.org/resource/Q8355 |
| http://el.dbpedia.org/resource/Βιολί |
| http://www.wikidata.org/entity/Q8355 |
| http://rdf.freebase.com/ns/m.07y_7 |
| http://d-nb.info/gnd/4019791-8 |

| y |
|---|
| http://dbpedia.org/resource/Fiddle |

# II. Practice with SPARQL queries

## Practice 1

#1.2 Please find the super class(hypernym) of ‘violin’ in WikiData.
How can we improve the results ?

**Hints**:

- WikiData SPARQL servicer: https://query.wikidata.org/
- WikiData Id Search Engine: https://www.wikidata.org/wiki/Special:Search
- The subject ‘violin’ and the property ‘subClassOf’
- For more info: https://www.wikidata.org/wiki/Wikidata:SPARQL_tutorial

# II. Practice with SPARQL queries

**The SPARQL of Practice #1.2**

```
#subclass of (P279)
#violin (Q8355)
SELECT ?hyper
WHERE
{ wd:Q8355 wdt:P279 ?hyper. }
```

OR

```
SELECT ?hyper ?hyperLabel
WHERE
{ wd:Q8355 wdt:P279 ?hyper.
  SERVICE wikibase:label { bd:serviceParam
wikibase:language "[AUTO_LANGUAGE]". } }
```

| hyper |
| --- |
| wd:Q57050725 |

| hyper | hyperLabel |
| --- | --- |
| wd:Q57050725 | necked box lutes played with a bow |

# II. Practice with SPARQL queries

**The SPARQL of Practice #1.2**

```
SELECT ?hyper ?hyperLabel
WHERE
{ wd:Q8355 wdt:P279+ ?hyper.
  SERVICE wikibase:label { bd:serviceParam
wikibase:language "[AUTO_LANGUAGE]". } }
```

| hyper | hyperLabel |
|---|---|
| wd:Q57050725 | necked box lutes played with a bow |
| wd:Q192096 | bowed string instrument |
| wd:Q55724840 | necked box lutes |
| wd:Q1051772 | chordophone |
| wd:Q1798603 | string instrument |
| wd:Q55724833 | necked lutes |
| wd:Q34379 | musical instrument |
| wd:Q30038759 | handle lutes |
| wd:Q39546 | tool |
| wd:Q267228 | sound generator |
| wd:Q1808578 | lutes |
| wd:Q2424752 | product |

# II. Practice with SPARQL queries

## Practice 1

#1.3 Please find all the relations between ‘violin’ and ‘violinist’ defined in WikiData.

**Hints**:

- WikiData SPARQL servicer: https://query.wikidata.org/
- WikiData Id Search Engine: https://www.wikidata.org/wiki/Special:Search
- For more info: https://www.wikidata.org/wiki/Wikidata:SPARQL_tutorial

# II. Practice with SPARQL queries

**The SPARQL of Practice #1.3**

```
#violin (Q8355)
#violinist (Q1259917)
SELECT ?property
WHERE
{ wd:Q8355 ?property wd:Q1259917. }
```

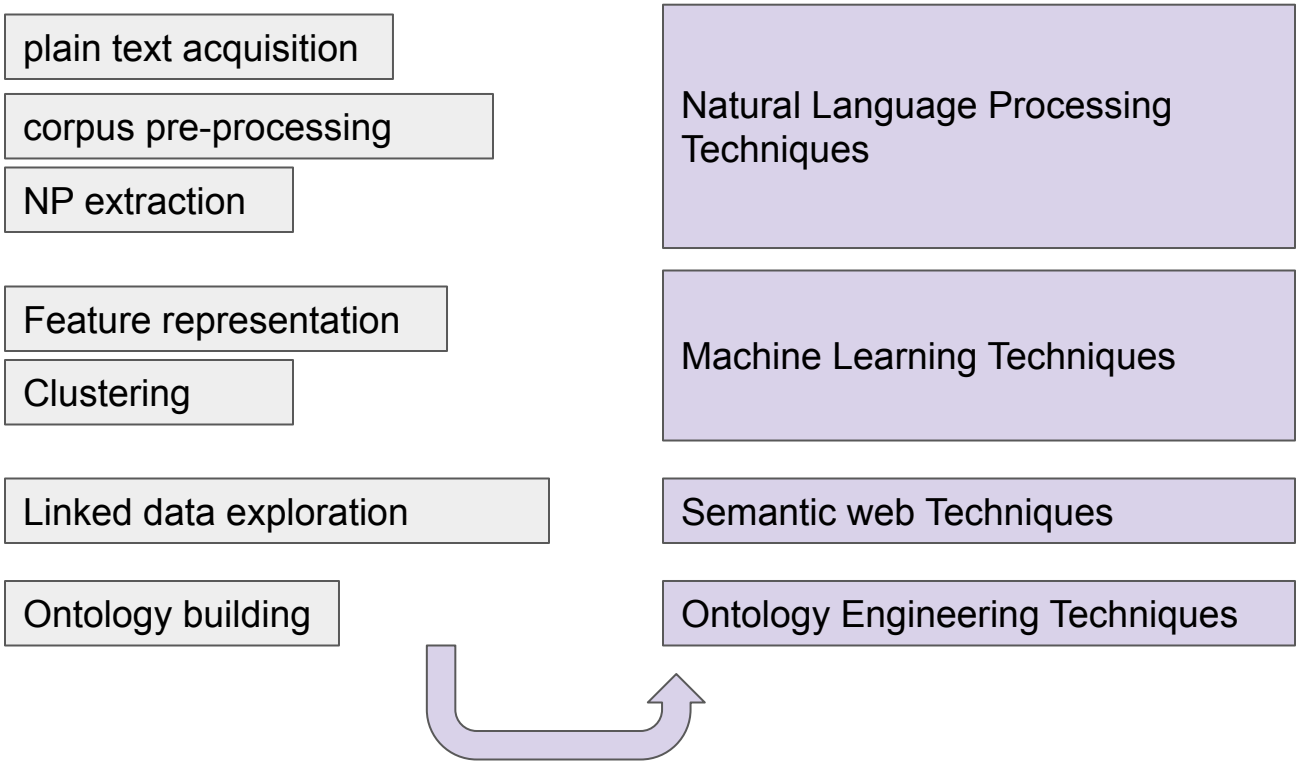
| property |
| --- |
| wdt:P1535 |
| wdt:P3095 |

# II. Practice with SPARQL queries

## The attributes of Practice 1

| | DBpedia | WikiData |
|---|---|---|
| synonyms | owl:sameAs<br>dbpedia2:names | NULL |
| hypernym | <http://purl.org/linguistics/gold/hypernym><br><http://purl.org/dc/terms/subject> | #subclass of (P279) |
| other relations | dbpedia2:related | #practiced by (P3095)<br>#used by (P1535) |

# III. Experiment on linked data



# Let’s do it step by step

# Reference

Source1: https://en.wikipedia.org/wiki/Google
Source2: The entity visualizer from spaCy.io
Source3: The Cross-Domain Linked Open Data Cloud from lod-cloud.net
Source4: http://data.doremus.org/ontology/
Source5: https://www.wikidata.org/wiki/Wikidata:WikiProject_Freebase
Source6: http://www.conceptnet.io/