Exercise Sheet 10: Datalog, VLog4j, and Property Graph

Maximilian Marx, Markus Krötzsch

Knowledge Graphs, 2020-01-07, Winter Term 2019/2020

Exercise 10.1. A Hamiltonian cycle in a directed graph is a directed cycle that visits each vertex exactly once. Show that for every $k \geq 1$, there is a Datalog query that finds a Hamiltonian cycle in the binary edge predicate of a graph containing exactly k vertices.

Is there also a fixed query deciding the existence of Hamiltonian cycles in such a graph with an arbitrary number of vertices?

Exercise 10.2. Which of the following graph patterns are expressible as (stratified) Datalog queries (all predicates mentioned are binary)? Explain your answer by either giving a Datalog query or by arguing why there is none.

1. Find nodes that are connected by an edge path of length ≥ 100
2. Find nodes that are connected by an edge path of length ≤ 100
3. Find nodes that are connected by an edge path of length $\neq 100$
4. Find nodes that are not connected by an edge path of length 100
5. In a graph with a parent predicate, find nodes with a common ancestor
6. In a graph with a parent predicate, find nodes that are cousins (of any degree)
7. Find nodes that are connected by predA but not by predB
8. Find nodes that are connected by an predA path, but not by an predB path
9. Find nodes that are connected by a path of nodes as in 7
10. Find nodes connected by an arbitrary path
11. Find nodes connected by an arbitrary path of even length

* 12. Check if the graph contains an even number of nodes

Exercise 10.3. DBpedia is a knowledge graph based on information extracted from Wikipedia. Use the VLog4j client[1] and the Wikidata[2] and DBpedia[3] SPARQL endpoints to integrate and compare the *parent* relations from DBpedia and Wikidata: Use SPARQL queries to fetch both *parent* relationships (for Wikidata, you can restrict to items with articles on English Wikipedia). Make sure your queries include a common feature that can be used for integration, e.g., the URL of the related Wikipedia article.[4] Lastly, use rules to compute the total number of (unique) relations found in both graphs, in Wikidata only, and in DBpedia only.

[1] https://github.com/knowsys/vlog4j/wiki/Standalone-client

[2] https://query.wikidata.org

[3] https://dbpedia.org/sparql

[4] DBpedia still stores `http` URLs, whereas Wikidata uses `https`. You can use `SUBSTR` and `BIND` in your SPARQL queries to align such URLs.

Exercise 10.4. Download and install Neo4j[5], or use the Neo4j Sandbox[6].

Use the `:play movies` command to load the movie example data set. Write Cypher queries that find

1. all actors who have co-starred in two movies,
2. for every actor, the length of the shortest path (along any relationship type) connecting this actor to Kevin Bacon,
3. pairs of persons and movies where the person has at least two relationships of distinct relationship types to the movie, and
4. the number of undirected triangles along any relationship type. How often is each triangle counted?

[5]https://neo4j.com/download/
[6]https://neo4j.com/sandbox-v3/