SPARQL Exam Preparation: Subqueries and Property Paths

Fall 2023

SPARQL Subqueries and Property Paths

SPARQL - Subqueries

Example: Select all authors, by whom they are influenced, and all the influencers' notable works.

- Subqueries are a way to embed SPARQL Queries within other queries.
- Result is achieved by first evaluating the inner query.

Code Example:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX dbo: <http://dbpedia.org/ontology/>

SELECT ?author ?influencer ?work
FROM <http://dbpedia.org/>
WHERE {
    SELECT ?author ?influencer
    FROM <http://dbpedia.org/>
    WHERE {
        ?author rdf:type dbo:Writer ;
                dbo:influencedBy ?influencer .
    }
    ?influencer dbo:notableWork ?work .
}
LIMIT 100
```

Questions and Answers:

1. **Q: What is the purpose of subqueries in SPARQL?**
   A: Subqueries allow embedding SPARQL queries within other queries, which helps in breaking down complex queries. The outer query can use the results of the inner query.
2. **Q: In the provided example, what does the inner query do?**
   A: The inner query selects authors and their influencers from the RDF graph, and the outer query then finds notable works of those influencers.

SPARQL - Property Paths

Definition: A property path is a possible route through an RDF graph between two graph nodes.

Examples:

- **Trivial Case:** property path of length 1, i.e., a triple pattern.
- **Alternatives:** match one or both possibilities.

```
{ :book1 dc:title|rdfs:label ?displayString }
```

- **Sequence:** property path of length ¿ 1.

```
{ ?x foaf:knows/foaf:name ?name . }
```

- **Inverse Property Paths:** reversing the direction of the triple.

```
{ ?x foaf:mbox <mailto:alice@example> .
  <mailto:alice@example> ^foaf:mbox ?x . }
```

Questions and Answers:

1. **Q: What is a property path in SPARQL?**
   A: A property path is a route through an RDF graph connecting two nodes via a sequence of properties, which may involve alternative or inverse paths.
2. **Q: Give an example of an alternative property path.**
   A: { `:book1 dc:title|rdfs:label ?displayString` } finds either the title or label of `:book1`.
3. **Q: How does SPARQL define an inverse property path?**
   A: An inverse property path reverses the direction of a relationship. For example, { `<mailto:alice@example>` $^{f}oaf : mbox?x\}findsentitieswithan$`foaf:mbox`$linkingtoAlice'semail.$

SPARQL - Property Paths (Inverse Path Sequences)

Inverse Path Sequences:

```
{ ?x foaf:knows/^foaf:knows ?y
  FILTER (?x != ?y) . }
```

Questions and Answers:

1. **Q: What does the code { ?x foaf:knows/**$^{f}oaf : knows?yFILTER(?x! = ?y)\}represent?$
   A:$Itretrievespairsofnodes$`?x`and`?y`$where$`?x`$knowssomeonewhoknows$`?y`$, excludingcaseswhere$`?x`$isthes$

2. **Q: Why is the FILTER (?x != ?y) used in inverse path sequences?**
   A: The `FILTER` condition ensures that the subject and object are distinct, preventing self-referential relationships in the result.

SPARQL - Property Paths (Arbitrary Length Match)

Arbitrary Length Match:

```
{ ?x foaf:mbox <mailto:alice@example> .
  ?x foaf:knows+/foaf:name ?name . }
```

Questions and Answers:

1. **Q: What does the + symbol represent in SPARQL property paths?**
   A: The `+` symbol denotes a path of arbitrary length, requiring at least one traversal. For example, `foaf:knows+` finds people known directly or indirectly by `?x`.

2. **Q: Explain the code { ?x foaf:knows+/foaf:name ?name }.**
   A: It finds the names of all individuals connected to `?x` via one or more `foaf:knows` relationships.

SPARQL - Property Paths (Multiple Concepts)

Inverse Path Sequences:

```
{ ?x foaf:knows/^foaf:knows ?y FILTER (?x != ?y) }
```

Arbitrary Length Match:

```
{ ?x foaf:mbox <mailto:alice@example> .
  ?x foaf:knows+/foaf:name ?name . }
```

Negated Property Paths:

```
{ ?x !(rdf:type|^rdf:type) ?y . }
```

Questions and Answers:

1. **Q: What is a negated property path in SPARQL?**
   A: A negated property path excludes specific relationships in a query. For example, { `?x !(rdf:type|`$^{r}df : type)?y\}findspairswhere$`?x`$and$`?y`$donothaveardf:typerelationshipin$

2. **Q: What is the purpose of using !(rdf:type—**$^{r}df : type)inaquery?$
   A:$Itrestrictsresultstopairsofentitiesthatarenotconnectedby$`rdf:type`$inanydirection.$

SPARQL - Complex Example with Property Paths

Complex Query Using Property Paths:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX : <http://dbpedia.org/resource/>
PREFIX dbo: <http://dbpedia.org/ontology/>

SELECT ?influencedByInfluencers
FROM <http://dbpedia.org/>
WHERE {
    :Jules_Verne dbo:influencedBy/^dbo:influencedBy ?influencedByInfluencers
    FILTER (?influencedByInfluencers != :Jules_Verne).
}
```

Questions and Answers:

1. **Q: Explain the use of dbo:influencedBy/**$^{d}bo : influencedByinthisquery.$
   A:$Thispropertypathfindsentitiesthatinfluencedthoseinfluencedby Jules Verne, creatinganindirectinf$

2. **Q: What is the purpose of the filter FILTER (?influencedByInfluencers != :Jules**$_{V}erne$**)?**
   A:$Itremovesanyresultwhere$`?influencedByInfluencers`$isJules Vernehimself, ensuringonlydistinctinflue$

SPARQL Review

- **Q: What does SPARQL stand for?**
  A: SPARQL stands for "SPARQL Protocol and RDF Query Language."
- **Q: What are the four main types of SPARQL queries?**
  A: SELECT, ASK, DESCRIBE, and CONSTRUCT.
- **Q: True or False: SPARQL is only a query language.**
  A: False. SPARQL also includes protocol and result formats for HTTP queries.
- **Q: What is a Federated SPARQL Query?**
  A: A federated query allows querying across multiple SPARQL endpoints, useful for integrating distributed RDF data.