Proceedings of the Federated Conference on
Computer Science and Information Systems pp. 739–744

ISBN 978-83-60810-51-4

The Design of eLeTK – Software System for Enhancing On-Line Educational Environments

Marian Cristian Mihăescu
Software Engineering Department
University of Craiova
Craiova, Romania
mihaescu@software.ucv.ro

Abstract—This paper presents the design of eLeTK, which is a software system that may be used for enhancing on-line educational environments. The main concept introduced in this paper is represented by educational data/knowledge flow. The data flow is transformed into a knowledge flow provided that all input data represents activity produced by an on-line educational environment. On the other hand, the output of the software system is redirected towards the educational environment in an attempt to enhance its capabilities. eLeTK may become a recommender system for students or professors, a knowledge self-assessment tool for students or an custom learning path builder. The core business logic of eLeTKis represented by custom integration of different machine learning algorithms adapted to work with data provided by on-line educational environments.

Keywords-educational data mining, software system, toolkit

I. INTRODUCTION

THIS paper addresses the problem of enhancing on-line educational environments. Enhancing is obtained by providing platform side intelligent functionalities in the shape of a recommender system or learning path builder. Such enhancements are obtained when several conditions are met: the courses have a well-structured hierarchical nature, necessary experiences are stored and a feasible data analysis process is set up.

Each of the above presented prerequisites is equal important. The first prerequisite is concerned with having a proper infrastructure setup. This means that the on-line educational environment has a well refined hierarchical structure in which all learning assets (e.g., disciplines, chapters, quizzes, etc.) are properly defined. For example, at chapter level there needs to be defined a concept map [1] with which all other learning assets are associated. That is why, each quiz question associated to a chapter is also associated with a concept.

The second prerequisite is concerned having enough data for feeding a data analysis process. The data represents experiences had by the involved partied during usage of the on-line educational system. From this point of view, two aspects are equally important: the number of features representing a user and the overall quantity of logged data. As a general key aspect the data needs as many conditionally independent attributes as possible and as much data as it can be obtained. Still, these aspects are not key aspects since more data and more feature do not necessarily mean an increase in the accuracy of data analysis process. This concern is mainly addressed by the effective debugging of the data analysis process.

The third prerequisite is concerned with setting up an adequate data analysis process. From this point of view two issues need to be addressed: setting up a proper requirement and choosing the right algorithm that produces usable and interpretable results. In general, there are two types of algorithms: unsupervised and supervised. The unsupervised algorithms (e.g., clustering, regression, etc.) are used to discover patterns in data. The supervised algorithms (e.g. classification, decision trees, SVM, neural networks, etc.) are used to classify new items, which in educational applications may be students or sometimes professors. The key aspect of the data analysis process is evaluating the quality of the data analysis process. This evaluation gives confidence in using the final results and may provide important information regarding necessary actions needed to improve the analysis process. The continuous improvement in quality of the data analysis process is the key aspect in having a truly machine learning based data analysis.

The design of eLeTK (e-Learning Enhancer Toolkit) is modular in packages such that continuous development is feasible. The main packages are: input loader, filters, data processing, evaluation, output builder and configuration. Each package contains classes that implement specific business logic such that they may be put together to form a data/knowledge flow.

The proposed design of eLeTK makes it suitable for building systems that run along on-line educational environments and enhance their educational purposes. From a software systems point of view, a setup of eLeTK works as a service for an on-line educational environment in an attempt to offer the intelligent character.

II. RELATED WORK

In the last decades there has been a lot of effort in the new domain of EDM (Educational Data Mining). In [2] there are presented many tools that were designed and implemented for this domain. The main issues that distinguish educational data mining from other domains where such state of the art algorithms are used are related to

domain specific data, domain specific objectives and goals, custom adaptation of classical data analysis techniques and custom interpretation and visualization of results.

The general approach of such systems is based on the some core processes. Firstly, it is assumed that academics responsible and educators design, plan, build and maintain an educational environment. Second, students interact with the educational environment thus producing interaction data. Interaction data along with educational environmental data (e.g., course information, academic data, quizzes, etc.) represent one input in the educational data-mining tool, which in our case is eLeTK. The data-mining tool is used to show discovered knowledge, recommendations or learning paths to students or other involved parties.

