The Seventh International Conference on eLearning (eLearning-2016), 29 – 30 September 2016, Belgrade, Serbia

ALGORITHM FOR PERSONALIZED LEARNING PROCESS

VUK VASIĆ
Belgrade Metropolitan University, Faculty of Information Technologies, vuk.vasic@metropolitan.ac.rs

ALEKSANDRA ARSIĆ
Mathematical Institute of the Serbian Academy of Sciences and Arts, aleksandra@mi.sanu.ac.rs

Abstract: *The optimization of e-learning process plays an important role in the modern studies. Every student is characterized by personal skills, knowledge, opportunities, motivation, cognitive aspects and learning history. For these reasons, each student needs to receive different learning content, which will optimize student's process of learning and give the best possible result in term of received knowledge. In this paper is proposed algorithm as an effective and flexible approach for intelligent personalization of e-learning routes, as sequences of Learning Objects (LOs) that fit students' knowledge. As a proof of concept software is being created. It uses algorithm which classify students in one of three groups based on test's results, which every student must to solve before start of the learning process. After classification is done, students are gathering learning sequence based on student's group. Sequence consists of LOs personalized prepared for every student. The proposed model has been evaluated in a simulated e-learning environment.*

Keywords: *E-Learning, personalized learning routes, learning objects, linear model for classification, k-nearest neighbour's algorithm*

1. INTRODUCTION

The functionality of learning systems has been growing very fast during last decade, with rapid development of technology. There are some popular platforms for learning, like Tutor [4], Moodle [2] or Sakai [3], but there is still a need for platform which is oriented on personalized learning system and content [5]. Personalization of learning concept is much more sophisticated problem, than personalization of system [6]. The optimization of e-learning process plays an important role in the modern studies. Every student is characterized by personal skills, knowledge, opportunities, motivation, cognitive aspects and learning history. For these reasons, each student needs to receive different learning content, which will optimize student's process of learning and give the best possible result in terms of received knowledge. Optimization can be obtained in different ways. For this study, authors proposed algorithm for personalization learning process. Personalized learning is a central design principle for a transformed education system [1]. The focus of personalized learning process is not on the technology, but on the learner's motivation and engagement. The technology is there just to support, not guide the learning process.

Each student is unique and learns in different way. Personalized learning system should be built on the idea that the student can designs his own learning path (indirectly or directly), has flexible learning anytime and anywhere, has quality teachers who motivated and engagement in the learning process. Learning environments should be able to notice how each student learns best and to generete the most appropriete learning path for him.

For generated personalize learning enviroment it is neccersary knowlage of the student's progress tracking, passed exams, taken courses, etc. Gathering of that information can be assured to the development algorithm and tools for e-learning platforms. This paper proposed an algorithm for generating personal learning path.

Paper is organized as follows. Section 2 gives literature review. Section 3 describes mathematical backgroud and algorithms used for concept of algorithm that we proposed. Algorithm ih proposed in section 4. Example of using algorithm is presented in section 5. Paper concludes with section 6.

2. LITERATURE REVIEW

There are many publications in the area of artificial intelligence addressed to e-learning domain about personal learning process. Usually, proposed algorithms analyse behavioural of students. They applied sophisticated algorithms which support interactions with students and give the best learning outcomes. Some of authors describe their algorithms in papers [7, 8]. Chen presentes application of Item Response Theory, which is used to determine learners' abilities and course materials difficulties in reasoning process. Similar approach by facilitates navigation through e-learning system using history of users interactions and behaviour.

Other algorithms to learning content personalization use semantic web technologies. Most of such methods use ontologies to standardize student model, monitoring progress, notes and passed exams [9, 10]. Such model is used mainly to predict, which part of knowledge should be learn by a student as a next one. This model is used as a base for algorithm proposed in this paper.

More advanced concepts are developed in paper [11]. Significant improvement in personalization was given by Xu [12], which created system based on model of autonomous agents. The system was designed using three layers, responsible for creation of adaptive interface for online users, exchange of information between intelligent agents and gathering of data. Each agent in the middle layer is responsible for different issues i.e. users activity, learner profile, modelling and planning. The communication of agents results in personalized behaviour of e-learning platform.

