Empowering Moodle with Rules and Semantics

Sergey Lukichev, Mircea Diaconescu, Adrian Giurca

Brandenburg University of Technology at Cottbus
{Lukichev, M.Diaconescu, Giurca}@tu-cottbus.de

Abstract. This short paper describes preliminary ideas for empowering e-learning platform Moodle with rules and semantics. Many existing web applications already contain a lot of structured information, which is still not presented in machine-readable way. Extracting this information from an existing e-learning platform may give benefits to course tutors such as more control over course management, advanced reports and filters, reasoning over the course content. We describe how to represent the existing Moodle content in RDF and how to add rules on top of the RDF fact base. A semi-automatic method for rule mining and rule development is discussed.

Keywords: *Moodle, RDF, Semantic Web, Rule Mining, RDF-izing Moodle, e-learning, reasoning*

1 Introduction

In spite of a large amount of theoretical research conducted in the area of the Semantic Web, existing Semantic Web technologies are still hard to use practically and it is difficult for end users to take a benefit of them. The RDFa standard [3] is called upon to help web developers to enrich their content with semantics. There are debates and opinions that these standards are still difficult to employ and an alternative approach, using Microformats[1], claims about simplicity in introducing semantics to the existing web pages.

In order to get rid of the burden of the manual content annotation, a lot of work has recently been focused on extracting semantics from the existing content. Wikis, blogs, various content management systems and e-learning platforms contain a lot of unstructured data and this data as it is now cannot release the full power of its semantics. It cannot be shared using common vocabulary, there are no reasoning and semantic querying capabilities. An issue of extracting semantics from existing content becomes important and a number of works has been introduced for the semantic blogging [5] and for extracting semantics from the Wikipedia templates [4]. The Structured Blogging initiative[2] has developed plug-ins for the semantic annotation of Wordpress, MovableType and Drupal contents using Microformats.

[1] Microformats: http://microformats.org/

[2] Structured Blogging: http://structuredblogging.org

The use of Semantic Web technologies in e-learning may bring benefits to tutors and students. For instance, a tutor may want to know *which courses take those students, who usually take the Web Technology course?*, or *what is an average grade of students, who did not use a recommended course tutorial?* A traditional solution to answering such questions is to use relational databases (SQL queries, views). But moving to RDF is a shift to the open world with many distributed resources, identified by URIs as a mechanism for referring to global constants on which there exists some agreement among multiple data providers. Queries can be performed not only over a single database, but over the content of several distributed educational systems, including resources, which are externally available on the web.

As a working platform we use Moodle, an Open Source e-learning application, written in PHP and widely used by teachers all over the world. As a reasoning platform we choose Jena 2[3], a Semantic Web Framework, which allows reasoning on top of RDF fact bases with rules. Courses in Moodle already contain enough information to answer sample queries (see above) and many others, but in order to get answers, semantics should be extracted first and then rules on top of the structured data have to be defined.

We describe how to obtain an RDF fact base from the Moodle content and to empower it with rules. Rules add flexibility to the content analysis and give more control to tutors over the courses and teaching process. The distinct point of our approach among others is that it employs rules, which are obtained semi-automatically in two steps: *i)* using data mining methods over RDF for deriving initial rule base and *ii)* extending the initial rule base with more complex rules by a rule modeler (for instance, a course lecturer). The approach consists of the following steps:

1. Deriving the Moodle information model and generating an RDF fact base out of the Moodle content on the base of the information model (Section 2);
2. Mining rules, using the RDF fact base, obtained in the previous step. The mining process is automatic and its result is a rule base with rules in a form of $P(\overline{x}) \rightarrow Q(\overline{x})$, for example, *A student, who takes the Web Technologies course, usually takes the Logic for Information Systems course.* (See Section 3 for details);
3. Extending the initial rule base, obtained in the previous step with handcrafted rules, which may contain additional conditions and define new concepts. Usually rules, defined on this step use logical atoms in their conditions, which have been obtained by the rule mining process. Rules on this step are created by tutors. For instance, *A student, who takes the Logic for Information Systems course and completes all assignments is a diligent student.* This is a derivation rule since it derives the concept of a *diligent student*, which may later be used in other rules or in querying. The condition part of this rule also contains the logical atom *A student takes a course*, defined in the step 2 by the rule mining process.

