A Semantic Approach to Multi-parameter Personalisation of E-Learning Systems

Humam K. Majeed Al-Chalabi[1](✉) and Ufuoma Chima Apoki[2]

[1] Faculty of Automatics, Computer Science and Electronics, University of Craiova, Craiova, Romania
hemoomajeed@gmail.com

[2] Faculty of Computer Science, Alexandru Ioan Cuza University, Iasi, Romania
apoki.ufuoma@info.uaic.ro

Abstract. This paper presents a model framework for achieving semantic adaptation in e-learning systems using multiple parameters for personalisation. The proposed model, which utilises semantic technologies, aims to boost learning experiences and outcomes within the process of learning. This is often achieved through a mechanism that adapts the educational contents of a course in keeping with student's preferences expressed by multiple parameters (such as their learning styles, media preferences, level of data, language, etc.) The variation process involves real-time mapping of learning resources and student data, semantic annotation, metadata enrichment of learning resources, creation of student profile with relevant preferences, and personalisation of every course in step with the foremost suitable (or preferred) parameters. Achieving this entailed the creation of an ontology and several other modules that work in the background of the adaptive process.

Keywords: E-learning systems · Adaptive learning · Semantic web · Ontologies · Adaptive educational systems

1 Introduction

The purpose of personalised learning as one of the major needs of this century is the ability to recognise students' needs and preferences, and also their capabilities [1]. An online learning environment brings together learners possessing different learning capacities based on backgrounds and needs, and therefore requires different learning paths to achieve optimal satisfaction for each learner [2]. Individual characteristics and preferences of each learner (such as educational background, learning styles, learning objectives, motivation) are useful (if properly utilised) in the provision of optimal paths in the quest to accomplish individual learning outcomes.

Several platforms like Learning Management Systems (LMS) and Intelligent Tutoring Systems (ITS) exist for the delivery of e-learning content to students, as well as the administration and monitoring of student activities.

D. Simian and L. F. Stoica (Eds.): MDIS 2020, CCIS 1341, pp. 381–393, 2021.
https://doi.org/10.1007/978-3-030-68527-0_24

They offer the possibility of presenting enrolled students with a wide range of courses with highly customisable features. Learning could take different forms which include Computer Managed Learning, Computer Assisted Instruction, Synchronous/Asynchronous Learning, Fixed/Adaptive Learning, Linear/Interactive Learning, and Individual/Collaborative Learning [3].

Personalisation and adaptivity (interchangeably used, most times) have become key necessities in e-learning systems. While personalisation focuses on the customisation of learning content by an instructor, adaptivity refers to software/technology that can alter learning paths or course content in real-time from information that is gained from monitoring students and their interactions with the learning system. However, the majority of e-learning platforms that exist do not offer many options for personalisation or adaptivity; they are mostly achieved by customised learning platforms or by extending LMS (such as Moodle) through plugins or web services.

Personalisation and adaptivity can be achieved by creating different learning paths and/or experiences utilising different features of users. While most systems focus on adaptivity in general, others focus on adaptivity based on few parameters which make such systems course-specific and not easily customisable for other courses [4]. Adaptivity and personalisation can be achieved through adaptive content (which is widely used in implementation), adaptive instruction, and adaptive presentation [5].

There are multiple parameters in the literature used in the personalisation of learning scenarios. Criteria that can be used for personalisation include the learner's preferences, the status and history of the learner, the parameters of the learning medium, and other pedagogical and domain parameters. Level of knowledge and learning styles are popularly utilised in modeling and implementation. Personalisation parameters are covered in [5–7].

To achieve personalisation with multiple parameters, the authors in [7] outlined four main strategies:

- Applying all parameters to personalise each course,
- Applying a subset of parameters which represents only the preferences of the learners,
- Applying a subset based on standardisation of course materials,
- Applying a subset of parameters suggested by a domain expert supervising the course.

The drawback of the first two strategies is the high number of possible learning paths when the set of possible parameters are greater than two or three with different dimensions. This will involve a lot of tests and questionnaires and workload to ensure all dimensions of each personalisation parameters are represented for each concept to be studied. The third strategy takes advantage of metadata standards which already exist, while the last option utilises the expertise of the course instructor. To properly combine multiple parameters, it is imperative to explore the last two options.