3. MATHEMATICAL BACKGRAOUND

In this section linear model for classification and k-nearest neighbours' algorithm [14] are described. These algorithms is used in concept of proposed algorithm.

3.1. Linear model for classification

The goal in classification problem is to take an input vector **x** and to assign it to one of $\boldsymbol{K}$ discrete classes C_k where $k = 1, \ldots, K$. The classes are disjoint, so that each input must be assigned to exactly one class. There are classes of linear models which is used for solving classification problems. For this study least squares method was used for solving classification problem.

Prediction is based on large training set. Training set is consisted from **N** input vectors **x.** The categories of the input vectors in the training set are known in advance. Every input vector **x** is assigned with target vector **t.** For instance, if we have $K = 3$ classes, then a pattern from class 2 would be given the target vector $\mathbf{t} = (0, 1, 0)^T$. We can interpret the value of t_k as the probability that the vector x belongs to class C_k. Dimension of vector x is arbitrary and depends of situation in which algorithm is used. Dimension of vector t must be equals to number of classes.

The result of running the machine learning algorithm for classification can be expressed as a function **y**(**x**) which takes a new vector **x** as input and generates an output vector **y**, encoded in the same way as the target vectors. This function is linear combination coordinates from vector x and form of function is given with expresion (1),

$$y(\boldsymbol{x}, \boldsymbol{w}) = w_0 + w_1 \cdot x_1 + \ldots + w_D \cdot x_D \qquad (1)$$

where D denotes dimension of input vectors.

Algorithm has a goal to determine the precise form of the function **y**(**x**). For that, algorithm uses all vectors from training set. Unknown coefficients w_i are determined during the *training* phase, also known as the *learning phase*.

Once the model is trained it can be used for classification of new vector, which are said to comprise a *test set*. When coefficients w_i is known, function (1) can be used for generating output vector y for new vector x. Each coordinate in vector y denotes possibility that x is assigned to corresponding class.

a) **Discriminant Functions**

A discriminant is a function that takes an input vector **x** and assigns it to one of *K* classes, denoted *Ck*. The simplest representation of a discriminant function is obtained by taking a linear function of the input vector so that

$$\boldsymbol{y}(\boldsymbol{x}) = \boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x} + \boldsymbol{w}_{\boldsymbol{0}}$$

where vector **w** is called a *weight vector*, and $\boldsymbol{w}_{\boldsymbol{0}}$ is a *bias.*

Each class C_k is described by its own linear model so that

$$\boldsymbol{y}_{\boldsymbol{k}}(\boldsymbol{x}) = \boldsymbol{w}_{\boldsymbol{k}}{}^{\boldsymbol{T}}\boldsymbol{x} + \boldsymbol{w}_{\boldsymbol{k}\boldsymbol{0}}$$

where $k = 1, \ldots, K$. We can conveniently group these together using vector notations so that

$$\boldsymbol{y}(\boldsymbol{x}) = \widetilde{\boldsymbol{W}}^{\boldsymbol{T}} \cdot \widetilde{\boldsymbol{x}}$$

where, $\widetilde{\boldsymbol{W}}$ is a matrix whose *k*-th column comprises the *D*+1-dimensional vector

$$\boldsymbol{w}_{\boldsymbol{k}} = \left(\boldsymbol{w}_{\boldsymbol{k}\boldsymbol{0}}, \boldsymbol{w}^{\boldsymbol{T}}{}_{\boldsymbol{k}}\right)^{\boldsymbol{T}}$$

and **x** is the corresponding augmented input vector

$$\widetilde{\boldsymbol{x}} = (\boldsymbol{1}, \boldsymbol{x}^{\boldsymbol{T}})^{\boldsymbol{T}}$$

where $x_0 = 1$.

A new input **x** is then assigned to the class for which the output

$$\boldsymbol{y}_{\boldsymbol{k}} = \widetilde{\boldsymbol{w}_{\boldsymbol{k}}}^{\boldsymbol{T}} \cdot \widetilde{\boldsymbol{x}}$$

is largest.

