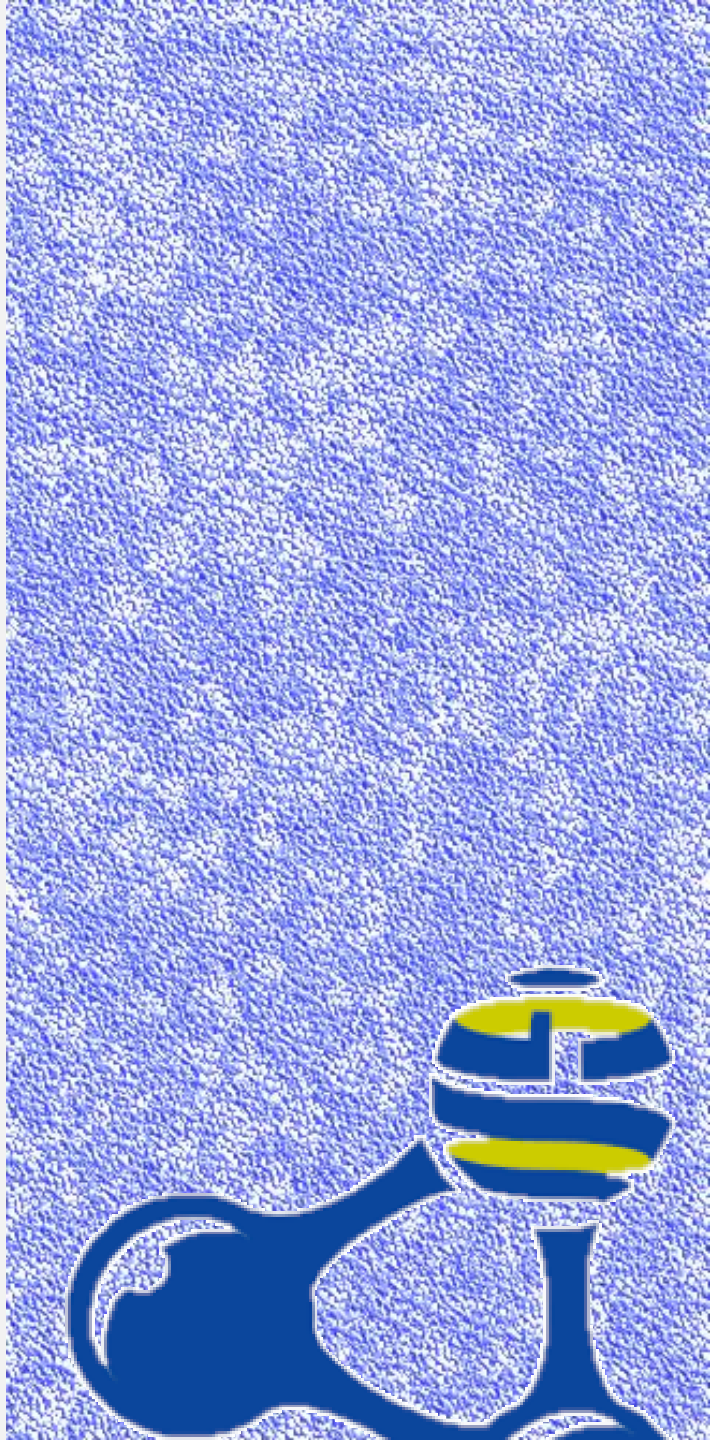
جامعة صفاقس
University of Sfax

EMBEDDING SPARQL IN TOOLS AND BOTS

Houcemeddine Turki
University of Sfax, Tunisia
LD4 Wikidata Affinity Group Weekly Meeting
24 August 2020



