Building an Adaptive E-Learning System

Conference Paper · April 2017

DOI: 10.5220/0006326103750382

Christos Chrysoulas
London South Bank University

Building an Adaptive E-learning System

Christos Chrysoulas[1] and Maria Fasli[1,2]
[1]School od Computer Science and Electronic Engineering, University of Essex, Essex, U.K.
[2]Institute of Analytics and Data Science, University of Essex, Essex, U.K.
{cchrys, mfasli}@essex.ac.uk

Keywords: E-Learning, Adaptive Learning, User Profile, Learning Path, Machine Learning.

Abstract: Research in adaptive learning is mainly focused on improving learners' learning achievements based mainly on personalization information, such as learning style, cognitive style or learning achievement. In this paper, an innovative adaptive learning approach is proposed based upon two main sources of personalization information that is, learning behaviour and personal learning style. To determine the initial learning styles of the learner, an initial assigned test is employed in our approach. In order to more precisely reflect the learning behaviours of each learner, the interactions and learning results of each learner are thoroughly recorded and in depth analysed, based on advanced machine learning techniques, when adjusting the subject materials. Based on this rather innovative approach, an adaptive learning prototype system has been developed.

1 INTRODUCTION

With the recent rapid advances in computer and network technologies, educational researchers have developed methods, tools and environments for computer-assisted learning [Hwang, 2002]. Several researchers have already addressed the importance of adaptive learning, either in traditional forms of instruction or in computer-assisted instruction. In addition, several personalization techniques have been proposed for developing web-based learning systems [Santally and Alain, 2006].

In studying the effect of adaptive learning in science courses, most researchers often pay attention to the impact of a single type of personalization, such as learning performance (including learner's profile and learning portfolio), learning style, cognitive style of individual students, on the determination of difficulty levels, learning paths or presentation styles of subject materials [Triantafillou et al. 2004]. However, the interactions among multiple sources of personalization information are rarely taken into consideration.

An adaptive e-Learning system gives the learner the opportunity to select learning materials or contents according his/her style, profile, interest, previous knowledge level. A number of works have been conducted in the area of adaptive learning [Kamceva and Mitrevski, 2012].

The study of how learners learn has been a concern for researchers for many years [Pinto, et al. 2008]. In traditional classroom system, an instructor can control this aspect based on what s/he sees of her/his learners' reaction. However, for e-learning to be effective, it should be adapted to one's personal learning style [Villaverde et al. 2006]. Traditional e-learning systems provide the same materials to all learners. E-learning systems should be capable of adapting the content of courses to the individual characteristics of learners. Adaptive e-learning systems attempt to address this challenge by changing the presentation of material to suit each individual learner. They collect information about learner's goals, preferences and knowledge in order to adapt the education needs of that learner.

An e-learning system must be based on learner's learning style which makes e-learning more effective and efficient. However, most e-learning systems do not consider learner characteristics. One of the most desired characteristics of an e-learning system is personalization, as people with different skill sets use the system. This paper presents an adaptive e-learning system based on the learner's learning style and preferences. The system identifies the learner's learning styles tendency through an initial assessment test. The test's score will be used by the system as a basis to provide the learner a presentation of learning materials more closely to his/her knowledge level. This is the first input of the user in the system in parallel with some basic information that they provide in the form of a profile. In this paper, a multi-source adaptive learning system is also proposed. The proposed system can easily construct adaptive subject series of tests and propose sources for the learners to study and tutors to talk to, by taking both student learning behaviours and learning styles as part of the personalization information.

The rest of the paper is structured as follows: Section 2 gives a brief overview of the abstract Adaptive e-learning architecture and an in depth insight on how the User profile is modelled in the proposed system. Section 3 discusses what a learning path is and its importance. Section 4 discusses the adopted machine learning approach in order to put the needed personalization and adaptivity into the system. Section 5 presents some initial results of the analysis while Section 6 concludes the work.

