MOOCLink: Building and Utilizing Linked Data from Open Courseware

Sebastian Kagemann

Arizona State University—Polytechnic Campus

CRA-W Distributed Research Experience

Mesa, Arizona 85212, USA

sakagema@indana.edu

Srividya Kona Bansal

Arizona State University—Polytechnic Campus

Department of Engineering

Mesa, Arizona 85212, USA

srividya.bansal@asu.edu

*Abstract*—Linked Data is an emerging trend on the web with top companies such as Google, Yahoo and Microsoft promoting their own means of marking up data semantically. Despite the increasing prevalence of Linked Data, there are a limited number of applications that implement and take advantage of its capabilities, particularly in the domain of education. We present a project, MOOCLink, which aggregates online courses as Linked Data and utilizes that data in a web application to discover and compare open courseware.

Keywords—linked data; education; ontology building

# Introduction (*Heading 1*)

“Linked Data is simply about using the Web to create typed links between data from different sources.” [1] This data may vary in source location, size, subject-matter and how congruously it is structured. Despite these differences, typed links can be made uniformly and explicitly, opening up opportunities for applications that were previously impractical.

Linked Data is relatively unexplored in the domain of education. Although there are several data models for structuring educational data as well as repositories adopting these models [citation needed], Linked Data-driven educational applications are far and few between. As a result, initiatives such as the LinkedUp Challenge [citation needed] have surfaced to encourage innovative applications focused on open educational data.

Massive Open Online Courses or MOOCs are online courses accessible to anyone on the web. Hundreds of institutions have joined in an effort to make education more accessible by teaming up with MOOC providers such as Coursera and edX. Delivering course content through lecture videos as well as traditional materials such as readings and problem sets, MOOCs encourage interactivity between professors and students around the world through discussion forums and graded assessments.

Coursera, a leading MOOC provider, offers a RESTful API for all information associated with their course details pages. This includes properties such as a courses’s title, instructor, syllabus, etc. Although Coursera course catalog data is structured and easily accessible as JSON, there is no option to retrieve and use it in a Linked Data format such as the Resource Description Framework (RDF). Moreover, there is little to no Linked Data available for MOOCs or an ontology that encompasses information unique to MOOCs. <<Additionally, the number of MOOC providers is growing >>

In order to incorporate MOOC data into the Linked Data cloud as well as demonstrate the potential of Linked Data in education, we propose to (i) build or extend an RDF ontology that denotes MOOC properties and relationships (ii) use our ontology to generate Linked Data from multiple MOOC providers and (iii) implement this data in a practical web application that allows users to discover courses across different MOOC providers.

# Background

## Resource Description Framework

As part of the initiative to link open data, several models for linking data have arisen. The most popular of these is the Resource Description Framework (RDF), which encodes data as subject, predicate, object triples. The subject and object of a triple are both Uniform Resource Identifiers (URIs) while the predicate specifies how the subject and object are related, also using a URI. Our Linked Data is presented as RDF/XML, an XML syntax for RDF.

<Should I explain what an ontology is?>

## Simple Protocol and RDF Query Language

Simple Protocol and RDF Query Language or SPARQL is an RDF query language that allows users to retrieve and manipulate data stored as RDF. SPARQL is used as MOOCLink’s query language for populating pages with course information. A SPARQL endpoint for our data can be accessed at <link>.

## Linked Data Principles

Tim Berners-Lee published a set of rules for publishing data on the Web so that data becomes part of a global space in which every point is connected [citation needed]. The rules are as follows:

1. Use URIs as names for things
2. Use HTTP URIs so that people can look up those names
3. When somone looks up a URI, provide useful information (RDF, SPARQL)
4. Include links to other URIs, so that they can discover more things

These principles provide a basis for contributing to a Linked Data cloud in which a variety of datasets from different fields of human knowledge are interconnected.

## Linked Education Data

Many models have been devised for structuring educational data, among the most popular are the IEEE Learning Object Metadata (LOM) specification and Sharable Content Object Reference model (SCORM). <Unique traits of LOM [citation]> <Unique traits of SCORM [citation]> Rather than repurposing one of these specifications which would require a full rewrite in terms of the Web Ontology Language or OWL, we choose to extend an existing Linked Data vocabulary, Schema.org to include more properties unique to open courseware.