Learning objects, which have been described as "*any entity, digital or non-digital, which can be used, re-used, or referenced during technology-supported* *learning*" [8] by the IEEE Learning Technology Standards Committee, have become fundamental in the development of educational resources in e-learning platforms. With constant developments in Information and Communications Technology (ICT) in the educational sector, the number and complexity of learning objects are on the rise. However, there is a lack of interoperability and compatibility of educational repositories, making it cumbersome in the design and maintenance of semantic education libraries and repositories, and the intelligent search of learning objects.

The semantic web, which is an extension of the current web, plays a huge part in the development of personalised learning in e-learning systems. The technologies of the semantic web, which include RDF, XML, and ontologies, can be useful for the intelligent discovery, annotation, semantic enrichment, and transformation of learning objects. Ontologies, which can be described as a "*specification of a conceptualisation*" [9], provide the advantage of solving the challenges of interoperability between the educational repositories of different e-learning systems. With well-defined ontologies, which can be extendable, personalised search and recommendation can be achieved, because computers are meaningfully able to process data due to the commonality of semantic meaning and relationships between terms [10].

As e-learning systems become more prevalent and hard to ignore in learning, there is a growing need for shared learning resources between already-existing learning systems for reusability and adaptability. A feasible approach requires a domain-independent, automatic, and unsupervised method to detect relevant features from heterogeneous learning resources, and associate them to concepts to be learnt which are modelled in a background ontology [11]. These learning resources need to be transformed through annotation into learning objects, which conform to metadata standards. These learning objects can then be set up hierarchically in an ontology. One challenge, though, with ontologies is providing semantic and structural uniformity. One way to solve this is the ontology mapping for the interoperability of learning resources presented in [12].

Within this framework, the rest of this paper is structured as follows: Sect. 2 gives a brief description of adaptivity in e-learning, and the opportunities of semantic web technologies in influencing adaptivity in online learning. Section 3 describes the architecture of the proposed model, and also details the technologies used in the design. The paper ends with Sect.4, which describes future work and improvements on the model and approach.

2 Literature Review

A major differentiating factor between online learning and traditional learning is the ability to redirect the focus of learning to a learner-centered environment. This involves the application of learning analytics tools, which include data collection from learners (log data, student characteristics, educational background, and academic performance) to manage and improve learning. Utilisation of these techniques produce adaptive and intelligent web-based educational systems.

Adaptivity in e-learning can involve various forms, which include adapting learning resources, support, display, and other instructional elements. In the literature, adapting learning resources is the most applied dimension in e-learning platforms, and they can be broadly grouped into content adaptation and link-level adaptation [10]. Content adaptation entails dynamically altering the contents of the learning resources such as fragments, segments, or pages, and having different forms of presentation of content. The other involves presenting the most suitable learning content in the right order based on the learner's needs and preferences. In both cases, it has become necessary to organise educational content as learning objects. The quality and efficiency of e-learning systems depend largely on the quality of suitably-selected learning objects, the relationships between the concepts they instruct, and the parameters for adaptation.

Learning resources used in e-learning platforms are mostly digital, allowing possibilities for modification to suit learners' preferences and needs. Learning content is abundant on (and off) the web in a variety of formats. The heterogeneity and amount of accessible learning resources are gradually becoming a challenge for learners and educational instructors/designers who design systems for e-learning purposes. Several metadata standards have been developed in a bid to solve the problems of non-uniformity [6]. However, many available learning resources do not fit these structures. To achieve intelligent and automatic searching and indexing of these diverse learning content, it is necessary to define a simpler metadata standard, which is independent of implementation [13]. Another method of achieving this is the research into the automatic annotation of diverse learning resources.

One major change which promises a huge evolution of learning processes is the transition from the traditional web to the semantic web. The semantic web (Web 3.0) involves the restructuring of data from relational databases to semantic graphs. The main technologies of Web 3.0 include eXtensible Markup Language (XML), Resource Description Framework (RDF), and Ontologies. XML allows for the arbitrary structuring of documents in a format, readable by both humans and machines, without explicitly stating what the structures mean. RDF, which was originally meant to be a metadata modelling language, is used to express data models referred to as objects or "web resources" and the relationships between them. With ontologies, concepts and relationships in a particular domain can be described. This facilitates the meaningful processing of data if there is a common understanding of the concepts and relationships between them [10].

The goal of this study is to create an ontology model and enrich it the components of a Relational Database (RDB) schema using classes provided by a semantic framework. With the semantic framework, it will be possible to make semantic web recommendations.

In this context, we will be using the D2RQ Framework [14] which enables the extraction and restructuring of data from RDBs in RDF graph format using the OWLready2 [15], which is a package for ontology-oriented programming in Python programming language. This makes the data also available for the semantic web. With the D2R server connection to the RDB, SPARQL queries [16] can be run to get semantic data.