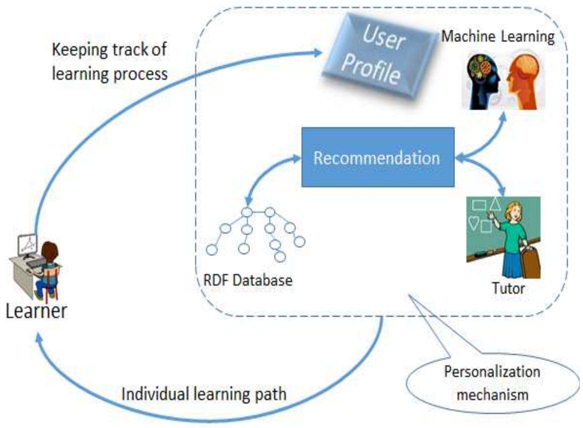

Figure 1: Adaptive E-learning System Architecture.

2 USER PROFILE DESIGN

As depicted in Figure 1, the system is using the intelligence coming from the machine learning part, to dynamically propose learning paths to individual learners, based on their requirements and needs. The learning paths are formed based on previous users, with similar characteristics, interactions with the system. Learning paths that led to successful outcome, meaning the leaner improved his/her skills are targeted. The Machine Learning part is responsible for accessing the dataset that is extracted from the RDF database and for feeding it to the actual machine learning algorithm (see Section 4 for more information). The usage of Resource Description Framework (RDF) database [W3C RDF Working Group, 2014] makes it easy for the system to update the user's profile thus keeping track of the whole learning process. The Recommendation part is also a sub-component of the Machine Learning part. The personalized recommendation(s) provided are the outcome of the machine learning interference with the system. Association analysis based on the Apriori [Agrawal and Srikant, 1994] algorithm is taking place.

We use the terms "user" and "learner" interchangeably to refer to the same entity. An "active" user/learner is the one that we are currently considering or dealing with. In order to be able to guide the user through the learning and assessment process, information about the user and his/her activities will need to be collated and recorded in a user profile. The User Profile will need to record both static and more dynamic information. The User Profile as a cornerstone component of the proposed E-learning system is well studied and documented during the development process. Figure 2 provides a schematic overview of the proposed User Profile.

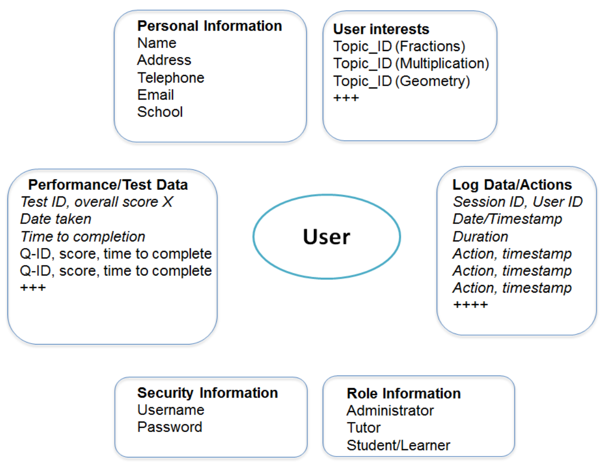

Figure 2: User Profile Modelling.

2.1 Security and Role

Security information is about the user's authority to use the system and it comes in the form of: (i) Username; and (ii) Password. OAuth 2.0 [OAuth 2.0, 2012] protocol is providing the needed security for the users to securely sign-in to the system.

Role information provides an insight in the relationship among the system's users, and can be described as: (i) Administrator; (ii) Tutor; and (iii) Student/Learner. In this way, the system is in position to easily provide different services to different type of users.

2.2 Personal Information

This is predominantly static data recording some basic information on the user: (i) Name; (ii) Address; (iii) Phone; (iv) School; (v) Email; (vi) Sex; and (vii) Telephone. There may be useful pieces of information such as the postcode that could help identify other users living in the locality that the active user may wish to connect to and the school that the user/learner is attending as this may be useful in identifying/putting him/her in a group or further on looking at class or even school level performance metrics and the performance of the active user within a class/school.

