Linking E-Learning Ontology Concepts with NLP Algorithms

Dmitry Mouromtsev, Fedor Kozlov, Liubov Kovriguina and Olga Parkhimovich
Laboratory ISST @ ITMO University, St. Petersburg, Russia

16th FRUCT Conference, 27-31 October 2014, Oulu, Finland
Introduction

● E-learning ontology describes relations between educational resources (course, module, lecture, task, term),
● Use semantics to make education materials reusable and flexible,
● We need to provide tools for tutors to improve their courses have to extract and interlink relevant educational materials (e.g., associate terminology in lectures and tests) and gather analytics
Some details...

1. The developed E-Learning ontology allows to create interdisciplinary relations between courses → a system may advice to repeat terms not only from the current course, but from the previous courses also (e.g., term “Vector” has links to courses “Vector algebra” and “Physics”);

2. When terms are linked to tasks (via “hasTerm” property), it is possible to get the data like
   - which terms present only in lectures, but not in tasks (teachers add new tasks),
   - which terms present only in tasks, but not in lectures (teachers add new explanatory content),
   - statistics;

3. A way to get balanced educational content.
Goal

● To develop the test ontology,
● To convert tasks from XML to RDF,
● To extract terms from tasks,
● To map tasks with ontology domain terms via extracted terms.
ECOLE: Front-end

- Ontology-based e-learning system
- User friendly interface
- Based on Django framework
ECOLE: Back-end

- Collection of educational materials from different open resources (DBpedia BNB),
- Analytics tools,
- Based on the Information Workbench platform.
The ontology of education resources

- Extends AIISO ontology for education process and structure
- Uses BIBO for bibliographic resources
- Uses MA-ONT for media resources
The ontology of tests

- Developed to describe the content of tests,
- Extends top-level ontology of the system,
- 12 classes,
- 10 object properties,
- 6 datatype properties,
Lemmatization of Domain Terms

- Extracts lemmas from subject term labels,
- Uses NLP procedures and dictionaries to generate lemmas,
- Stores lemmas in triples.
Terms extraction and linking

E-Learning System

Back-end provider

Collect tasks using SPARQL
Forms the plain text content

Extract terms with NLP procedures

Dictionaries

NLP module

Search system terms with the same lemmas

Link terms with tasks
Send new system terms for validation
NLP Algorithm

- Uses Part-of-Speech patterns
- Grammatical agreement is specified in the patterns

Dictionary

- Words+Lemmas+Morphology+Some semantic tags
- Superlemmas (a common lemma is generated for lexical variants)
- Derivation paradigms (a common lemma is generated for words with the same root)

Tokenization → Morphological and Semantic Tagging → Extracting Candidate Terms → Producing the Canonical Form of the Term + Lemma(s)

Patterns Set
New System Terms

- If the extracted candidate term doesn't match any system terms, it is included as a new system term,
- New system terms are validated via SPARQL queries to DBpedia

```
SELECT DISTINCT ?term {
  VALUES ?subject {
    category:Concepts_in_physics
    category:Physical_optics
    category:Optics
  }
  {?name_uri dbpedia-owl:wikiPageRedirects ?term ;
   rdfs:label ?label .}
  UNION
  { ?term rdfs:label ?label } FILTER( STR(?label) = "Diffraction")
}
```
Implementation: Test parsers

- Converts tests of the course from XML to RDF,
- Uses Information Workbench XMLProvider to automatically update,
- Describes mapping using XPath functions.
Implementation: NLP module

- Uses dictionaries in NooJ format:
  \[<\text{LEMMA}>+<\text{PART OF SPEECH TAG}>+<\text{INFLECTIONAL PARADIGM}>+<\text{OTHER ANNOTATIONS}>\]
  
  air, N+FLX=TABLE
  Michelson, N+ProperName

- English NooJ resources are reused, Russian lexical resources are original,
- A separate procedure implemented in Python is launched for lemmatization and term extraction.
# Evaluation and Results

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Percent of linked tasks, %</td>
<td>95</td>
</tr>
<tr>
<td>Percent of non-linked tasks, %</td>
<td>5</td>
</tr>
<tr>
<td>Number of different candidate terms</td>
<td>155</td>
</tr>
<tr>
<td>Number of manually extracted terms</td>
<td>30</td>
</tr>
<tr>
<td>Percent of system terms, matched by candidate terms, %</td>
<td>50</td>
</tr>
<tr>
<td>Percent of candidate terms, matched by system terms, %</td>
<td>8</td>
</tr>
<tr>
<td>Percent of candidate terms to be included to the system terms after the validation procedure, %</td>
<td>6</td>
</tr>
<tr>
<td>Percent of false candidates, %</td>
<td>86</td>
</tr>
</tbody>
</table>
A ladder is 5m long. How far from the base of a wall should it be placed if it is to reach 4m up the wall?

Nothing to extract!!!
Conclusion

- The ontology of tests, dictionaries and patterns, test parsers, etc. have been developed,
- The tasks of the test have been linked with system terms,
- Statistics gathering module has been developed.
Future Work

- Improve term extraction procedure by adding parallel texts of tasks,
- Process units of measure in tasks to predict “hidden terms”,
- Use relations between subject terms to improve the quality of term extraction procedure,
- Refine term knowledge rating by replacing it by the proper ranking formula.
Thank you!!!

The front-end of the e-learning system: http://ecole.ifmo.ru

Example of subject terms analytics for module "Interference and Coherence":
http://openedu.ifmo.ru:8888/resource/Phisics:m_InterferenceAndCoherence?analytic=1

The source code: https://github.com/ailab.itmo/linked-learning-solution