Within ontologies and the semantic web, reasoners function to derive information from a knowledge base in an inference engine. The W3C (World wide web consortium) has a list of reasoners which include FaCT++, HermiT, and Pellet [17]. These reasoners can be used when they are interfaced to an ontology through an API. Pellet (developed in Java) is the first sound and complete OWL-DL reasoner with extensive support for reasoning with individuals, user-defined datatypes, and debugging support for ontologies. HermiT checks the consistency of an ontology, and can be used to identify subsumption relationships between ontology classes.

Semantic Web Rule Language (SWRL) integrates rules, concepts, and the relationships between concepts defined in Web Ontology Language (OWL), thereby extending their expressiveness. The possibility of generating new knowledge is achieved by creating rule sets in SWRL and it fundamentally serves as the inference engine [18].

3 Semantic Model for Recommending Learning Resources

Our research is different from related works in several regards. We provide personalisation to the students with a possibility of different parameters. This allows for flexibility and prevents the personalisation algorithm from being course-specific. Personalisation can be achieved by using parameters a course instructor considers necessary or according to the learning materials available for a course. Also, adaptive learning is achieved through different methodologies such as ontologies and inference rules. The proposed model is intended to maximise the learner's ability to learn and also provide individualised learning paths based on the learners' preferences. This is realised by obtaining the learner's initial abilities and preferences and using semantic reasoning and rule-based reasoning to predict the optimum learning path.

Figure 1 describes the architecture and main model elements of the proposed model. It includes the following: the learner model (which holds preferences and capabilities), learning objects metadata for annotation and semantic enrichment, ontologies describing learning objects and learners, and the personalisation parameters that will be used for adaptation. The learners and the course instructor access the LMS through the user interface. The learner model stores information about the learner in the Learner Data Repository. Information stored include learner characteristics and educational background. Learning resources are semantically enriched with information related to personalisation parameters, and subsequently stored in the Learning Object Data Repository. The information from the learner model and domain model are subsequently utilised in the mechanism for adaptation.

The path to personalised learning paths for the model will be described. The course instructor adds learning resources for each course according to different

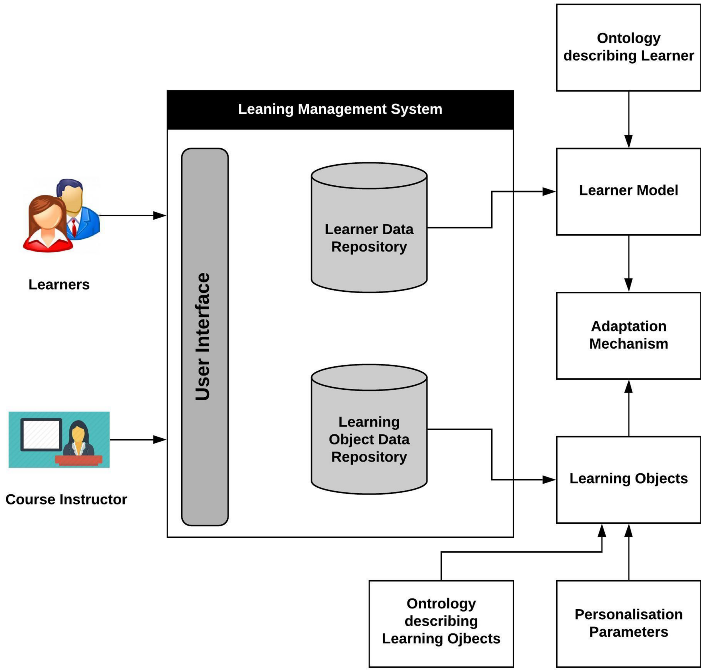

Ontology describing Learner
Leaning Management System
Learners
User Interface
Learner Data Repository
Learner Model
Adaptation Mechanism
Course Instructor
Learning Object Data Repository
Learning Objects
Ontrology describing Learning Ojbects
Personalisation Parameters

Fig. 1. The architecture of the model

dimensions of personalisation parameters with the user interface. The learning resources are subsequently transformed into learning objects by adding descriptions specific to the LOM (Learning Object Metadata) standard [19]. The course instructor then chooses the parameters that will be used in a certain course (for example, language, level of knowledge, and Honey-Mumford Learning Style). An alternative will be to select the most important parameters based on available learning objects in a course using an algorithm and a concept-parameter matrix for the learning objects in the course. This is achieved through the relationships between data elements and personalisation parameters in the domain ontology as shown in Fig. 2.