2.3 User Interests

User interests in the system essentially represent the topics that the learner is working on (or wishes to work on) and improve his/her performance. There are two ways that user interests can be collected: implicitly and explicitly. Implicitly capturing user interests would entail that the user behaviour (topics chosen to read, or specific tests chosen that cover specific topics) would need to be observed (unobtrusively) and then these mapped against the system's database (RDF database) by using semantic similarity measures [Slimani, 2013]. The explicit way of recording user interests would entail that at the time of registration and then periodically, the user would explicitly indicate his/her interests in topics drawn from the database. In other words, the user needs to be shown parts of the database capturing the topics and choose from these.

User interests are declared in advance by the user and hence captured explicitly. In this way, the user interests then simultaneously indicate the learning objectives and therefore in a way what needs to be achieved by the user. So for instance, an interest in fractions means that the user wants to master the topic of fractions.

2.4 Performance/Test Data

The dynamic data within the user profile are in essence the data generated from the user taking tests. Information like: (i) Test id; (ii) Overall score; (iii) Date taken; (iv) Time to completion; (v) Qx-id, score (or simply correct/incorrect).

We may wish to record the time it takes a user to complete a question as a) this may be different from user to user; b) it can be used to distinguish between difficult and more easy questions (and even use this information later on to adjust the level of difficulty of a question). Also the level of difficulty of the individual questions involved in the test can give an insight on the overall level of difficulty of the test. The data generated from the tests will be used to capture and record the user's progress on a topic. A test can have multiple topics covered through the questions.

2.5 Log Data and Actions within the System

As the user interacts with the e-learning system, s/he is doing so by performing a set of actions. As the user logs in with a unique ID, therefore his/her activity can be tracked. We would presume that these log data are "raw".

Assuming that activity will be recorded in sessions, the raw data would look like: (i) Session ID/UserID; (ii) Date/Timestamp; (iii) Duration; and (iv) Action x, timestamp x. Where Action can be:

1. Test_Taken, TestID;
2. Topic_Browsed, TopicID
3. Topic_searched, TopicID
4. Talked to a Tutor, TutorID

We can make a distinction in the educational platform between self-directed and directed learning [Brookfield, 2009]. The actions we may wish to record vary somewhat between these two, although they have many elements in common. In both areas, the concept of "engagement" is very important. Engagement could be measured by a combination of the following:

(A) Self-directed: (i) How often one logs into the system; (ii) Session duration; (iii) Page view duration; (iv) Abandoned tests; (v) Results review (has the learner always reviewed results?); (vi) Following links; and (vii) Repeating topics, that is, taking another test in the same topic.

(B) Directed: (i) How often the learner has contacted the educator; (ii) The feedback the educator has given; (iii) The additional tests the educator has assigned; (iv) Whether the user has in fact taken these or not; and (v) Links that the educator has recommended.

Such "actions" (or in other words, how is it that the user interacts with the system) would be important as they would tell us what users do and helps us identify learning paths (see Section 3) by aggregating either a specific user's actions or the actions of multiple users. In other words, the log data could be mined to identify actions of individual users and/or groups and distinguish between successful learning paths (or sequences of actions) and not so successful learning paths.

3 LEARNING PATH

The end objective is for learners to master specific topics and the ways they can assess their own progress is by taking tests, talking to tutors, and visiting educational sources (webpages) proposed by the system or/and tutors. There are two aspects of the learning path: the modelling of what a learning path is and then its extraction. The modelling of what a learning path is and its conceptual meaning is important. Then what follows is the actual extraction of the learning path in terms of SPARQL queries [W3C SPARQL Query Language for RDF, 2013].

A learning path is not just a series of tests taken, but it is a series of actions that lead to a successful outcome. A learning path expresses a set of actions as taken by a learner in relation to a specific topic that needs to be mastered (or in order to achieve a threshold of performance in a specific topic). A learning path can be extracted with SPARQL queries from the RDF database. A learning path is all of the actions that are related to a topic or a combination of topics, culminating in the successful achievement of the outcome, measured by a test.

All of these data points and their attributes are used by the machine learning to identify the attributes of successful learning paths. Whether or not the learning path is successful is indicated by the answer to. The following subsection 3.1 presents in depth two use cases of how a learning path is constructed and who the involved parties and roles are.