In 2013, the Learning Resource Metadata Initiative (LRMI) specification was incorporated into Schema.org’s vocabulary for tagging educational content [citation needed]. The properties added in this adoption introduced fields for online course details including the type of learning resource, time required, and so on. While there was significant overlap between LRMI’s additions to Schema.org, Schema.org’s Creative Work properties and MOOC course details like those provided in Coursera’s API, several crucial missing data fields such as syllabus details, course difficulty, and predicates linking courses to other objects, justified our decision to extend the vocabulary.

## Building Ontologies

After determining that there was no educational resource ontology fully suitable for MOOC data, we extend Schema.org’s ontology. In order to support uniformity in our data, we use RDF/XML from ontology creation to final data generation. <in retrospect, Schema.org maintains an OWL version that we could have used> Schema.RDFS.org hosts an RDF/XML version of Schema.org’s ontology which we import into Stanford Protégé to extend with additional types and properties.

Identify applicable sponsor/s here. If no sponsors, delete this text box (*sponsors*).

# MOOCLink

MOOCLink is a web application which aggregates online courses as Linked Data and utilizes that data to discover and compare online courseware. The following outlines our approach including choosing our providers, data modeling, data generation, and the development of the web component.

## MOOC Providers

Coursera is the largest MOOC provider in the world with 7.1 million users in 641 courses from 108 institutions as of April 2014 [citation needed]. These courses span 25 categories including 4 subcategories of computer science. All course details are retrievable by HTTP GET method using Coursera’s RESTful course catalog API and returned in JSON. <free with paid certifications>

edX is another premier MOOC provider with more than 2.5 million users and over 200 courses as of June 2014 [citation needed]. edX courses are distributed among 29 categories, many of which overlap with Coursera’s. edX does not provide an API for accessing their course catalog, however, as of June 2013, edX’s entire platform is open-source. <completely free>

Udacity, founded by Google VP, Sebastian Thrun, is a vocational course-centric MOOC provider with 1.6 million users in 12 full courses and 26 free courseware as of April 2014 [citation]. The majority of Udacity courses are within the field of computer science. Udacity does not provide an API for accessing their course catalog data.

## Data Model

Schema.org is organized as a hierarchy of types, each associated with a set of properties. CreativeWork is Schema.org’s type for generic creative work including books, movies, and now educational resources. The LRMI specification adds properties to CreativeWork including the time it takes to work through a learning resource (timeRequired), the typical age range of the content’s intended audience (typicalAgeRange), as well as specifying properties previously available in Schema.org’s CreativeWork type like the subject of the content (about) and the publisher of the resource (publisher).

CreativeWork provides a base type for our ontology extension, which adds types Course, Session and Category and their associated properties tailored to MOOC data. The extension is made in Stanford Protégé, which we use to import the Schema.org vocabulary provided in RDF/XML at Schema.RDFS.org. From the GUI of Protégé, types and properties are added as well as sample individuals to be used as a model for data generation. The hierarchy of our ontology extension is outlined in the Implementation section of this paper.

## Data Generation

Coursera HTTP requests

edX and Udacity crawlers using Scrapy

Jena conversion from JSON to RDF

## Web Application

Fuseki back-end

SPARQL queries

UI design

JSPs and Servlets / TomCat

# Implementation

After the text edit has been completed, the paper is ready for the template. Duplicate the template file by using the Save As command, and use the naming convention prescribed by your conference for the name of your paper. In this newly created file, highlight all of the contents and import your prepared text file. You are now ready to style your paper; use the scroll down window on the left of the MS Word Formatting toolbar.

## Ontology Schema

The template is designed so that author affiliations are not repeated each time for multiple authors of the same affiliation. Please keep your affiliations as succinct as possible (for example, do not differentiate among departments of the same organization). This template was designed for two affiliations.

### For author/s of only one affiliation (Heading 3): To change the default, adjust the template as follows.

#### Selection (Heading 4): Highlight all author and affiliation lines.

#### Change number of columns: Select the Columns icon from the MS Word Standard toolbar and then select “1 Column” from the selection palette.

#### Deletion: Delete the author and affiliation lines for the second affiliation.

### For author/s of more than two affiliations: To change the default, adjust the template as follows.

#### Selection: Highlight all author and affiliation lines.

#### Change number of columns: Select the “Columns” icon from the MS Word Standard toolbar and then select “1 Column” from the selection palette.

#### Highlight author and affiliation lines of affiliation 1 and copy this selection.

#### Formatting: Insert one hard return immediately after the last character of the last affiliation line. Then paste down the copy of affiliation 1. Repeat as necessary for each additional affiliation.