Table 1 shows the relationships between dimensions of personalisation parameters in the ontology and data elements of the LOM standard. The first column describes possible parameters that can be used to provide personalisation for students. These are usually specified by an instructor or an educational expert. The column, linguistic terms, details different dimensions of the parameters, which learners can be grouped into. Linguistic terms are better suited for characterisation because they provide more concrete demarcations than numerical scales [7]. The next 3 columns represent elements of the LOM standard which the personalisation parameters are mapped to, with the element name and value space mapped to parameters and linguistic terms, respectively.

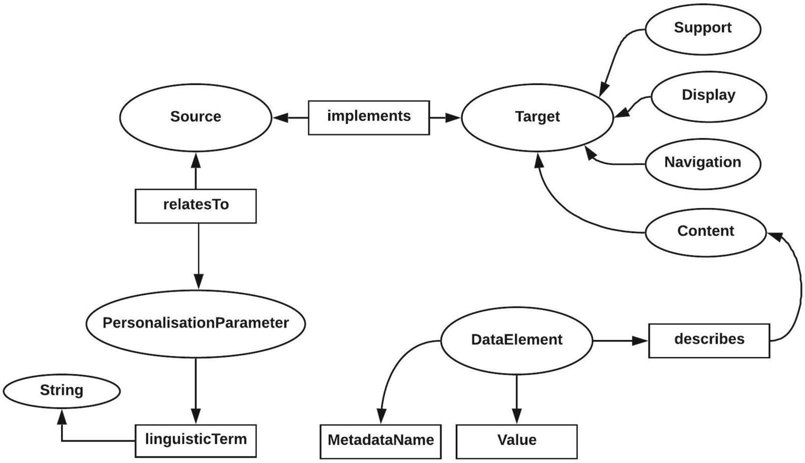


Fig. 2. Ontology mapping between data elements and personalisation parameters

The learner also accesses the LMS through the user interface, which serves as the communication component of the interactions between the learner and the learning system. The LMS was built with Laravel [20], which is an open-source PHP-web framework for the development of web applications. When the user signs up for a course, (s)he is required to take questionnaires or tests for the most important parameters in that course. If the user has previously used the system, the system will have a history of his previous (static) preferences, and (s)he won't have to go through those tests again.

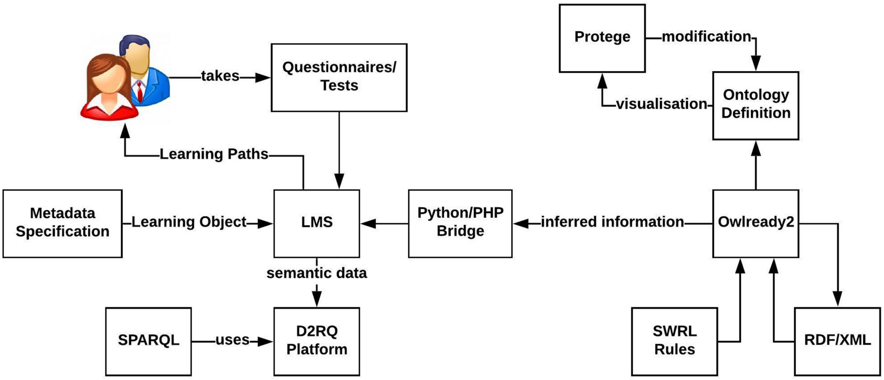


Fig. 3. Main elements of the model architecture

Table 1. Ontology mapping between personalisation parameters and LOM standard

Parameter	Linguistic terms	Nr.	Element name	Value space
Language	English	1.3	Language	en
	English	5.11	Language	en
Media Preference	Audio, text, video	4.1	Format	Audio, image, text, video
		5.2	Learning Resource type	diagram, figure, graph, slide, narrative text
Navigation Preference	Breadth-first, depth-first	1.7	Structure	Hierarchical
Level of Knowledge	Beginner, intermediate, advanced	5.8	Difficulty	Easy, medium, difficult
Bloom's Taxonomy of Learning Goals	Knowledge, comprehension, application	5.2	Learning Resource Type	Exercise, narrative text, exam, experiment, self-assessment, lecture
		9.1	Purpose	Concepts, theories, ideas, examples, exercises, tests
Felder-Silverman Learning Style	Active, reflective	5.1	Interactivity type	Active, expositive, mixed
		5.3	Interactivity level	Low, medium, high
	Sensory, intuitive	9.1	Purpose	Facts, details, principles, theories
	Sequential, global	1.7	Structure	Collection, networked, hierarchical, linear
	Visual, verbal	4.1	Format	Audio, image, text, video
		5.2	Learning Resource Type	diagram, figure, graph, index, narrative text, simulation, slide
Witkin Cognitive Style	Field-dependent, Field-independent	9.1	Purpose	Global approach, analytical approach