There are many general data analysis tools which are not designed to work with data from a specific domain. Among such tools there are DBMiner, Clementine, Intelligent Miner, Weka, etc. [7]. The main drawback of these tools in the context of EDM is that they can not be used in educational contexts by students or professors. In a most optimistic case, these tools are used by experienced data analysts with a good background in a specific on-line educational environment. In this way there are performed an off-line data analysis which may produce knowledge regarding analyzed data or recommendations for involved parties. From this point of view, eLeTK represents a new layer between a general data analysis tool and a specific on-line educational environment.

So far, in the above presented general context there were created tools oriented towards educators [3, 4] and tools oriented towards academics responsible and administrators [5, 6]. These tools perform tasks as associations, pattern analysis, classification, clustering, text mining, statistics and visualization.

In the area of on-line educational environments, there are two types of systems: classical learning content management systems and intelligent web-based educational systems. Some examples of commercial learning content management systems are Blackboard, Virtual-U, WebCT, TopClass, etc. and some example of free LCMS are Moodle, Ilias, Claroline, aTutor, etc. [8]. On the other hand, some examples of intelligent educational systems are SQL-Tutor, German Tutor, ActiveMath, VC-Prolog-Tutor, AHA!,InterBook, KBS-Hyperbook, WebCOBALT [9].

The main output of all web-based educational systems is the activity data performed by their users. These web-based education systems can normally record the student's accesses in web logs that provide a raw trace of the learners' navigation on the site. There are several types of logs [17] and there are also AI techniques for monitoring student learning process [18].

The main tasks that are generally implemented are data preprocessing (e.g., data cleaning, user identification, session identification, transaction identification, data integration) [10, 11], data analysis (e.g., decision tree construction, rule induction, artificial neural networks, instance-based learning, Bayesian learning, logic programming, statistical algorithms, etc.) [7]or web mining (e.g., clustering, classification, outlier detection [7], association rule mining, sequential pattern mining [12], text mining [13]).

One of the main drawbacks of this approach is that is fully data centered. Thus, the change regarding user preferences [14] aspects are neglected.

A plus in the domain is brought by custom usage of a hierarchical way of structuring e-Learning content. The conceptual-visual dynamic schemes (CVD-schemes) are the marked oriented graphs introduced in cognitonics domain [15, 16] for inventing effective teaching analogies. Such graphs establish a correspondence between the components of a piece of theoretical material to be studied and the components of a well-known or just created by the teacher but bright fragment of the inner world's picture of the learner.

III. Employed Infrastructure and Methods

A. Tesys On-Line Educational Environment

Tesyson-line educational environment is primarily a classical collaborative software system in which all involved parties (e.g., administrators, professors and students) perform their main responsibilities. Administrators are responsible for managing the general infrastructure as curriculum of learning programs (i.e., the studied disciplines and the assigned professors), users, etc. Some examples of actions that are currently performed by administrators are enrolling students to needed learning programs, giving them grants to take the failed exams, passing them into the next year of study, communicating with students and professors.

An importantaspect regarding the functionality of the e-Learning regards the types of activities the students are performing. Some of the currently implemented activities are login, logout, taking tests and exams, communicating with other students or with professors, etc.All these activities represent a repository of experiences that are very valuable in a data analysis process.

Currently, the Tesys e-Learning platform has five setups each with one up to five study programs. For example, one setup manages four different study programs with duration of three years where more than 100 professors and almost 1000 students are currently active. This setup manages the following learning assets: 120 courses, almost 1000 chapters, almost 5000 quiz questions, almost 1000 taken quizzes and exams and almost 10,000 sent messages. All performed activity is logged into files or in a database. In this way, for a student there may be computed a lot of features describing the performed activity.

In general the features are of two types: one regards indicators of the quiz activities and one regards the time spent on different activities. Some of the features from the first category are:positivCount – the number of correctly answered questions; correctPercent – the percentage of correctly answered questions from the total number of questions; totalTries – the total number of tries (answered questions); avgTries – medium number of tries per question. Some of the features from the second category are avgQuestionTime – on average, how long (in minutes) it takes for a student to answer a question; totalTime – total time spent on testing.

B. Data Analysis Techniques and Weka

The data analysis [20] techniques fall in general three categories: unsupervised, supervised and rule based. The most common unsupervised method is clustering and may be of several types: partitioning, EM, hierarchical, fuzzy, etc. The main supervised methods relate to classification algorithms: Decision Trees, Bayesian Networks, Vector Space Classification, CART. The most common algorithm for building association rules is Apriori.

All these algorithms are mature and proved their effectiveness in different domains outperforming classical statistical data analysis procedures.