PARTNERS

WIKIMEDIA AND LIBRARIES USER GROUP

WIKIMEDIA MEDICINE

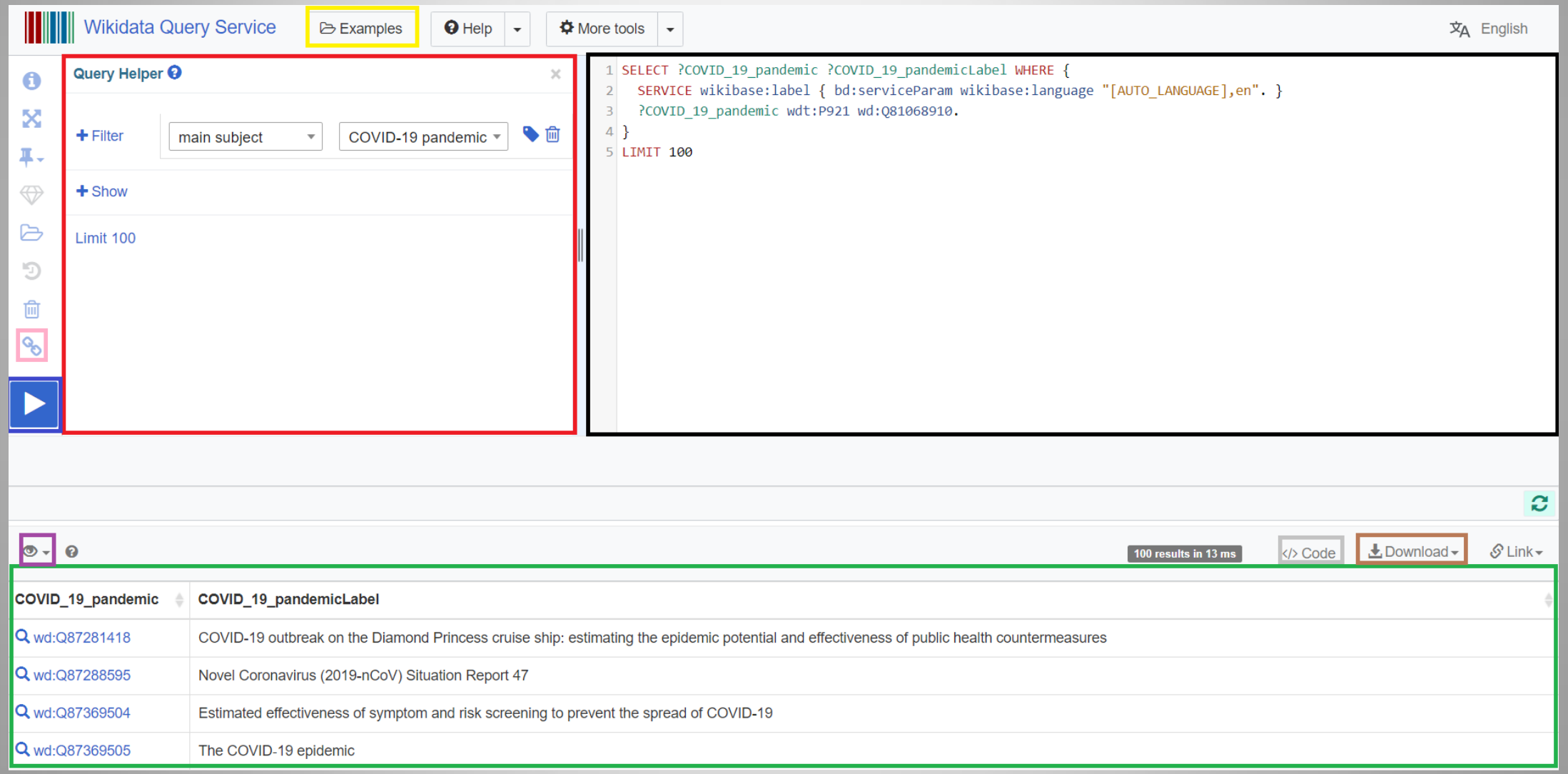
UNIVERSITY OF SFAX

SPARQL

- A semantic query language
- Applied to RDF graphs where the knowledge is represented as statements in the form of triples
- The skeleton of SPARQL is inspired from SQL query language
- The SPARQL endpoint of Wikidata is available at https://query.wikidata.org

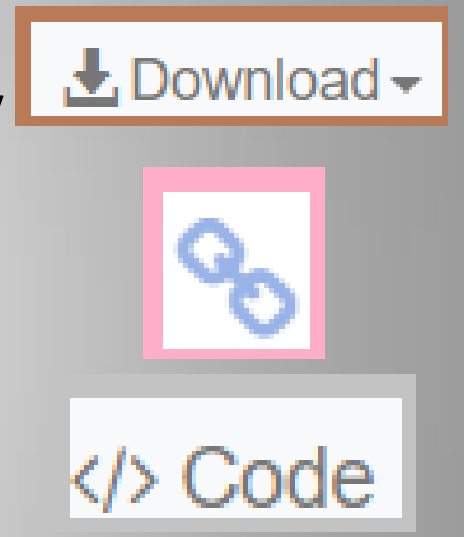
```
SELECT ?COVID_19_pandemic ?COVID_19_pandemicLabel WHERE {
  SERVICE wikibase:label { bd:serviceParam wikibase:language "[AUTO_LANGUAGE],en". }
  ?COVID_19_pandemic wdt:P921 wd:Q81068910.
}
LIMIT 100
```