#### Reassign number of columns: Place your cursor to the right of the last character of the last affiliation line of an even numbered affiliation (e.g., if there are five affiliations, place your cursor at end of fourth affiliation). Drag the cursor up to highlight all of the above author and affiliation lines. Go to Column icon and select “2 Columns”. If you have an odd number of affiliations, the final affiliation will be centered on the page; all previous will be in two columns.

## Identify the Headings

Headings, or heads, are organizational devices that guide the reader through your paper. There are two types: component heads and text heads.

Component heads identify the different components of your paper and are not topically subordinate to each other. Examples include Acknowledgments and References and, for these, the correct style to use is “Heading 5”. Use “figure caption” for your Figure captions, and “table head” for your table title. Run-in heads, such as “Abstract”, will require you to apply a style (in this case, italic) in addition to the style provided by the drop down menu to differentiate the head from the text.

Text heads organize the topics on a relational, hierarchical basis. For example, the paper title is the primary text head because all subsequent material relates and elaborates on this one topic. If there are two or more sub-topics, the next level head (uppercase Roman numerals) should be used and, conversely, if there are not at least two sub-topics, then no subheads should be introduced. Styles named “Heading 1”, “Heading 2”, “Heading 3”, and “Heading 4” are prescribed.

## Figures and Tables

#### Positioning Figures and Tables: Place figures and tables at the top and bottom of columns. Avoid placing them in the middle of columns. Large figures and tables may span across both columns. Figure captions should be below the figures; table heads should appear above the tables. Insert figures and tables after they are cited in the text. Use the abbreviation “Fig. 1”, even at the beginning of a sentence.

1. Table Type Styles

| Table Head | Table Column Head | | |
| --- | --- | --- | --- |
| Table column subhead | Subhead | Subhead |
| copy | More table copya |  |  |

1. Sample of a Table footnote. (*Table footnote*)
2. Example of a figure caption. (*figure caption*)

Figure Labels: Use 8 point Times New Roman for Figure labels. Use words rather than symbols or abbreviations when writing Figure axis labels to avoid confusing the reader. As an example, write the quantity “Magnetization”, or “Magnetization, M”, not just “M”. If including units in the label, present them within parentheses. Do not label axes only with units. In the example, write “Magnetization (A/m)” or “Magnetization {A[m(1)]}”, not just “A/m”. Do not label axes with a ratio of quantities and units. For example, write “Temperature (K)”, not “Temperature/K”.

##### Acknowledgment *(Heading 5)*

The preferred spelling of the word “acknowledgment” in America is without an “e” after the “g”. Avoid the stilted expression “one of us (R. B. G.) thanks ...”. Instead, try “R. B. G. thanks...”. Put sponsor acknowledgments in the unnumbered footnote on the first page.

##### References

The template will number citations consecutively within brackets [1]. The sentence punctuation follows the bracket [2]. Refer simply to the reference number, as in [3]—do not use “Ref. [3]” or “reference [3]” except at the beginning of a sentence: “Reference [3] was the first ...”

Number footnotes separately in superscripts. Place the actual footnote at the bottom of the column in which it was cited. Do not put footnotes in the reference list. Use letters for table footnotes.

Unless there are six authors or more give all authors’ names; do not use “et al.”. Papers that have not been published, even if they have been submitted for publication, should be cited as “unpublished” [4]. Papers that have been accepted for publication should be cited as “in press” [5]. Capitalize only the first word in a paper title, except for proper nouns and element symbols.

For papers published in translation journals, please give the English citation first, followed by the original foreign-language citation [6].

1. G. Eason, B. Noble, and I. N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955. *(references)*

We suggest that you use a text box to insert a graphic (which is ideally a 300 dpi TIFF or EPS file, with all fonts embedded) because, in an MSW document, this method is somewhat more stable than directly inserting a picture.

To have non-visible rules on your frame, use the MSWord “Format” pull-down menu, select Text Box > Colors and Lines to choose No Fill and No Line.

1. J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
2. I. S. Jacobs and C. P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.
3. K. Elissa, “Title of paper if known,” unpublished.
4. R. Nicole, “Title of paper with only first word capitalized,” J. Name Stand. Abbrev., in press.
5. Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].
6. M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.
7. Horridge, M., Knublauch, H., Rector, A., Stevens, R., & Wroe, C. (2004). A practical guide to building OWL ontologies using the Protégé-OWL plugin and CO-ODE Tools Edition 1.0. *University of Manchester*.

http://www.lrmi.net/worlds-leading-search-engines-recognize-lrmi-as-education-metadata-standard