[3] Jena Semantic Web Framework: http://jena.sourceforge.net/

2 The Moodle Information Model

In this section we describe an excerpt from the Moodle information model (Figure 1), derived from the Platform Specific Model (PSM) of the Moodle database schema. The excerpt contains a number of classes and relations as a basis for RDF-izing existing Moodle content.

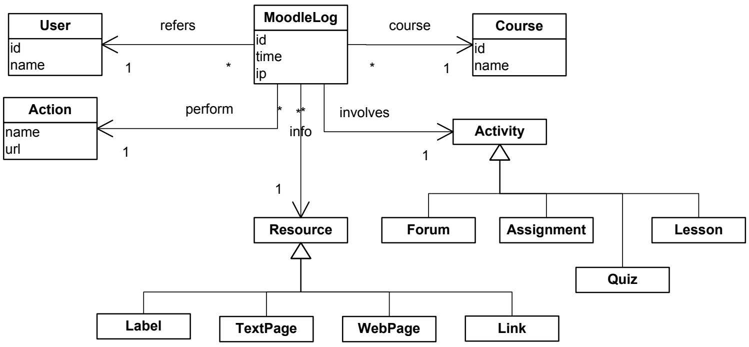

Fig. 1. An excerpt from the Moodle Information Model

The core class is the *MoodleLog* class, which represents a log entry in the Moodle database. All kind of actions, performed by students and tutors are recorded. Each record has one *User*, which may perform exactly one *Action* at a time over a specific *Resource* of a *Course*. A *Resource* is either a *Label*, a *TextPage*, a *WebPage* or a *Link*.

Every course may have a number of activities. Each *Activity* is either a *Forum*, an *Assignment*, a *Quiz* or a *Lesson*.

An RDF representation of this model is derived straightforward. Every UML class on the diagram corresponds to the RDF Schema class and every UML attribute/association corresponds to the RDF Schema property.

The RDF fact base is updated from the Moodle database automatically by a script, which runs periodically. Another option to maintain consistency between Moodle content and its RDF representation is an extension of Moodle, which generates RDF triples at the same time when the Moodle log is updated.

3 Rule Mining

The rule mining algorithms used here are provided by R. Agarwal et al in their seminal papers about mining association rules ([1], [2]). The *MoodleLog* class from the Moodle Information Model is a transactional database. A typical instance of this class (at the PSM[4] level) has the following values for its properties:

[4] Platform-Specific Model in the Model-Driven Architecture approach

```
((id,320834), (time, 1174041794), (userid, 219), (module, forum),
 (action, "view discussion"), (url, "discuss.php?d=413&parent=1350"))
```

These property-value pairs are translated into an RDF set of triples. As in the classical problem of mining association rules the goal is to generate all association rules that have *support* and *confidence* greater than the user-specified minimum support (minsup) and minimum confidence (minconf) respectively. Recall that the formal problem of mining association rules as in [1] is: Let $\mathcal{I} = \{i_1, \ldots, i_n\}$ a set of items. Let $\mathcal{D}$ be a set of transactions (i.e. our MoodleLog extension), where each transaction T is a set of items such that $T \subseteq \mathcal{I}$. Each transaction has a unique identifier (i.e. `id`). We say that a transaction T contains $X \subseteq \mathcal{I}$ if $X \subseteq T$. An association rule is an implication of the form $X \Rightarrow Y$ where $X \subseteq \mathcal{I}$, $Y \subseteq \mathcal{I}$, and $X \cap Y = \emptyset$. The rule $X \Rightarrow Y$ holds in the transaction set $\mathcal{D}$ with confidence c if $c\%$ of transactions in $\mathcal{D}$ that contain X also contain Y. The rule $X \Rightarrow Y$ has support s in the transaction set $\mathcal{D}$ if $s\%$ of transactions in $\mathcal{D}$ contain $X \cup Y$.