SPARQL ENDPOINT OF WIKIDATA



EASY EXPORT OF RESULTS

- Download
  - Query results can be downloaded in a variety of formats including CSV and TSV
- Link
  - Short links can be generated for queries
- Code
  - Code to embed SPARQL queries in computer programs can be generated



EASY EMBEDDINGS OF QUERIES

URL HTML Wikilink PHP JavaScript (jQuery) JavaScript (modern) Java Perl Python

Python (Pywikibot) Ruby R Matlab listeria

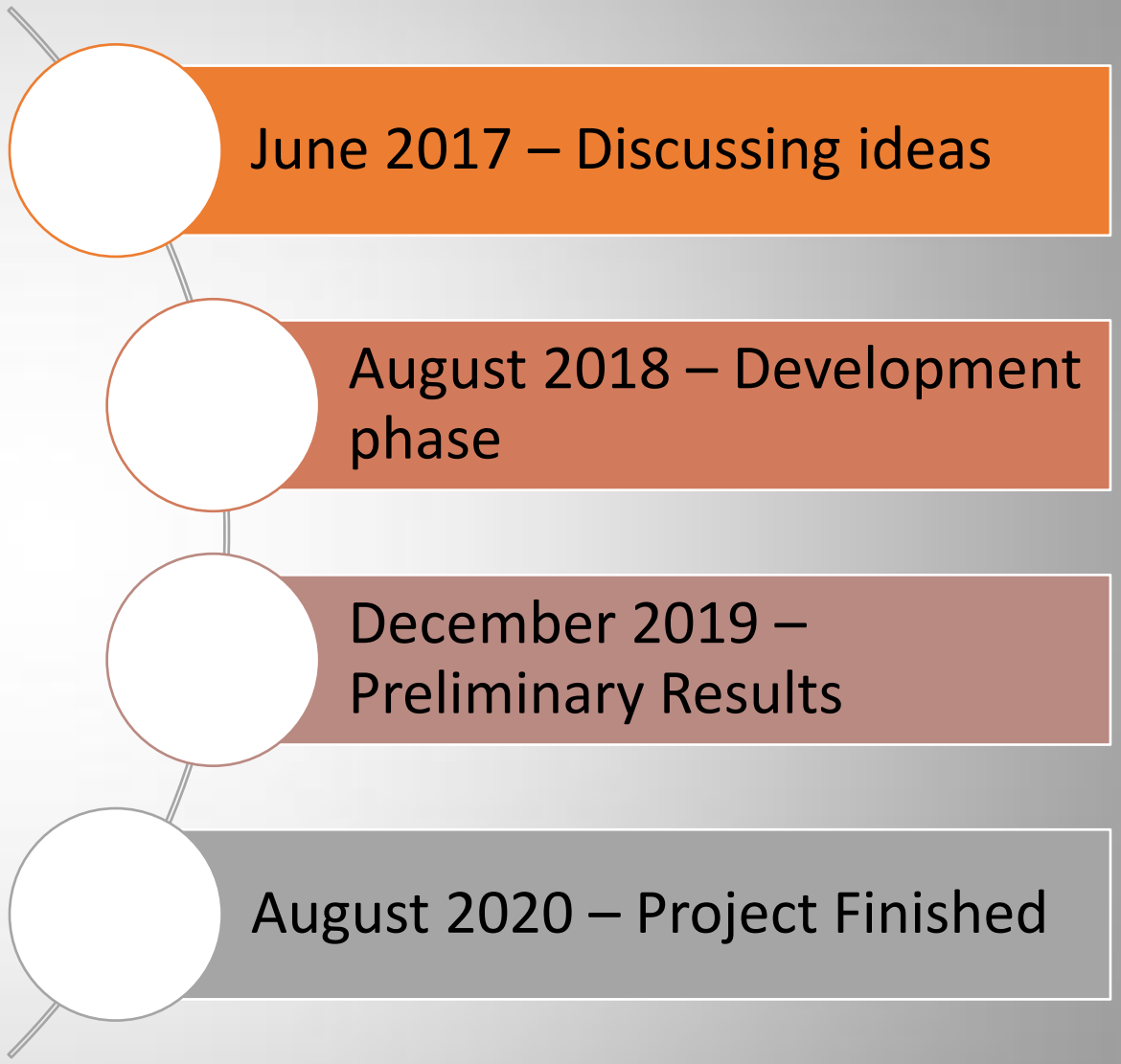
```
class SPARQLQueryDispatcher {
    constructor( endpoint ) {
        this.endpoint = endpoint;
    }

    query( sparqlQuery ) {
        const fullUrl = this.endpoint + '?query=' + encodeURIComponent( sparqlQuery );
        const headers = { 'Accept': 'application/sparql-results+json' };

        return fetch( fullUrl, { headers } ).then( body => body.json() );
    }
}

const endpointUrl = 'https://query.wikidata.org/sparql';
const sparqlQuery = `SELECT * WHERE {
  ?x wdt:P31 wd:Q17633526.
  }`;
```