3.1 Use Cases

Initially a user (Learner 1, from now on), logs to the system. Subsequently an initial Test is proposed to Learner 1 in order to estimate the level of his/her knowledge. Based on that, the system may propose some *Links* for extra studying, some *tutors* to speak to, or even some *extra tests*. After this initial phase, let's assume that Learner 1 chooses Topic 1 and takes a Test T1. He scores 30. From the score achieved (and assuming that 40 or 50 is a threshold of performance that needs to be achieved), it is obvious that Learner 1 requires extra support and help in Topic 1.

What should follow then is an initial recommendation on improving knowledge on Topic 1. The recommendation can be based i) on what actions similar learners followed as actions and appear to have helped them in succeeding in their tests; ii) on what the specific topic that a learner did poorly in is about (for example Topic 1 depends on understanding and mastering two subtopics 1.1 and 1.2 which are essential for answering questions on Topic 1); and iii) a combination of (i) and (ii).

So Learner 1 visits page P1 and page P2 associated with Topic 1, and speaks to a tutor or both. After that, Learner 1 takes Test T2 in Topic 1. Learner 1 achieves a score of 60. Learner 1 managed to improve the score in Topic 1 (which was the target). Clearly something that has happened between taking tests T1 and T2 has led Learner 1 to improve his/her performance. This could be the result of the recommendation made by the system (see (i)-(iii) above) or Learner 1 engaging with other Learners or following suggestions for additional study and tests by the tutor.

Therefore it is evident that it would be helpful to them, if the system is able to provide some useful hints of what previous Learners have done (tests taken, tutors they spoke, etc.) and how they performed when they interacted with the system. The Machine Learning part will provide the system with a pool of possible questions, tutors and links to be followed by a Learner based on the interactions that previous Learners of the system have had. These suggestions will be per Topic.

Two Use-cases/Scenarios have been spotted where the system can propose – interact with the Learner. The first one is just after the Learner has logged-in into the system, in order to take an initial assigned test (assuming that there is a need for that), and the second one after finishing the initial test, and the Learner proceeds to the choice of the Topic to study.

Use-case: Initial Assigned Test
ID:1
Brief description: This use-case describes how the initial assigned test is provided to the Learner, and the options following the completion of it.
Primary actors: Learner
Secondary actors: Platform, Tutors
Preconditions: Have securely logged-in into the System

Main flow:

a) Learner chooses Topic x

b) The System proposes an initial test to check the level of the Learner

c) The System checks the level of the Learner

d) The System provides the learner with information like tutor(s) to talk to, webpages to visit, etc., based on similar behaviour of previous Users/Learners. Can also provide non-system actions like, practice more, spend more time on studying, etc.

e) The Learner may follow or not the suggestions provided by the system

Post conditions:

None

Alternative flows:

It is possible to take the assigned test before choosing the Topic.

Use-case: Learner chooses a Topic to follow

ID:2

Brief description:

This use-case describes the steps a Learner is following from choosing a topic to study till the completion of his learning session/process.

Primary actors:

Learner

Secondary actors:

Platform, Tutors

Preconditions:

The Learner must complete the initial assigned test first (if it is considered a hard condition)

Main flow:

a) Learner chooses Topic x
b) The system provides them with information like tutor(s) to talk to, webpages to visit, etc., based on actions of past Users/Learners that have been identified as having gone through successful learning paths
c) Learner may follow or not the suggestions
d) Learner chooses a Test to take. The suggestions produced out of the previous step (b) would probably be desirable to remain visible on the screen (maybe on the right side)
e) Learner finishes the Test
f) The System based on how the Learned did in the Test, can propose new Tests, Tutors to speak to and websites for the Learner to follow
g) Learner chooses another Topic. New suggestions based on actions of past Users/Learners that have been identified as having gone through successful learning paths previously (and similar behaviour) are suggested to the Learner
h) Learner ends the learning activity

Post conditions:

None

Alternative flows:

None.

An activity diagram presenting and visualizing the aforementioned two Use Cases can be found in Figure 3.