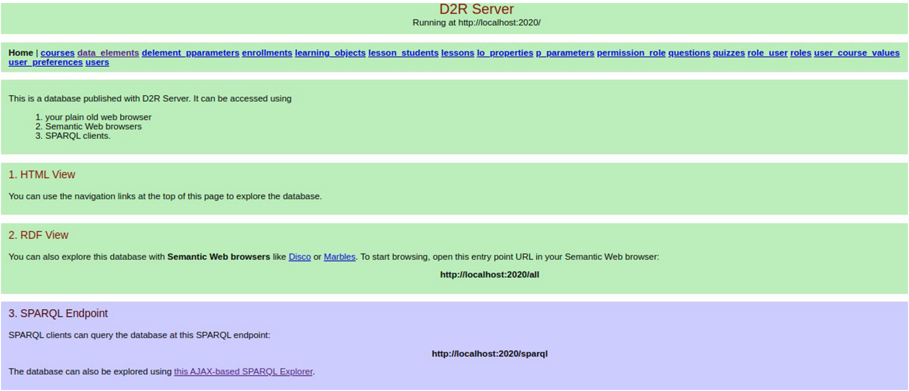
Figure 3 shows the main model elements of the architecture and the procedure for recommendation. The D2R server of the D2QR platform transforms the data stored in the LMS relational database to a semantic database, which is organised in tables. The next phase involves creating the ontology using the OWLready2 platform. The ontology is populated with classes, individuals, and properties using a SPARQL client and ontology-oriented programming (Python and OWLready2). SWRL rules defined in the ontology receive recommendations that determine how learning objects are related to personalisation parameters. With the inference engine, the recommended content is inferred for the learners. The results of these recommendations are provided to the LMS through a Python-PHP bridge. The LMS subsequently presents the personalised learning path to the students.

3.1 Semantic Mapping to the LMS

Mapping the relational database of the LMS to a semantic form can be achieved using the D2RQ platform, which is a system that allows for accessing relational databases as virtual, read-only RDF graphs, without having to recreate the RDB into an RDF store. The main components of interest of the D2RQ platform include a declarative mapping language, which describes relations between an ontology and a relational data model. Another is the D2R server, which uses an HTTP connection to provide a linked data view and HTML view for debugging, and a SPARQL Protocol endpoint over the database which can be accessed with a SPARQL client. These components provide the following functionalities:

- Querying the LMS database (which is non-RDF) using SPARQL,
- Accessing the contents of the LMS database as Linked Data over the Web,
- Creating custom dumps of the LMS database in RDF formats to load into an RDF store,
- Accessing information from the LMS database with a SPARQL client.

Figure 4 shows the D2RQ Engine for the LMS classes, which also specifies a SPARQL endpoint for the LMS dataset mapping. This provides possibilities for executing SPARQL queries to map classes and properties. The queries produce results which can be used in OWLready2 in JSON or XML syntax.



D2R Server
Running at http://localhost:2020/

Home | courses data_elements delement_pparameters enrollments learning_objects lesson_students lessons lo_properties p_parameters permission_role questions quizzes role_user roles user_course_values user_preferences users

This is a database published with D2R Server. It can be accessed using

1. your plain old web browser
2. Semantic Web browsers
3. SPARQL clients.

1. HTML View

You can use the navigation links at the top of this page to explore the database.

2. RDF View

You can also explore this database with **Semantic Web browsers** like Disco or Marbles. To start browsing, open this entry point URL in your Semantic Web browser:

http://localhost:2020/all

3. SPARQL Endpoint

SPARQL clients can query the database at this SPARQL endpoint:

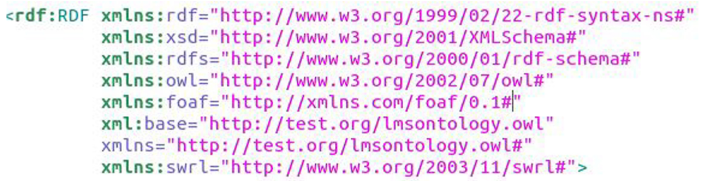
http://localhost:2020/sparql

The database can also be explored using this AJAX-based SPARQL Explorer.