WHY THIS IS USEFUL

- Extracting Wikidata statements to process
- Rule-Based Knowledge Graph Validation

PROJECT TIMELINE

Three years of work



WIKIPROJECT COVID-19

Processing data

- Adding reference support to Wikidata statements

Validating data

- Inferring Wikidata property constraints and statements
- Verifying the consistency of epidemiological data

INFERRING WIKIDATA PROPERTY CONSTRAINTS AND STATEMENTS

Task	Description
Defining the scheme of a Wikidata property	
T1	Identify common use cases of R: (C_x,C_y) pairs
T2	Identify inverse properties of R corresponding to each common use case: (C_x, R^{-1},C_y) statements
Identifying the deficiencies of the scheme	
T3	For each returned R^{-1}, identify R(X,Y) relations supported by references and corresponding to the most common (C_x, R^{-1}, C_y) statement but not available in Wikidata
T4	Identify R(X,Y) relations not corresponding to the most common scheme of R
Assessing the reference support of relations using the studied Wikidata property	
T5	Identify Wikidata properties used to define references for relations using R

VERIFYING THE CONSISTENCY OF EPIDEMIOLOGICAL DATA

Task	Description
Validating qualifiers of COVID-19 epidemiological statements	
V1	Verify *Z* as a date > November 01, 2019
V2	Verify *Q* as any subclass of (P279*) of medical diagnosis (Q177719)
Ensuring the cumulative pattern of *c*, *d*, *r*, and *t*	
V3	Identify *c*, *d*, *r* and *t* statements having a value in date *Z*+1 not superior or equal to the one in date *Z* (Verify if $d^Z \leq d^{Z+1}$, $r^Z \leq r^{Z+1}$, $t^Z \leq t^{Z+1}$, and $c^Z \leq c^{Z+1}$)
V4	Find missing values of *c*, *d*, *r* and *t* in date *Z*+1 where corresponding values in dates *Z* and *Z*+2 are equal
Validating values of epidemiological data for a given date	
V5	Identifying *c*, *d*, *r*, *h*, and *t* statements with negative values
V6	Identify *h* statements having a value superior to the number of cases for a date *Z*
V7	Identify *c* statements having a value superior or equal to the number of clinical tests for a date *Z*
V8	Identify *c* statements having a value inferior to the number of deaths for a date *Z*
V9	Identify *c* statements having a value inferior to the number of recoveries for a date *Z*
V10	Comparing the epidemiological variables of a general outbreak with the ones of its components
Validating case fatality rates	
V11	Comparing *m* with *d* / *c* for a date *Z*
V12	Missing *m* values with existing *d* and *c* for a date *Z*

RESULTS

Assignment	Findings
Inferring Wikidata property constraints and statements	• Added constraints to six biomedical Wikidata properties • Identified 11236 inconsistencies
Verifying the consistency of epidemiological data	• Identified 5639 inconsistencies • Identified 7116 missing statements

DEMO: ADDING REFERENCE SUPPORT TO WIKIDATA STATEMENTS

A Wikidata bot that includes

- An embedded SPARQL query to extract Wikidata relations lacking references
- A Biopython-based algorithm to find references for unsupported statements in PubMed Central
- An algorithm to add retrieved references using QuickStatements API

WORK TO BE PUBLISHED

- Houcemeddine Turki, Dariusz Jemielniak, Mohamed Ali Hadj Taieb, Jose Emilio Labra Gayo, Mohamed Ben Aouicha, Mus'ab Banat, Thomas Shafee, Eric Prud'Hommeaux, Tiago Lubiana, Diptanshu Das, and Daniel Mietchen on behalf of WikiProject COVID-19
- Using SPARQL to *validate* COVID-19 information in collaborative knowledge graphs: a study of Wikidata

FUNDING

WIKICRED GRANT INITIATIVE

MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

WIKIMEDIA FOUNDATION

THANK YOU

turkiabdelwaheb@hotmail.fr
+21629499418
User:Csisc
@csisc1994