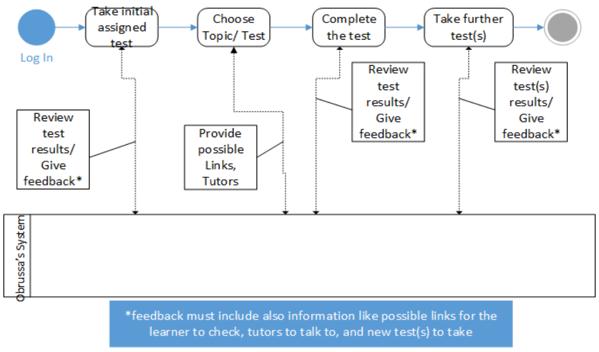

Figure 3: Use-Cases Activity Diagram.

4 MACHINE LEARNING APPROACH

The Query Results Management (QRM) component/module is responsible for managing the data that are extracted from the queries to the RDF database and for assembling the dataset that will be fed to the algorithm. In Figure 4, an illustration of where the QRM manager component is situated in relation to the whole educational system and to the Machine Learning component is presented. The QRM component, a fundamental component is responsible for supporting the following functionalities: (i) Establishing a safe connection to the RDF database; (ii) Querying the RDF database, receiving the data; and (iii) Saving the data in a file and in the proper format for the Machine Learning Management (MLM) component.

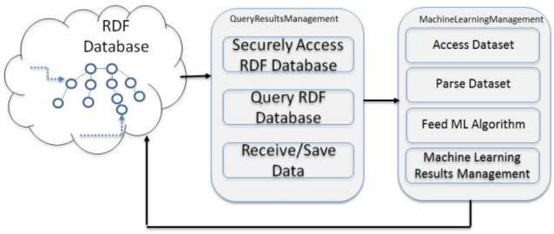

Figure 4: Component based System Architecture.

The Machine Learning Management (MLM) component/module is responsible for accessing the dataset that is formed from the Query Results Management component (QRM) and for feeding it to the actual machine learning algorithm. In Figure 4, an illustration of where the MLM manager component is situated in relation to the whole educational system and to the Query Results Management component is presented. The MLM component will be responsible for supporting the following functionalities: (i) Accessing the dataset formed by the QRM component; (ii) Parsing the dataset, thus filtering out the un-successful learning paths; (iii) Feeding the filtered dataset to the machine learning algorithm; and (iv) Saving the results coming out from the algorithm, to be used for further visualization purposes. This sequence of actions is described in more detail in the following subsections.

4.1 Query Results Management Component

Figure 5 presents the interactions of the QRM module with the RDF Database and the Machine Learning Module.

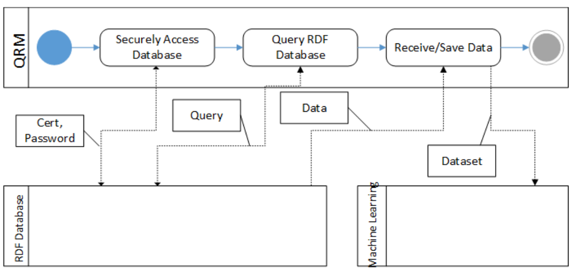

Figure 5: QRM Component Interactions.

QRM will communicate with the RDF Database to query it and get the results. The results could be in XML, JSON, CSV or TSV format. The QRM Manager Component will be responsible to Securely Accessing RDF Database by setting up a two way Secure Sockets Layer (SSL) connection to the Apache Jena Fuseki server [Fuseki, 2016] in order to securely query the RDF database.

4.2 Machine Learning Management Component

The QRM component is responsible for the two first steps of the aforementioned procedure. The MLM component will be responsible for the three last steps of the described procedure. Figure 6 presents the interactions of the MLM component with the QRM Component (and the educational platform/ system).

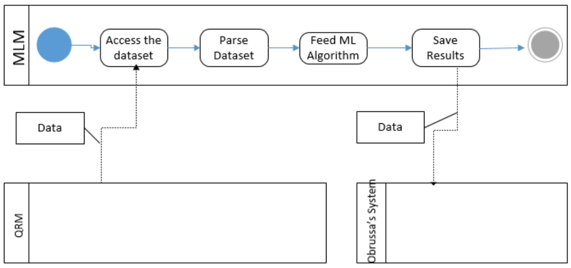

Figure 6: MLM Component Interactions.

