

Projects Presentation

Riccardo Tommasini

Premise

- Projects are OPTIONAL
- Projects are an opportunity to dirt your hands
- Knowledge Engineering is a collaborative work -> groups are encouraged
- Projects MUST be delivered before October 1st

Project Structure

- Projects are divided into three Tasks of increasing complexity
 - Knowledge Engineering
 - Data Linking
 - Semantic Web Application

Project Structure

- Projects are divided into three Tasks of increasing complexity
 - Knowledge Engineering (1,5)
 - Data Linking (1,5)
 - Semantic Web Application (1,5)

Knowledge Engineering

- Each group MUST
 - design an ontological model about a given application domain
 - provide competency questions, documentation, and examples,
 - present the design motivating their choices.

Knowledge Engineering

- Evaluation Points
 - REUSE of existing vocabularies and ontologies
 - Follow best-practice [1]
 - USE of W3C Standards, RDF, OWL2,
 - USE of existing resources, e.g., Widoco [2]

Data Linking

- Each group **MUST**
 - **complete the Knowledge Engineering TASK**
 - convert/annotate a given dataset into RDF using the the ontology previously developed
 - translate the competency questions into SPARQL queries and provide evidence of the results
 - publish the dataset as Linked Data

Data Linking

- Evaluation Points
 - Linked Data Principles [3]
 - Automated annotation methods, RML/Ontop
 - Multiple Publication Methods, SPARQL, DUMP, REST
 - USE of annotation vocabularies, VOID, DCAT, SPARQL-SD,
 - Linking to other vocabularies

Semantic Web Application

- Each group **MUST**
 - **complete the Data Linking TASK**
 - use the annotated dataset to create value in a Web Application
 - Valid SW Apps are just Web App, that exploit KE and LD tools

Semantic Web Application

- Evaluation Points
 - Linked Data Principles (again) [3]
 - How does it create value?

Project Proposals

- Project Proposals are “Full-Stack”,i.e., they cover all the three aspects
- BUT the annotated dataset is provided group-wise
- Student can propose their ideas too

Proposals: ColorWave

- An application should receive an URL of an Image, extract the colours and shapes from the image and return the RDF annotation of such image.
- Similarly to <https://www.degraeve.com/color-palette/>, but the results are in RDF
- Alternatively one can use D3.js to provide “live” descriptions of coloured shapes

Proposals: LinkedContainers

- Converting Dockerfile or Docker Images in RDF: the application should receive a link to a Docker Image from DockerHub, retrieve the data via the API and convert them in RDF.
 - Similarly to <https://microbadger.com/images/ubuntu> but the results are in RDF
- Alternatively one can provide a text editor where to write a Dockerfile and build an image locally.

Proposals: Foodle Maps

- Provide a Google Maps extensions that shows the location of particular dishes
- Imagine you're hungry and you're looking for a burrito, a veggie burrito.
- The application should find the restaurants serving such dish and locate the burritos accordingly

Proposals: PutArtApart

- Provide a Google Maps extensions that shows the location of particular artworks
- Imagine you're in a city and you'd like to know which patins are around.
- The application should find the museums (e.g.) showing such patins and locate the paints accordingly.

Proposals: Research

- schema.org is a shared project across the biggest search engine vendors, i.e., Google, Bing, Yahoo!, and Yandex.
- It contains a variety of terms to annotate web pages semantically
- schema.org extensions follow a prescribed process. The group interested in pursuit this project should focus on this process.
- DOMAINS:
 - Streaming Data and Events
 - Microservices and Cloud

References

- [1] https://protege.stanford.edu/publications/ontology_development/ontology101.pdf
- [2] <https://github.com/dgarijo/Widoco>
- [3] <https://www.w3.org/DesignIssues/LinkedData.html>