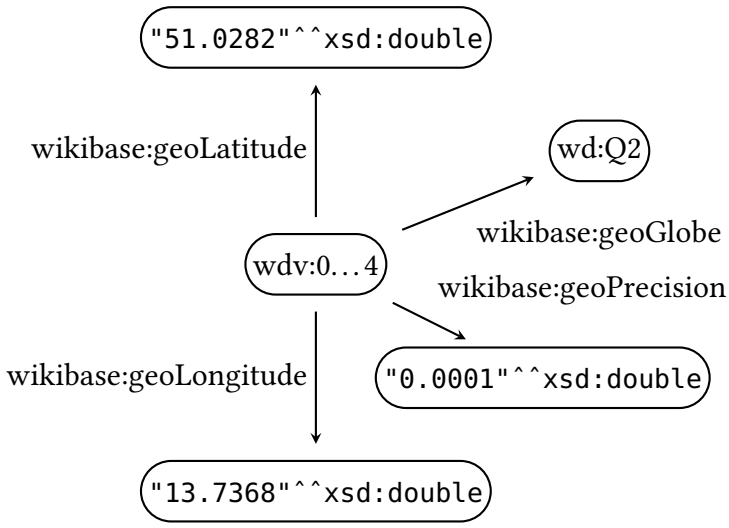
Exercise Sheet 11: Knowledge Graph Quality and Validation

Maximilian Marx, Markus Krötzsch

Knowledge Graphs, 2024-01-16, Winter Term 2023/2024

Exercise 11.1. The following extract from Wikidata shows how geographic coordinates are encoded. Develop a SHACL schema that validates statement values for geographic coordinates in Wikidata.



Hint: Refer to the RDF Dump Format description[1] for details on the encoding.

Exercise 11.2. Show that deciding whether a given RDF graph is valid with respect to some fixed ShEx schema is NP-hard by reducing from 3-colourability.

Hint: You can use the RDF Shape playground[2] to test ShEx validation.

Exercise 11.3. Participants and winners of sports tournaments are modelled in Wikidata using properties P1334 (“participant in”) and P2522 (“victory”).

Write a program that, using the Wikidata Query Service,[5] extracts the subgraph of Wikidata where there is an edge from vertex w to vertex v if v is a participant of some tournament with winner w, and produces as output two files containing

- the graph in METIS graph format (cf. Exercise sheet 1), and
- and a dictionary mapping every vertex ID to the English label of the corresponding Wikidata item (with each line being of the form `n,"l"`, where `n` is the vertex ID and `l` is the item label), respectively.

Exercise 11.4. Write a program that takes as input

- a directed graph in METIS format
- and a dictionary file in the format of Exercise 11.3 mapping vertex IDs to labels,

[1] https://www.mediawiki.org/wiki/Wikibase/Indexing/RDF_Dump_Format#Globe_coordinate

[2] http://rdfshape.weso.es

and outputs the vertex labels of the given graph, ordered by (decreasing) PageRank.

Modify the program from Exercise 11.3 to only consider tournaments that were (transitively) part of the 2018 FIFA World Cup (Q170645), and compute the PageRank of this graph. How do you interpret the results?

Hint: You can exploit the fact that most of the entries in the adjacency matrix are 0 by using *sparse matrices* as implemented by, e.g., SciPy.[1] NetworkX[2] offers `pagerank_scipy`,[3] an implementation of PageRank using sparse matrix arithmetic.

[1] https://www.scipy.org/

[2] https://networkx.github.io/

[3] https://networkx.github.io/documentation/stable/reference/algorithms/generated/networkx.algorithms.link_analysis.pagerank_alg.pagerank_scipy.html#networkx.algorithms.link_analysis.pagerank_alg.pagerank_scipy