Battles in Wikipedia

June 19, 2020

0.1 Battles in Wikipedia

First, I will get all the battles in WikiMedia's database, Wikidata, via SPARQL.

Alternatively, the query can be run in the browser via this link: https://query.wikidata.org/#prefix%20schema%3A%20%3Chttp%3A%2F%2Fschema.org%2F%3E%0APREFIX%

[2]:
```
from SPARQLWrapper import SPARQLWrapper, JSON
import pandas as pd

sparql = SPARQLWrapper("https://query.wikidata.org/sparql")
```

[3]:
```
# From https://www.wikidata.org/wiki/Wikidata:SPARQL_query_service/queries/
 ↪examples#Cats
sparql.setQuery("""
prefix schema: <http://schema.org/>
PREFIX wikibase: <http://wikiba.se/ontology#>
PREFIX wd: <http://www.wikidata.org/entity/>
PREFIX wdt: <http://www.wikidata.org/prop/direct/>

SELECT ?wikidata_id ?name ?article ?coordinates ?starttime ?endtime ?
 ↪participantLabel ?warLabel
WHERE {

 # Select all battles in the database
    ?wikidata_id wdt:P31 wd:Q178561.

 # Then select, when possible:
 # (by removing the OPTIONAL{} tag it becomes a mandatory requisite for the␣
 ↪query.

 # The coordinates:
    OPTIONAL{?wikidata_id wdt:P625 ?coordinates.}

 # The start time:
    OPTIONAL{?wikidata_id wdt:P580 ?starttime.}

 # The end time:
    OPTIONAL{?wikidata_id wdt:P582 ?endtime.}

 # The participants:
    OPTIONAL{?wikidata_id wdt:P710 ?participant.}

 # The war which it belonged to:
    OPTIONAL{?wikidata_id wdt:P361 ?war.
             ?war wdt:P31 wd:Q198}

    OPTIONAL {
        ?wikidata_id rdfs:label ?name filter (lang(?name) = "en") .
    }
    OPTIONAL {
      ?article schema:about ?wikidata_id .
      ?article schema:inLanguage "en" .
      FILTER (SUBSTR(str(?article), 1, 25) = "https://en.wikipedia.org/")
    }

  # Get the label for participants and war.

    SERVICE wikibase:label { bd:serviceParam wikibase:language␣
 ↪"[AUTO_LANGUAGE],en". }




}
""")
```

[4]:

```
sparql.setReturnFormat(JSON)
results = sparql.query().convert()
```

[5]:

```
results_df = pd.json_normalize(results['results']['bindings'])
results_df.head(4)
```

[5]:

```
  wikidata_id.type                       wikidata_id.value article.type  \
0              uri  http://www.wikidata.org/entity/Q169602          uri
1              uri  http://www.wikidata.org/entity/Q170113          uri
2              uri  http://www.wikidata.org/entity/Q170113          uri
3              uri  http://www.wikidata.org/entity/Q170148          uri

                                       article.value name.xml:lang name.type  \
0  https://en.wikipedia.org/wiki/Battle_of_Jakobs…            en   literal
1  https://en.wikipedia.org/wiki/Battle_of_Saint_…            en   literal
2  https://en.wikipedia.org/wiki/Battle_of_Saint_…            en   literal
3        https://en.wikipedia.org/wiki/Ragnar%C3%B6k            en   literal

                 name.value                             coordinates.datatype  \
0      Battle of Jakobstadt  http://www.opengis.net/ont/geosparql#wktLiteral
1  Battle of Saint Gotthard  http://www.opengis.net/ont/geosparql#wktLiteral
2  Battle of Saint Gotthard  http://www.opengis.net/ont/geosparql#wktLiteral
3                  Ragnarök                                              NaN

  coordinates.type                 coordinates.value  …      warLabel.value  \
0          literal  Point(25.870833333 56.503611111)  …  Great Northern War
1          literal    Point(16.21666667 46.94166667)  …  Austro-Turkish War
2          literal    Point(16.21666667 46.94166667)  …  Austro-Turkish War
3              NaN                               NaN  …                 NaN

  participantLabel.xml:lang participantLabel.type     participantLabel.value  \
0                       NaN                   NaN                        NaN
1                        en               literal  Principality of Wallachia
2                        en               literal             Ottoman Empire
3                       NaN                   NaN                        NaN

  starttime.datatype starttime.type starttime.value endtime.datatype  \
0                NaN            NaN             NaN              NaN
1                NaN            NaN             NaN              NaN
2                NaN            NaN             NaN              NaN
3                NaN            NaN             NaN              NaN

  endtime.type endtime.value
0          NaN           NaN
1          NaN           NaN
2          NaN           NaN
3          NaN           NaN

[4 rows x 22 columns]
```

Unfortunately, WikiMedia's database does not store battle result systematically.

This information seems to be available only via scraping of infoboxes.

The WikiProjects that worked with wars (https://www.wikidata.org/wiki/Category:WikiProject_Military_Histor and https://www.wikidata.org/wiki/Wikidata:WikiProject_WWII) apparently have not focused on capturing this kind of information.

Also, we only have "Participants" which does not tell us who is in which side.

Now let's do some cleaning:

[6]:

```
# Getting only the desired fields

results_df = results_df[["article.value", "name.value", "coordinates.value",␣
 ↪"participantLabel.value", "starttime.value", "endtime.value", "warLabel.
 ↪value"]]
```

```

# Time format:
list(results_df["endtime.value"])[-1]
```

[6]:
```
nan
```

As we can see, the time is almost as specified previously. The difference is the ”Z”, which indicates how precise the information is. This can be easily changed if needed.

[7]:
```
#Seeting proper names


results_df.columns = ["Wiki URL", "Battle name", "Battle location",␣
 ↪"Belligerents", "Battle start date", "Battle end date", "Part of war" ]
```

[8]:
```
urls = list(set(results_df["Wiki URL"]))[1:]
```

[11]:
```
import wikipedia
import time

names_to_pageids ={}
for name in urls:
    name = name.split("/")[4]
    if name not in  names_to_pageids:
        try:
            battle = wikipedia.page(name)
            names_to_pageids[name] = battle.pageid
            time.sleep(1)
        except:
           print("Failed for " + name)
```

[25]:
```
for key in names_to_pageids.keys():
    url = "https://en.wikipedia.org/wiki/" + name
    names_to_pageids[url] = names_to_pageids.pop(key)
```

[29]:
```
results_df["Page ID"] = results_df["Wiki URL"].map(names_to_pageids)
```

[30]:
```
# Exporting in an excel format
results_df.to_excel('Battles in WikiMedia.xlsx', sheet_name='Battles in␣
 ↪WikiMedia', index = False)
```

The rows are repeated, as each Belligerent is stored in a different line.

Values are missing for many cases.

Note: there are more Battles in WikiMedia’s database than in the Wikipedia list.

[31]:

```
print("There are {} different battles in WikiMedia's database".
 ↪format(len(set(results_df["Battle name"]))))
```

```
There are 8457 different battles in WikiMedia's database
```

[32]:

```
print("There are {} different battles in WikiMedia's database for which all␣
 ↪required information (but result) is available".format(len(set(results_df.
 ↪dropna()["Battle name"]))))
```

```
There are 156 different battles in WikiMedia's database for which all required
information (but result) is available
```

[33]:

```
results_df.dropna().to_excel('Battles in WikiMedia_only_full_battles.xlsx',␣
 ↪sheet_name='Battles in WikiMedia', index = False)
```