### Reverse Engineering of SPARQL Queries using Examples

Shweta Yakkali | sxy9369@rit.edu  
Advisor: Dr. Carlos Rivero | crrvcs@rit.edu  
Rochester Institute of Technology

#### PROBLEM DESCRIPTION
- An enormous range of broad interest public data is available by Semantic Web public system interfaces for ordinary users.
- A bottleneck in making effective use of query interfaces is due to Web Identifiers (URIs). The URIs provide a low-level representation which is not easy for users.
- SPARQL is a powerful query language for querying Semantic data i.e. RDF Triples. The obstacle is users need to be familiar with the syntax and the RDF triple pattern.

#### SOLUTION
- A system which queries RDF data with positive and negative examples and generates a query based on the examples.
- The system helps unfamiliar users with the dataset structure and SPARQL syntax to generate desired query and result set.

#### RDF and SPARQL
- RDF (Resource Description Framework) provides resources in the form of `<subject – predicate – object>` expressions.
- SPARQL is a query language to retrieve and manipulate data stored in RDF format.

#### RESULTS
The system generates a reverse engineered query and results for the following cases:

**Case 1: List of Spanish Speaking Countries:**
- Positive and negative examples:
  - Chile +Bolivia +Venezuela +Spain -Brazil -Angola
- Reverse engineered SPARQL query:
  - `SELECT * WHERE { ?x a <http://dbpedia.org/class/yago/WikisourceSpanish-speakingCountriesAndTerritories> ;
  \}$`

**Case 2: User-based filtering on initial results**
- Positive and negative examples:
  - Deepika_Padukone +Ranveer_Singh +Preity_Zinta -Fawad_Khan -Gal_Gadot -Emma_Watson
- Reverse engineered SPARQL query:
  - `SELECT * WHERE { ?x a <http://dbpedia.org/class/yago/IndianPeople> ;
  \}$`

**Case 3: OPTIONAL keyword due to NULL values:**
- Positive and negative examples:
- Reverse engineered SPARQL query:
  ?x <http://dbpedia.org/property/Place> "Oxford" .
  \}$`

#### SYSTEM ARCHITECTURE
- The SPARQLByE System consists of:
  - User Interface Module
  - Reverse Engineering Module
  - RDF Data Service SPARQL end-point

#### CONCLUSION
- Querying RDF data with positive and negative examples is easier and effective.
- SPARQLByE provides concurrent execution across all browsers and scales well for all open source Semantic Web data.
- The inaccurate or less relevant results can be reduced by weighted example filtering.

#### REFERENCES