Task is to determine the parameter matrix **W** by minimizing a sum-of-squares error function. Consider a training data set $\{\mathbf{x}_n, \mathbf{t}_n\}$ where $n = 1, \ldots, N$, and define a matrix **T** whose *n*-th row is the vector $t_n{}^T$, together with a matrix $\widetilde{\boldsymbol{X}}$ whose *n*-th row is $\widetilde{\boldsymbol{x}}_n{}^T$.

b) **Least squares for classification**

The sum-of-squares error function can then be written as

$$E_D(\widetilde{W}) = \frac{1}{2} Tr\left\{(\tilde{X}\widetilde{W} - T)^T \, (\tilde{X}\widetilde{W} - T)\right\} \qquad (2).$$

Where Tr denotes trace of a matrix.

Our task is to find solution for function (2) which gives the least possible value of function (1). Setting the expression's derivative with respect to $\widetilde{W}$ to zero, we obtained the solution for $\widetilde{W}$ in the next form

$$\widetilde{W} = (\tilde{X}^T\tilde{X})^{-1}\tilde{X}^T T = \tilde{X}^{\dagger} T$$

where $\tilde{X}^{\dagger}$ is the pseudo-inverse of the matrix $\tilde{X}$.
We than obtain the discriminant function in the form

$$y(x) = \widetilde{W}^T\tilde{\boldsymbol{x}} = T^T(\tilde{X}^{\dagger})^T\,\tilde{\boldsymbol{x}}$$

The least-squares approach gives an exact closed-form solution for the discriminant function parameters.

3.2. K-nearest neighbour's algorithm

Sometimes there is a need to determine the most similar elements from training set with a new one. There is a method which can be used for that. For this study, K-nearest neighbour's algorithm is used. This method calculates distances from new vector z to the all other elements from training set. For instance, Euclidian metrics can be used for calculating distance or some other metrics. The nearest K elements are these who have the smallest distances from new elements z. To do this, we consider a small sphere centered on the new vector **z** and we allow the radius of the sphere to grow until it contains precisely K data elements from training set. More about this method can be found in [14].

4. ALGORITHM

In this section an algorithm for personalized e-learning process is proposed. Algorithm uses linear model for students' classification and k-nearest neighbour's algorithm for generating learning path.

4.1. Structure of lessons

Each lesson is structured as a sequence of Learning Objects (LOs). In our case DITA Learning Objects are used. Every LO is independent piece of knowledge. LOs should not have connections to other LOs. Each LO is categorized by IEEE classification and has own level of knowledge. Level can be: Beginner, Middle or Advanced. When student is reading LOs he is tracked by time spending on each LO. Each LO should have at least 5 minutes of content and not more than 30 minutes of content [13]. By this time, we can conclude about students have pre-knowledge.

4.2. Concept of proposed algorithm

The student's learning path depends of his answers given on pre-lesson test. When student start with lesson for the first time, he must to do test. Test consists ten questions which answers are used for student categorization in one of three groups (Easy, Medium or Hard). Questions in test are divided in three groups. The first one has five easy questions, second group consists three medium questions and two hard questions are part of the last one. Test's results are saved as a binary 10-dimensional vector. If student answered correct to some question, appropriate coordinate in vector is set to 1, otherwise 0. After student has finished the test, he was categorized in student groups by test result. Minimization the least squares method is used as a linear model for classification. Unknwn coeficients in function (1) is calculated just ones, based on training set. Every time, after that calculus, functon (1) is used with known coeficients and in constant O(1) time gives as a result classification vector.

Training set for classification is generated by students's test where professor detemined in which class belongs each one. We created a model which classifies new user based on training set. When student is classified, goal is to generate learning path for him. There is a need to find K students which have the most similar answers on their tests. For finding the K nearest student to new one, authors used k-neighbors algorithm.

The closest K students are then considered for creating a learning path for student that have just done the test. The similiest K students are members of same cathegory as a new student. Learning path was built from LOs where average learning time by each of K user is at least 5 minutes.