Fig. 4. Classes of the LMS mapped on the D2R Server

3.2 Creating a Domain Ontology in OWLready2

In designing ontologies, the first ideal step would be identifying the goal and scope of the ontology. When ontologies are designed properly, they can be used to accurately describe a domain. It is, however, imperative to balance the expressiveness and complexity of the design. For this research, we took advantage of existing standard vocabularies and ontologies, and applied their classes and properties when creating the mapping file that was used on the D2RQ platform. Learning Objects were modelled according to IEEE LOM ontology standards and the learners were modelled according to the basic FOAF (Friend of a Friend) [21] ontology standard as shown in Fig. 5.



```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
         xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
         xmlns:owl="http://www.w3.org/2002/07/owl#"
         xmlns:foaf="http://xmlns.com/foaf/0.1#"
         xml:base="http://test.org/lmsontology.owl"
         xmlns="http://test.org/lmsontology.owl#"
         xmlns:swrl="http://www.w3.org/2003/11/swrl#">
```

Fig. 5. Using existing standards in the ontology design

In the literature, there are three main strategies for accessing ontologies in a programming language [15]. The first strategy involves using a query language such as SPARQL. The second strategy involves the use of an Application Programming Interface such as OWL API and Jena. The third strategy, which was used, involves ontology-oriented programming (OWLready2 and Python, in this case). This utilises the advantage of the similarities between object models and ontologies, with classes, properties, and individuals in an ontology corresponding to classes, attributes, and instances, respectively, in object models. It also allows for the definition of classes and hierarchies, variables and restrictions, the relationships between classes. Figure 6 shows a visual representation of the resulting ontology using Protege.

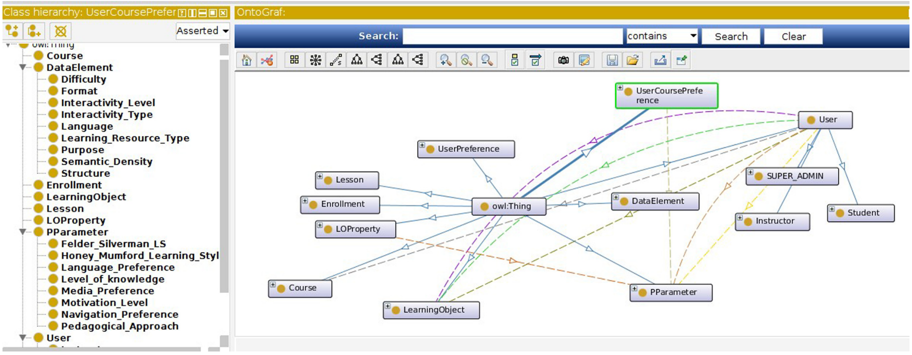
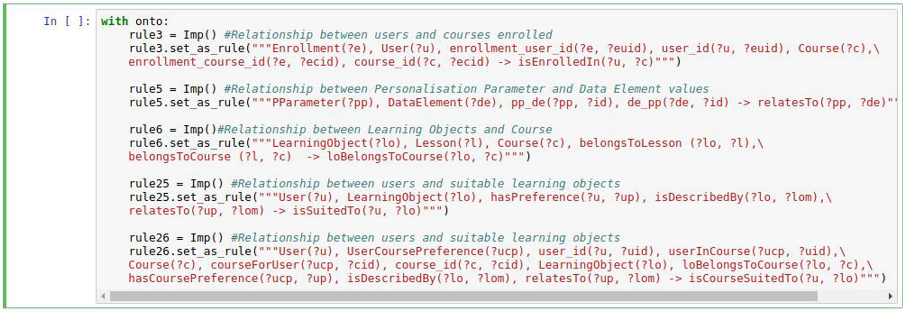