Applying the AprioriTid algorithm [2] on our RDF fact base generated from the Moodle logs, we can easily obtain a set of binary association rules involving RDF(S) triples. This is why the use of Jena 2 as a reasoning engine is appropriate. Notice that the algorithm shows best results on large data sets. Our actual Moodle log has more than 500000 entries. An example of a (typed) mined Jena 2 rule is:

```
(?X moodle:perform moodle:view) <-
        (?X rdf:type moodle:User) // type information
        (?X moodle:perform moodle:view_discussion)
        (moodle:view_discussion moodle:url "discuss.php?d=413&parent=1350"^^xs:string)
        (moodle:view moodle:involves moodle:quiz372)
        (moodle:quiz372 rdf:type moodle:Quiz) // type information
```

4 Extending the Rule Base

A rule base, obtained by the rule mining process (Section 3) can be extended with more complex rules, which may define new concepts, for instance, *A student, who has more than 10 forum posts is active in the forum discussion.*

```
(?X, userdef:activeInForum ?Forum) <- (?X rdf:type moodle:Student) (?X userdef:post ?Forum)
                                      (?X userdef:numberOfPosts ?Y) ge(?Y, "10"^^xs:integer)
```

This rule defines a concept of a student, who is active in the forum discussion. Such rules are created manually and a rule modeler should have a possibility to define rules in a user-friendly environment. One possible solution is an editor, which allows composing rules by non-technical users using a point-and-click interface with cut-and-paste pop-up windows and drop-down menus. Such interface allows selecting a predicate from a predefined list and filling term slots with values and variables. An example of such editor is provided in ILOG JRules[5]. Since the target rule platform is Jena 2, rules from the editor are translated into Jena 2 rules. This can be done using I1 rule interchange service, which allows rule interchange between different rule languages and platforms. The core of the

[5] ILOG JRules: http://ilog.com/products/jrules

interchange service is the R2ML loss-free rule interchange format ([7]), developed in the REWERSE Working Group I1[6].

Another possible solution is to use Attempto Controlled English[7] (ACE). ACE allows writing rules, using unambiguous subset of the English language. The work on the ACE to R2ML transformation (in order to obtain Jena 2 rules) has been started ([8], Chapter 1), but we consider the first solution with point-and-click interface as more appropriate for non-technical users.

5 Conclusion

In this paper we have described preliminary ideas for empowering e-learning platform Moodle with rules and semantics. Our further work on this project is towards practical implementation. The primary goal is to give tutors a useful application for flexible course management and reporting by implementing a Moodle module, based on the rule and semantics technologies such as RDF and Jena 2 and on Web 2.0 technologies as Ajax. The possible extension of the project is e-learning platforms integration, using Semantic Web Services, for instance, for rule interchange and querying of several fact bases.

References

1. R. Agrawal, T. Imielinski, A. N. Swami. Mining association rules between sets of items in large databases. In P. Buneman and S. Jajodia, editors, *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, pages 207–216, Washington, D.C., 26–28 1993.
2. R. Agrawal, R. Srikant, Fast algorithms for mining association rules. In J. B. Bocca, M. Jarke, and C. Zaniolo, editors, *Proc. 20th Int. Conf. Very Large Data Bases, VLDB*, pages 487–499. Morgan Kaufmann, 1994.
3. B. Adida, M. Birbeck, RDFa Primer 1.0, 12 March 2007, http://www.w3.org/TR/xhtml-rdfa-primer/
4. Auer, S.; Lehmann, J.: What have Innsbruck and Leipzig in common? Extracting Semantics from Wiki Content. June 2007, Innsbruck, Austria
5. K. Mller, U. Bojars and J. G. Breslin. Using Semantics to Enhance the Blogging Experience. In 3rd European Semantic Web Conference (ESWC2006), Budva, Montenegro, June 2006.
6. G. Klyne and J.J. Caroll (Eds.), Resource Description Framework (RDF): Concepts and Abstract Syntax, W3C, 2004
7. G. Wagner, A. Giurca and S. Lukichev, Language Improvements and Extensions, REWERSE I1 Deliverable I1-D8, published as a technical report, March 2006, http://rewerse.net/deliverables-restricted/i1-d8.pdf
8. G. Wagner, S. Lukichev, N. E. Fuchs, Tool Improvements and Extensions 2, REWERSE I1 Deliverable I1-D11, published as a technical report, March 2007, http://rewerse.net/deliverables-restricted/i1-d11.pdf

[6] REWERSE I1 (Interchange services, downloads, etc): http://rewerse.net/I1

[7] Attempto Project Home Page: http://attempto.ifi.unizh.ch