Algorithm diagram is on Figure 3.1

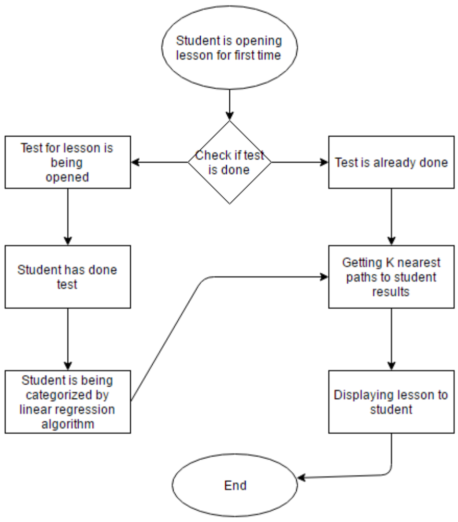

Figure 3.1: Proposed algorithm for personalized learning process

4.3. Implementation in LAMS

In this paper LAMS learning management system [15] of Metropolitan University is used for getting LOs of existing lessons. LAMS provides a teacher to create and deliver learning content, monitor student and assess student performance. LAMS also provides students to have ability to assess them self during learning process [16]. LAMS and a system for the storage and manipulation of learning objects (LOs) which is usually realized as a Content management system (CMS). We used database with already created DITA learning objects for testing our algorithm.
System for personalized learning process is created using PHP programming language with MySQL database, both for LAMS and for personalized learning system. From

LAMS database we are using users, courses, lessons and learning objects.

Tests for students are created in our Smart Personalized Learning Management System (Smart PLMS). Smart PLMS is responsible for linear classification and finding the most similar K students which were used for generating new personalised learning path which is different for each student.

5. EXAMPLE

Training set is created for testing proposed algorithm. This set consists of 100 student's tests and professor's categorization for each of them. Notice that each test is 11 digits' binary vector. First binary number in vector is always 1 and others are question results. For each binary vector x we have target binary 3-dimensional vector **t** with class where student who is test owner belongs. Vector **t** contains one element with value 1 and the all others have value 0. Index of element with value 1 is the same as index of target class.

As a result of linear classification we are getting 11x3 matrix W. When matrix W is multiplied by student's test we get 3-dimensional vector z. Each coordinate in vector z denotes possibility that x is element of corresponding class.

When student access the Smart LMS system with his username and password all courses screen is displayed like on Figure 5.1.

Figure 5.1: Courses on Smart LMS

After choosing course and lesson, test for lesson is being showed. Example of test is shown on Figure 5.2.
After student has done the test, result vector T was generate. Test student was done the test and his answers are in vector T:

$$T = [1, 1, 1, 1, 1, 1, 0, 1, 0, 0, 0]$$

Proposed algorithm calculated classification vector z with coordinates:

[0.1331198385 0.7149755415 0.418144297]

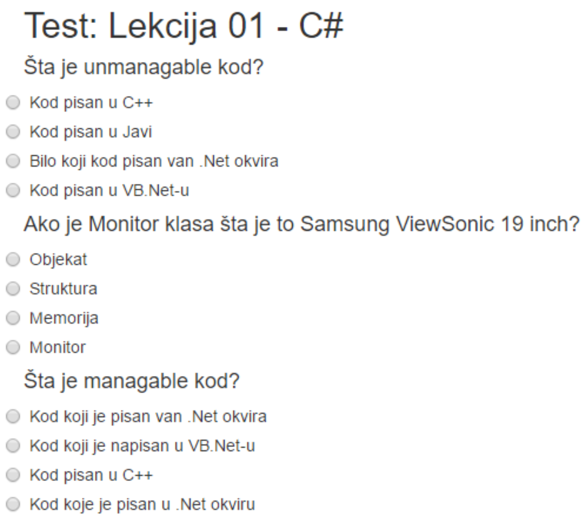

Figure 5.2: Example test for C# lesson 1.

The highest result is the student's category. In this case, the highest possibly has second class. So our student is part of Middle knowledge category. This result in accordance with our expectations, because student gives correct answers to questions from easy class (first five questions) and two from three questions from medium class. This test student didn't know answers to questions from advance class.