Weka [21] is a java software implementation of a large set of machine learning and data mining algorithms. The implementations are generic and therefore virtually data coming from any domain (e.g., e-commerce, bioinformatics, e-Learning, etc.) may be analyzed. From this point of view, the main issue regards designing of new algorithms, implementing them into Weka (or similar libraries) in an attempt of improving the time and space complexity.

IV. eLeTK Architecture

A. General Requirements

The main issues related to existing libraries implementing machine learning and data mining algorithms relate to: a) are based on non-extensible frameworks, b) it is relatively difficult to integrate and accommodate a domain specific machine-learning problems, c) do not provide flexible, uniform and consistent integration mechanism.

Designing and developing a software system that addresses the above presented issues in the domain of Educational Data Mining is an interdisciplinary problem related to the following equally important areas: Machine-Learning/Data Mining/Information Retrieval, Software Engineering and Human Computer Interaction.

The constraints from the algorithms point of view regards the diversity of algorithms, the wide range of data models, the difficulty of obtaining interoperability among models and difficulty of integration into specific information systems which represent the main source of data. For example, Wekaworkbench is quite weak regarding the interoperability among different data modeling methodologies.

The main general activities that need to be managed regard capturing raw data and representing items, performing a data analysis process, management of obtained data models, assessing the performance of the obtained models and presentation into an understandable format to users. These processes need robust and scalable data management in a flexible integration architecture.

The overall activity is regarded as a data workflow. The processing pipeline of the data workflow always starts from raw data provided by the Information System. During the data analysis data may take different shapes such as structured as needed by the data analysis engine, as a data model, knowledge or other type of representation. The activities that may be performed upon different types of data are streaming, loading, modeling, moving, recording, replicating or crawling. All these activities may be performed also as a scheduled job, without administrator's presence. Each activity may be defined as a data workflow in which there are specified all the necessary activities, such as data preprocessing and translation, data consistency verification, data quality check or data validation. Data workflow may also be merged, copied, crawled or recovered.

One of the most important core information about eLeTK is the Data Resource Registry (DSR) which records metadata about performed activities, available services, data, models, knowledge or workflows. DSR manages the core assets handled by an instantiation of eLeTK.

An important activity within the data workflow is the performance assessment. This activity builds trust into eLeTK and makes users (e.g., administrators, end-users, etc.) confident that the obtained knowledge may be safely used. The first step in assessment is the error analysis, which needs error metrics (e.g., precision, recall, FMeasure, etc.) that are in close relation with the employed data analysis technique. An important activity is the debugging of the machine learning process. This takes care of learning curves, convergence problems and provides a deep insight of the data analysis process such that it is obtained a clear feedback regarding necessary actions that may bring further improvements. The most common actions are to collect more data, try a smaller/larger number of parameters for items representation, avoid overfitting/bias or add polynomial features. Regarding this aspect, plotting learning curves, cost functions or other specific quality indicators may give also important indications regarding what actions may be tried when debugging the data analysis process.

Another important activity that is performed within eLeTK is the evaluation of the obtained model/hypothesis. This is accomplished by using training and testing sets in a 10-fold cross validation scheme. A model selection procedure may be also created by using a validation set and by computing validation errors for each challenger model that is taken into consideration.

B. Data Analysis Workflow Design

The data analysis workflow (DAW) design meets certain requirements that make sure the business goals of eLeTKare met. The DAW implements a simple query model. This means that creation DAWs starts with setting up raw source data, specifies the preprocessing activities, specifies the algorithmic pipeline, the model representation, the validation methodology and the output representation and visualization. Each DAW has the goal of minimizing data movements by having a preliminary estimation of job size in time and memory. An important part of the workflow is represented by the scheduler system, that specifies the frequency by which a DAW is executed.

The data analysis workflow is a pipeline which has the main layers/components: data layer, loader layer, modelers, performance assessors, viewers, fault handlers and configuration management.

The design of the DAW is created such that end-user simplicity and operational are ensured. This issue mainly implemented by custom wizards that allow quick and effective DAW creation.