Fig. 6. Visualising the ontology with Protege

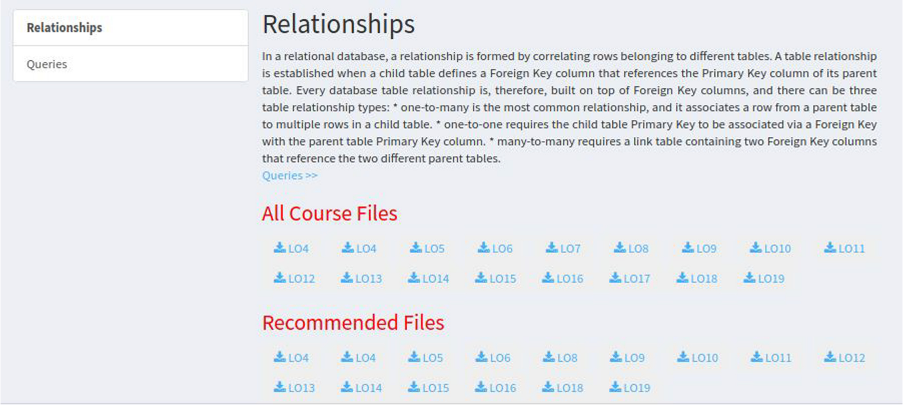
3.3 SWRL Rules

SWRL rules are basically used to integrate 'if ..., then ... ' associations in ontologies. SWRL rules were used to define the relationships between data elements of the LOM standard and different dimensions of a personalisation parameter. These rules are then used to identify the suitability of a learning object (based on its data properties) to a learner (based on the results of tests and questionnaires). Some examples of SWRL rules are described in Fig. 7. In Fig. 8, a list of



```
In [ ]: with onto:
            rule3 = Imp() #Relationship between users and courses enrolled
            rule3.set_as_rule("""Enrollment(?e), User(?u), enrollment_user_id(?e, ?euid), user_id(?u, ?euid), Course(?c),\
            enrollment_course_id(?e, ?ecid), course_id(?c, ?ecid) -> isEnrolledIn(?u, ?c)""")

            rule5 = Imp() #Relationship between Personalisation Parameter and Data Element values
            rule5.set_as_rule("""PParameter(?pp), DataElement(?de), pp_de(?pp, ?id), de_pp(?de, ?id) -> relatesTo(?pp, ?de)""

            rule6 = Imp()#Relationship between Learning Objects and Course
            rule6.set_as_rule("""LearningObject(?lo), Lesson(?l), Course(?c), belongsToLesson (?lo, ?l),\
            belongsToCourse (?l, ?c)  -> loBelongsToCourse(?lo, ?c)""")

            rule25 = Imp() #Relationship between users and suitable learning objects
            rule25.set_as_rule("""User(?u), LearningObject(?lo), hasPreference(?u, ?up), isDescribedBy(?lo, ?lom),\
            relatesTo(?up, ?lom) -> isSuitedTo(?u, ?lo)""")

            rule26 = Imp() #Relationship between users and suitable learning objects
            rule26.set_as_rule("""User(?u), UserCoursePreference(?ucp), user_id(?u, ?uid), userInCourse(?ucp, ?uid),\
            Course(?c), courseForUser(?ucp, ?cid), course_id(?c, ?cid), LearningObject(?lo), loBelongsToCourse(?lo, ?c),\
            hasCoursePreference(?ucp, ?up), isDescribedBy(?lo, ?lom), relatesTo(?up, ?lom) -> isCourseSuitedTo(?u, ?lo)""")
```

Fig. 7. SWRL rules



Relationships

In a relational database, a relationship is formed by correlating rows belonging to different tables. A table relationship is established when a child table defines a Foreign Key column that references the Primary Key column of its parent table. Every database table relationship is, therefore, built on top of Foreign Key columns, and there can be three table relationship types: * one-to-many is the most common relationship, and it associates a row from a parent table to multiple rows in a child table. * one-to-one requires the child table Primary Key to be associated via a Foreign Key with the parent table Primary Key column. * many-to-many requires a link table containing two Foreign Key columns that reference the two different parent tables.

Queries >>

All Course Files

LO4 LO4 LO5 LO6 LO7 LO8 LO9 LO10 LO11
LO12 LO13 LO14 LO15 LO16 LO17 LO18 LO19

Recommended Files

LO4 LO4 LO5 LO6 LO8 LO9 LO10 LO11 LO12
LO13 LO14 LO15 LO16 LO18 LO19

Fig. 8. Recommending learning objects

learning objects recommended for a user based on her preferences for the lesson on “Database Relationships” is shown.

The SWRL rules are executed sequentially in Pellet (which is embedded in OWLready2). The values that belong to the class of individuals and inferred values can be stored in the ontology.

4 Conclusion and Future Work

It has become a fact that incorporating e-learning into traditional learning curriculums is inevitable, and most educational institutions are using learning management systems to augment classroom activities. We have, in this research paper, proposed a semantic approach for adapting learning resources while incorporating multiple learning parameters.

This process involved using semantic technologies, ontologies, and adaptation rules. The advantages of this approach is the fact that the methodology is not restricted to specific courses and the course instructor can choose specific parameters based on experience, or parameters can be selected based on the learning resources in the course. For future work, we will focus on using an LMS such as Moodle, which has an expressive and comprehensive RDB. This is useful for incorporating dynamic adaptivity. Also, we will work on defining the algorithm to select parameters based on the course materials.