The MLM Component will support the following functionality:

Accessing/Parsing the Dataset: After securely connecting to the Apache Jena Fuseki server, and executing SPARQL query(ies) against the endpoint, we are obtaining a dataset containing the interactions the learners had with the system. In the next step, the MLM accesses the dataset and parses it in order to filter out the unsuccessful learning paths. The filtered dataset would only contain the learning paths where the number of correct answered items (questions) are more than the incorrect ones. For the current implementation meaning that only the over/equal to 50 scored learning paths will qualify.

Feeding the ML Algorithm: After filtering out the unsuccessful learning paths, the dataset is ready to be fed to the machine learning algorithm. The machine learning algorithm used in the presented work is the Apriori [Agrawal and Srikant, 1994], since it deals well with datasets containing both numerical and categorical values. The Apriori algorithm is responsible for: (i) finding frequent itemsets, and (ii) mining association rules from the extracted itemsets. See Sections 4.3 and 4.4 for more details.

Saving the results: In this step, the MLM component saves the output of the machine learning algorithm, so as to be used, for recommendation, visualisation and/or other purposes.

The pseudocode of the Apriori algorithm is presented in Algorithm 1.

Algorithm1. The Apriori algorithm.

C_k: Candidate itemset of size k
L_k : frequent itemset of size k

L_1 = {frequent items};
for (k = 1; L_k != ∅; k++) **do begin**
    C_{k+1} = candidates generated from L_k;
    for each transaction t in database do
        increment the count of all candidates in
C_{k+1} that are contained in t
L_{k+1} = candidates in C_{k+1} with min_support
    end
return $\cup_k L_k$;

4.3 Finding Frequent Itemsets

The support of an itemset is defined as the percentage of the dataset that contains this frequent itemset. Frequent itemsets are a collection of items that frequently occur together. In our specific case an itemset is having the following format:

{userId, learningEventUri, leraningEventType, difficultyLevel, contentUri, topicUri, nextlearningEventUri, timestamp}

Support applies to an itemset, so we can define a minimum support and get only the itemsets that meet that minimum support. Support can range from 0 to 1. The confidence is defined for an association rule like {Learner 1} ➞ {Topic 1}. The confidence for this rule is defined as support ({Learner 1, Topic 1})/support ({Learner 1}). The support and confidence are ways someone can quantify the success of our association analysis. Let us assume we want to find all sets of items with a support greater than 0.6. We could generate a list of every combination of items and then count how frequently these occur.

4.4 Mining Association Rules from the Extracted Itemsets

Association rules suggest that a strong relationship exists between two items. From the dataset we have, if we have a frequent itemset, {Learner 1, Question 1, Topic 1}, one example of an association rule is Topic 1 ➞ Question 1. This means if someone chooses Topic 1 Question 1, then there is a statistically significant chance that the Learner will answer Question 1. The converse does not always hold.

In Section 4.3, an itemset is quantified as frequent if it met our minimum support level. There is a similar measurement for association rules. This measurement is called confidence. The confidence for a rule P ➞ H is defined as support (P | H)/ support (P). Similarly to the frequent itemsets generation in Section 4.3, we can generate many association rules for each frequent itemset. It would be desirable if we could reduce the number of rules to keep the problem tractable. We can observe that if a rule does not meet the minimum confidence requirement, then subsets of that rule also will not meet the minimum. We can use this property of association rules to reduce the number of rules we need to test.

5 RESULTS

A series of tests performed in order to check the validity of the proposed approach. Python used for the implementation of the machine learning proposed approach/architecture. The outcome of the association and data analysis that took place has shown that the proposed framework is in position to provide an insight on the behaviour of the students, meaning how they interacted with the system and spot common patterns that lead or not to successful completion of a learning path:

1. **Users and Learning Paths:** The results contain the full information of the users that followed a successful or unsuccessful learning path. The format is: {userId, learning EventUri, leraningEventType, difficultyLevel, contentUri, topicUri, nextlearningEventUri, timestamp}.
2. **Successful Users and type of interaction per topic:** The results contain the full information describing how each successful user answered the questions and interacted with the system (talked to tutor or/and followed links, etc.) per topic. The format is: {userId, leraningEventType, topicUri}.
3. **Percentage for each question per topic:** The results contain the information describing how each question x belonging to topic y was dealt by successful or not users. The format is: {topicUri, contentUri, Percentage}.
4. **Percentage for each topic, and interactions involved:** The results contain the information describing how successful or not users did it per topic, how many tutors were called and how many links were followed. {topicUri, percentage, NumberOf Tutors, NumberOfFollowedLinks}.