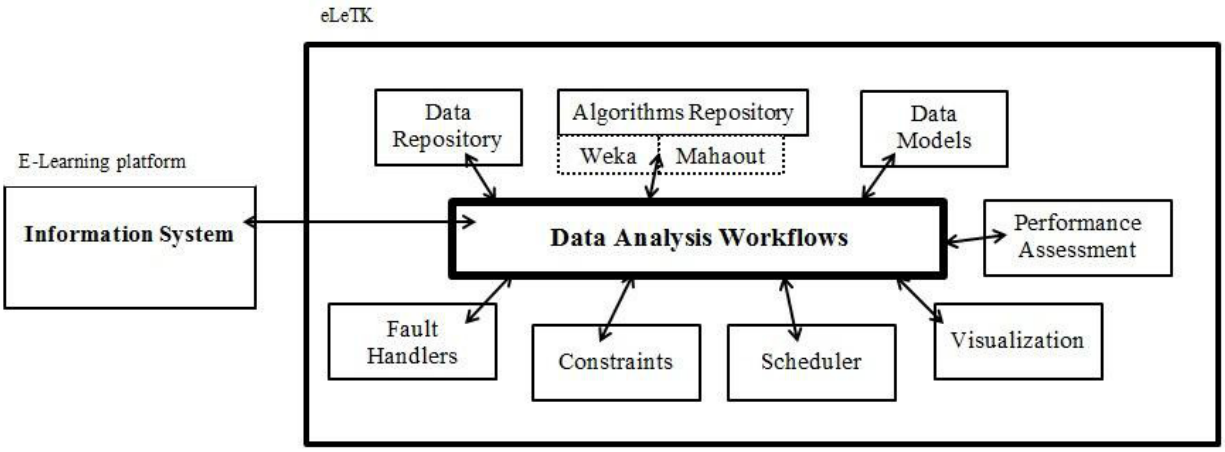

Figure 1. eLeTK System Overview

The Data Repository represents the place where all raw data collected from the information system is collected. The Algorithms Repository is represented by Weka or other library that implements necessary algorithm or extensions of already implemented algorithms. The Data Models manages the obtained knowledge. It may be represented by a set of clusters, rules or classifiers. The Performance Assessment module manages different schemes implementing various errors metrics. The implemented error metrics may be used in conjunction with corresponding algorithms that are managed by the Algorithms Repository. The Fault Handlers implement the verifications regarding correct data integration, data availability and data integrity. The Constraints module manages the specific context variables in which a workflow runs. The constraints regard quality metrics thresholds, number of clusters, rules selection strategies expressed at highest level available for eLeTK managers. The Scheduler module is responsible for the way each DAW runs. Setting up a scheduled job is the last activity that needs to be performed after the data analysis pipeline is set up for a new DAW. A scheduled job may be run at request or at certain intervals of time. Visualization module is available only for development and debugging purposes.

C. Software Architecture of eLeTK

The software architecture of eLeTK follows the above presented general system specifications. It ensures the proper integration of different modules is such a way that development (e.g. adding new data models or new fault handlers) is performed in a productive manner.

Another property of the software architecture is modifiability. This means that steps that make up a DAW can be added/edited/deleted in a reasonable way.

The software architecture integrates a logging mechanism such that error analysis and debugging may be performed with rapid discovery of faults.

There are two types of users for the eLeTK system. One is represented by the system administrators. Their main job is to set up the DAWs by specifying raw data sources, data modelers, performance assessors, fault handlers and a schedule. The administrators need to be experienced data analysts such that created DAWs have a good structure, be reliable and provide usable and high quality knowledge. The second type of users is represented by ones who just run the DAWs. They are represented by users from of the information system (e.g., students, professors, etc.) and they are regarded as end-users. From this perspective they represent the main beneficiaries of the eLeTK system.

Within the software architecture there is implemented a Configuration Manager which is responsible adding/editing/deleting raw data sources, available algorithms and data models, current constraints, etc. The Configuration manager is also responsible for checking the health of existing DAWs and launch reliable fault recovery processes that make sure that data consistency is preserved. In this context, a performance watchdog is very necessary tool in order to have a quick information regarding failures and heavy data workloads.

V. Sample Setup and Usage Scenario

The first step is to perform eLeTK setup. The final goal of a setup procedure is to create a valid DAW. This requires a minimum configuration of a raw data source, a data model, a constraints setup and a set of visualization rules.

The raw data source may be the database of the information system. If this is the case, the credentials of the database (e.g., URL, username, password) must be provided and a scheduled job may be defined to bring the records into the Data Repository in a structured way (e.g., XML). Another scheduled job may be created to create a data file according with the format required by the algorithm and used implementation. Once the training and testing data is available the data model may be created.

Here is a sample arff file that may be used by algorithms implemented in Weka workbench.

```
@relation activity
@parameter avgTimeSpent{0, 1}
@parameter correctPercentOfAnswers{0, 1}
```

@paremeter class {low, average, high}

@data
100, 65, average
200, 85, high
…

The data model may be represented by a classifier. Thus, an in memory classifier is available for classifying new data. Some of the most common classification algorithms implemented by Weka are Decision Trees (e.g., ID3, J48, CART), Naïve Bayes or even advanced algorithms such as Support Vector Machines.