References

1. National Academy of Engineering: 14 grand challenges for engineering in the 21st century. http://www.engineeringchallenges.org/challenges.aspx. Accessed 20 Sep 2020
2. Brusilovsky, P.: Adaptive hypermedia for education and training (2012). https://doi.org/10.1017/CBO9781139049580.006
3. Tamm, S.: Types of E-learning (2019). https://e-student.org/types-of-e-learning/. Accessed 12 Sep 2020
4. Apoki, U.C., Ennouamani, S., Al-Chalabi, H.K.M., Crisan, G.C.: A model of a weighted agent system for personalised e-learning curriculum. In: Simian, D., Stoica, L.F. (eds.) MDIS 2019. CCIS, vol. 1126, pp. 3–17. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-39237-6_1
5. Vandewaetere, M., Desmet, P., Clarebout, G.: The contribution of learner characteristics in the development of computer-based adaptive learning environments. Comput. Hum. Behav. **27**, 118–130 (2011). https://doi.org/10.1016/j.chb.2010.07.038
6. Apoki, U.C., Al-Chalabi, H.K.M., Crisan, G.C.: From digital learning resources to adaptive learning objects: an overview. In: Simian, D., Stoica, L.F. (eds.) MDIS 2019. CCIS, vol. 1126, pp. 18–32. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-39237-6_2
7. Essalmi, F., Ayed, L.J.B., Jemni, M., Kinshuk, Graf, S.: A fully personalization strategy of E-learning scenarios. Comput. Hum. Behav. **26**, 581–591 (2010). https://doi.org/10.1016/j.chb.2009.12.010
8. Learning Technology Standards Committee (IEEE): The learning object metadata standard - ieee learning technology standards committee. https://www.ieeeltsc.org/working-groups/wg12LOM/lomDescription. Accessed 17 Sep 2020
9. Gruber, T.R.: A translation approach to portable ontology specifications. Knowl. Acquis. **5**, 199–220 (1993). https://doi.org/10.1006/knac.1993.1008
10. Kurilovas, E., Kubilinskiene, S., Dagiene, V.: Web 3.0 - based personalisation of learning objects in virtual learning environments. Comput. Hum. Behav. **30**, 654–662 (2014). https://doi.org/10.1016/j.chb.2013.07.039
11. Vicient, C., Sánchez, D., Moreno, A.: An automatic approach for ontology-based feature extraction from heterogeneous textualresources. Eng. Appl. Artif. Intell. **26**, 1092–1106 (2013). https://doi.org/10.1016/j.engappai.2012.08.002
12. Arch-int, N., Arch-int, S.: Semantic ontology mapping for interoperability of learning resource systems using a rule-based reasoning approach. Expert Syst. Appl. 40, 7428–7443 (2013). https://doi.org/10.1016/j.eswa.2013.07.027
13. Perisic, J., Bogdanovic, Z., Duric, I.: Semantic model for adaptive e-learning systems. In: Symposium proceedings-XV International symposium Symorg 2016: Reshaping the Future Through Sustainable Business Development and Entrepreneurship, Belgrade, Serbia, pp. 345–353 (2016)

14. D2RQ: Accessing Relational Databases as Virtual RDF Graphs. http://d2rq.org/. Accessed 22 Sep 2020
15. Lamy, J.B.: Owlready: ontology-oriented programming in Python with automatic classification and high level constructs for biomedical ontologies. Artif. Intell. Med. **80**, 11–28 (2017). https://doi.org/10.1016/j.artmed.2017.07.002
16. SPARQL Query Language for RDF, W3C. https://www.w3.org/TR/rdf-sparql-query/. Accessed 23 Sep 2020
17. OWL/Implementations. https://www.w3.org/2001/sw/wiki/OWL/Implementations. Accessed 23 Sep 2020
18. Hassanpour, S., O'Connor, M.J., Das, A.K.: Visualizing logical dependencies in SWRL rule bases. Presented at the (2010). https://doi.org/10.1007/978-3-642-16289-3_22
19. Learning object standard - EduTech Wiki. http://edutechwiki.unige.ch/en/Learning_object_standard. Accessed 25 Sep 2020
20. The PHP Framework for Web Artisans. https://laravel.com/. Accessed 25 Sep 2020
21. Brickley, D., Miller L.: FOAF vocabulary specification 0.91. Citeseer (2007)