6 CONCLUSIONS

The paper presents an adaptive e-learning system based on well defining the explicit characteristics of the individual user/learners. The system identifies the learner's learning styles through an initial assessment test and some initial information a user is providing during sign-in. The test score is used by the system to complement the learner with the needed material for him to successfully complete his learning path.

The paper investigates the detection of E-learners' preferences within learning style dimensions and showing relationship between identifying the personality and learning materials presentation. It contributes how to develop e-learning to different learning styles and combine the advantages of learning management systems with those of adaptive systems. The experiments were performed with 300 learners to show the impact of learning styles on learners' preferences. Several patterns were found where learners with different starting points managed to evolve based on the proposed machine learning based recommendation system.

Future work has to be two-fold: perform experiments with significantly larger datasets, and try to enrich the proposed system with more machine learning methods (e.g. neural nets (NN), support vector machines (SVNs), etc.) thus enhancing the recommendation efficiency of the whole system.

ACKNOWLEDGEMENTS

The work was supported by the «DADO: Data Analytics Driven by Ontologies» project, a collaborative research and development project funded by Innovate UK.

REFERENCES

Agrawal, R., Srikant, R. (1994). Fast algorithms for mining association rules, in: Proc. 20th Int. Conf. Very Large Data Bases, VLDB, 1994, pp. 487– 499.

Brookfield, S. (2009). Self-directed learning. In R. Maclean & D. N. Wilson (Eds.), International handbook of education for the changing world of work (pp. 2615–2627). Dordrecht: Springer. http://dx.doi.org/10.1007/978-1-4020-5281-1.

Fuseki: serving RDF data over HTTP. (2016) https://jena.apache.org/documentation/serving_data/

Hwang, G. J. (2002). On the development of a cooperative tutoring environment on computer networks. IEEE Transactions on System, Man and Cybernetic Part C, 32(3), 272–278.

Kamceva, E., Mitrevski, P. (2012). On the General Paradigms for Implementing Adaptive e- Learning Systems", ICT Innovation Web Proceedings ISSN 1857-7288, pp.281-289.

OAuth 2.0: The OAuth 2.0 Authorization Framework (2012). https://tools.ietf.org/html/rfc6749.

Pinto, J., Ng, P., Williams, S. K. (2008). The Effects of Learning Styles on Course Performance: A Quantile Regression Analysis. Franke College of Business, Working Paper Series – 08-02.

Santally, M. I., Alain, S. (2006). Personalisation in Web-based learning environments. International Journal of Distance Education Technologies, 4(4), 15–35.

Slimani, T. (2013). Description and Evaluation of Semantic Similarity Measures Approa-ches. *International Journal of Computer Applications* 80(10):25-33.

Triantafillou, E., Pomportsis, A., Demetriadis, S., Georgiadou, E. (2004). The value of adaptivity based on cognitive style: an empirical study. British Journal of Educational Technology, 35(1), 95–106.

Tsai, C.-C. (2004). Beyond cognitive and metacognitive tools: The use of the Internet as an ''epistemological'' tool for instruction. British Journal of Educational Technology, 35, 525–536.

Villaverde, J.E., Godoy, D., Amandi, A. (2006). "Learning styles" recognition in e-learning environments with feed-forward neural networks. Journal compilation and Blackwell Publishing Ltd, pp. 197-206.

W3C Resource Description Framework Working Group (2014). https://www.w3.org/ standards/techs /rdf#w3c_all.

W3C SPARQL Query Language for RDF (2013). https://www.w3.org/TR/rdf-sparql-query/