Here is a sample decision tree obtained form the above sample dataset.

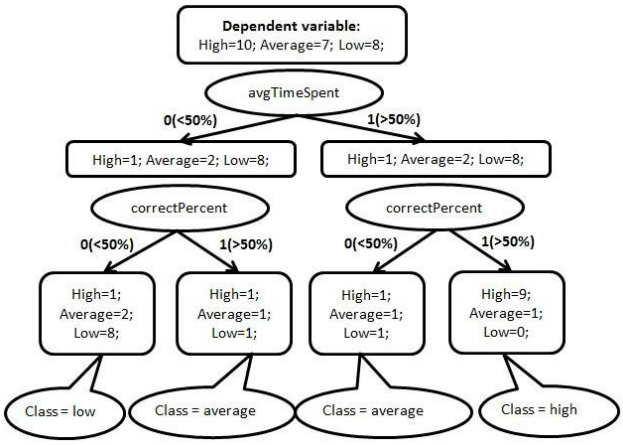

Figure 2. Sample Decision Tree

The constraints setup contains the thresholds for specific error metrics (e.g., precision, recall, FMeasure, etc.) that allow a reliable classification. In this wayeLeTK provides also an indication regarding the confidence in obtained results. The constraints may also be specific to information system users such as students or professors. For example, a student may require the classification of available resources such that his target class is high. Another example, from the perspective of professors is to set up a higher level for FMeasure value regarding the classification of students that may start studying the next chapter. In this way, the effectiveness of the e-Learning system is increased.

The above presented decision tree represents an in-memory data model which may be used when needed to classify a new student. For example, a forest of decision treesmay be created such that a decision tree is created for each asset associated to a chapter from a discipline. Let us suppose that the student sets his target class as high. The forest of decision trees may be queried and all the assets that are classified as average or low may be obtained with a corresponding information regarding the activities that must be performed.

The visualization rules provide the real and final expression of what is behind a certain assignment of an item into a certain class. Taking into consideration the above scenario one output of the visualization module may have the following form:

Dear Student John Doe,
The eLeTKclassified you as **average** and advices you to:
Study more on "definitions"
Study more on "tree traversals"
Redo quizzes on "rotations".

In a similar way, a classification of students gives important feedback to professors. For example, a professor may query the current situation of enrolled students and discover that here are students which spent lots amounts of time in certain activities (e.g., study and/or have quiz activity only regarding some concepts) although their classification is still bad. In this situation,eLeTK provides important feedback regarding an optimal learning path that need to be taken into consideration. A sample feedback for the professor may have the following form:

Dear Professor John Doe,
Student Jack Dow spends a lot of effort compared to his knowledge achievements. You may advise him to:
*Study more on "*graph representations*"*
*Study more on "*connectivity*"*

In this manner, there may be created a wide range of Data Analysis Workflows, each of them solving a particular problem.

VI. Future Work

This paper presents the design of a software system, called eLeTK, whose aim is to run along on-line educational systems. The main purpose of eLeTK is to offer feedback for users of the on-line educational system in an attempt to enhance the educational proficiency. The intuition of such a system is that it represents a substitute for the knowledge that a real professor acquires in a face-to-face educational systems.

The main advantages of eLeTK is that it has access to all performed activity of students and that the data analysis procedures are hundred percent objective. In classical, face-to-face educationalsystems the main issue regards the ability of the professor to effectively and objectively assess and guide the student.

The main target of eLeTK is to reach a high level of accuracy and thus give the classical on-line educational system the possibility to act as high quality traditional learning environment.

The software architecture is a modular one and is built around the idea of Data Analysis Workflow. A DAW represents a data processing pipeline which is custom designed to accommodate data coming from on-line educational systems.

From a general perspective, eLeTK may work as a service virtually for any on-line educational system. The main requirements for the on-line educational system are to offer access to activity data, to have a custom setup procedure according with provided data, to create required DAW and to accommodate feedback offered by eLeTK.

The future works regard the development of a quality assessment procedure of the offered feedback. This

assessment procedure is supposed to offer valuable feedback that may be used in continuous improvement of eLeTK.

Future works also regard continuous improvements of existing modules by integration of more advanced state of the art algorithms, performance assessment procedures, visualization capabilities of results and other new features.