After student is being classified into category, there is a need to find K nearest student with similar test result. In our study, K is set on value five. For calculated K nearest student, we use Euclidean's metric. When five the most similar students to new one is known, last step is to generating learning path for him. If there are no five nearest students, student is getting learning path with all learning objects and then he is tracked for later K nearest tests. For generating learning path, it is necessary to determine for each LO will it be visible for student or not. We considered five vectors with times for each of five the most similar students which student spent on reading and learning each LO.

Dimension of these vectors is equal to number of LOs in that lesson. These vectors are not binary, there elements denote time spent by student, learning from corresponding LO. Time is expressed in seconds. If pervious student didn't read some LO, element on corresponding position is set on value 0. Finally, the output from proposed algorithm is binary vector, same length as total number of LOs in lesson. This vector consists information about which LO will be present student and which are not. On figure 5.3 there is an example of generated learning path for student.

After student finishes the lesson he has possibility to view all other LOs that were not used in his learning path. This enabled student to view learning objects that are maybe useful for his learning. Student that has custom learning path is also been tracked by time spending on each LO so K nearest student's results are getting better after each student finishes lesson.

Figure 5.3: Example of generated learning path for student

6. CONCLUSION

The goal of personalized learning process is student learning outcomes. This paper presents new approach to personalization of learning content implemented to the new platform which uses same resourses as exsisting LAMS platform. Thus, each student obtains new learning content, which is personalized to its needs and abilities, and improves efficiency of learning process. Created software is prepared to be used in distributed environment of e-learning platforms, however it requires implementation of web services, which would publish the courses from different platforms on the Internet.

Acknowledgment

The work presented here was supported by the Serbian Ministry of Education, Science and Technological Development (project III44006).

REFERENCES

[1] John Bordeaux, Organizational Knowledge Design (jbordeaux.com/all-learning-is-personalized)
[2] http://moodle.org
[3] http://sakaiproject.org
[4] http://www.atutor.ca
[5] Pahl C. „Managing evolution and change in web-based teaching and learning environments“, Computers and Education, vol. 40, pp. 99-114. 2003
[6] Sehring H.W., Bossung S., Schmidt J.W. (2005), Active Learning By Personalization - Lessons Learnt from Research in Conceptual Content Management, Proceedings of the 1st International Conference on Web Information Systems and Technologies, pp. 496-503.
[7] Chen C., Lee H., Chen Y. (2005), Personalized e-learning system using Item Response Theory, Computers & Education, vol. 44, pp. 237-255.
[8] Martinez M. (2002), What is personalized learning?, The E-learning Developers' Journal, pp. 1-7.
[9]Gascuena J.M. (2006), Fernandez-Caballero A., Gonzales P., Domain Ontology for Personalized E-Learning in Educational Systems, Sixth IEEE International Conference on Advanced Learning Technologies, pp. 456-458.
[10] Gomes P., Antunes B., Rodrigues L., Santos A., Barbeira J., Carvalho R. (2006), Using Ontologies for eLearning Personalization, 3rd E-learning Conference – Computer Science Education, Coimbra, Portugal.
[11] Henze N., Dolog P., Nejdl W. (2004), Reasoning and Ontologies for Personalized E-Learning in the Semantic Web, Educational Technology & Society, vol. 7, no. 4, pp. 82-97.
[12] Xu D., Wang H. (2006), Intelligent agent supported personalization for virtual learning environments, Decision Support Systems, vol. 43, pp. 825-843.
[13] Kelvin Thompson, Francisa Yonekura (2005), Practical Guidelines for Learning Object Granularity from One Higher Education Setting, Interdisciplinary Journal of Knowladge and Learning Objects, vol 1, pp. 177
[14] Christopher M. Bishop, "Pattern Recognition and Machine Learning ", Springer 2006.
[15] https://www.lamsfoundation.org/
[16] S. Cvetanović, M. Raspopović, A. Arsić, V. Vasić, „Integration software architecture of e-learning system with Facebook“, The Sixth International Conference on e-Learning (eLearning-2015), 24 - 25 September 2015, Belgrade, Serbia