ACKNOWLEDGMENT

This work was supported by the strategic grant POSDRU/89/1.5/S/61968, Project ID61968 (2009), co-financed by the European Social Fund within the Sectorial Operational Program Human Resources Development 2007 – 2013.

REFERENCES

[1] J. D. Novak andA. J. Cañas, "The Theory Underlying Concept Maps and How to Construct and Use Them", Technical Report IHMC CmapTools, 2006.

[2] C. Romero and S. Ventura, "Educational Data Mining: A Survey from 1995 to 2005", Expert Systems with Applications, 33(1), pp. 135-146, 2007.

[3] C. Romero,S. Ventura, P.D. Bra, "Knowledge discovery with genetic programming for providing feedback to courseware author". User Modeling and User-Adapted Interaction: The Journal of Personalization Research, 14(5), pp. 425–464, 2004.

[4] A. Merceron, K. Yacef, "Mining student data captured from a web-based tutoring tool: Initial exploration and results", Journal of Interactive Learning Research, 15(4), pp. 319–346, 2004.

[5] H. Grob, F. Bensberg, F. Kaderali, "Controlling open source intermediaries – a web log mining approach", In Proceedings of the 26th international conference on information technology interfaces pp. 233–242, 2004.

[6] T. Urbancic, M. Skrjanc, P. Flach, "Web-based analysis of data mining and decision support education" AI Communications, 15, 199–204, 2002

[7] W. Klosgen, J. Zytkow, "Handbook of data mining and knowledge discovery", New York, Oxford University Press, 2002.

[8] M. Paulsen, "Online education and learning management systems", Bekkestua: NKI Forlaget, 2003.

[9] P. Brusilovsky and C. Peylo, "Adaptive and intelligent web-based educational systems, International Journal of Artificial Intelligence in Education, 13, 156–169, 2003.

[10] M. Koutri, N. Avouris, S. Daskalaki, "Ch. A survey on web usage mining techniques for web-based adaptive hypermedia systems", 2004.

[11] M. E. Zorrilla, E. Menasalvas, D. Marin, E. Mora, J. Segovia, "Web usage mining project for improving web-based learning sites", In Web mining workshop, Cataluna, 2005.

[12] R. Agrawal, R Srikant, "Mining sequential patterns", In Eleventh International conference on data engineering (pp. 3–14). Taipei, Taiwan, IEEE Computer Society Press, 1995.

[13] M. Grobelnik, D. Mladenic, M. Jermol, "Exploiting text mining in publishing and education", In Proceedings of the ICML-2002 workshop on data mining lessons learned (pp. 34–39), 2002.

[14] A. Burlea Schiopoiu, A. Badica, C. Radu, "The evolution of e-learning platform TESYS user preferences during the training processes", In Proceedings of ECEL2011, 11-12 Novembre Brighton, UK, pp.754-761, 2011.

[15] V. Fomichov and O. Fomichova, "The Theory of Dynamic Conceptual Mappings and its significance for Education, Cognitive Science, and Artificial Intelligence.", Informatica. An Intern. Journal of Computing and Informatics (Slovenia), 1994, 18 (2).

[16] V. Fomichov and O. Fomichova, "Cognitonics as a New Science and Its Significance for Informatics and Information Society", Special Issue on Developing Creativity and Broad Mental Outlook in the Information Society (Guest Editor Vladimir Fomichov), Informatica. (Slovenia), 2006, 30 (4).

[17] Srivastava, J., Cooley, R., Deshpande, M., & Tan, P, "Web usage mining: Discovery and applications of usage patterns from web data", SIGKDD Explorations, 1(2), 12–23, 2000.

[18] David Camacho, Álvaro Ortigosa, Estrella Pulido, María D. R-Moreno."AI techniques for Monitoring Student Learning Process". In Advances in E-Learning: Experiences and Methodologies. A book edited by Francisco J. Garcia-Peñalbo. Publisher: Information Science Reference-IGI Global, USA. Chapter 9, pp. 149-172, 2008.

[19] Dumitru Dan Burdescu, Cristian Marian Mihăescu, "Tesys: e-Learning Application Built on a Web Platform", International Joint Conference on e-Business and Telecommunications -International Conference on e-Business (ICE-B), , pp.:315-318, 2006.

[20] Jiawei Han, Micheline Kamber and Jian Pei, Data Mining: Concepts and Techniques, 3rd ed., The Morgan Kaufmann Series in Data Management Systems, 2011.

[21] Witten, I. H. & Frank, E., Data Mining: Practical Machine Learning Tools and Techniques (Second Edition), Morgan Kaufmann, 2005.