

# A Collaborative Educational Association Rule Mining Tool

## Abstract

This paper describes a collaborative educational data mining tool based on association rule mining for the ongoing improvement of e-learning courses and allowing teachers with similar course profiles to share and score the discovered information. The mining tool is oriented to be used by non expert instructors in data mining so its internal operation has to be transparent to the user and the instructor can focus on the analysis of the results and make decisions about how to improve the e-learning course. In this paper, a data mining tool is described in a tutorial way and some examples of rules discovered in an adaptive web-based course are shown and explained.

*Keywords: Educational Data Mining Tool, Association Rule Mining, Collaborative Recommender System*

## 1. Introduction

Educational Data Mining (EDM) is the application of Data Mining (DM) techniques to educational data (Romero and Ventura, 2007). EDM has emerged as a research area in recent years for researchers all over the world in many different areas (e.g. computer science, education, psychology, psychometrics, statistics, intelligent tutoring systems, e-learning, adaptive hypermedia, etc.) when analysing large data sets in order to resolve educational research issues (Backer and Yacef, 2010). On one hand, the increase in both instrumental educational software as well as state databases of student information has created large repositories of data reflecting how students learn (Koedinger et al., 2008). On the other hand, the use of Internet in education has created a new context known as e-learning or web-based education in which large amounts of information about teaching-learning interaction are

endlessly generated and ubiquitously available (Castro et al., 2007). All this information provides a gold mine of educational data (Mostow & Beck, 2006).

The EDM process converts raw data from educational systems into useful information that can be used by educational software developers, teachers, educational researchers, etc. This process does not differ much from other application areas of data mining like business, genetics, medicine, etc. because it follows the same steps. The educational data mining process (Romero et. al., 2004) is based on the same steps as the general data mining process (see Figure 1), seen in continuation:

- Preprocessing. The data obtained from the educational environment has to first be preprocessed to transform it into an appropriate format for mining. Some of the main preprocessing tasks are: cleaning, attribute selection, transforming attributes, data integration, etc.
- Data mining. It is the central step that gives its name to the whole process. During this step, data mining techniques are applied to previously preprocessed data. Some examples of data mining techniques are: visualization, regression, classification, clustering, association rule mining, sequential pattern mining, text mining, etc.
- Postprocessing. It is the final step in which the results or model obtained are interpreted and used to make decisions about the educational environment.

**Figure 1.** Educational Data Mining process.

Nowadays, a lot of general data mining tools and frameworks are available to DM users. Some examples of commercial mining tools are DBMiner (DBMiner, 2010), SPSS Clementine (Clementine, 2010) and DB2 Intelligent Miner (Miner, 2007). Some examples of

public domain mining tools are Weka (Weka, 2010), Keel (Keel, 2010) and Rapid Miner (RMiner, 2010). However, all these tools are not specifically designed for pedagogical/educational purposes and are cumbersome for an educator to use since they are designed more for power and flexibility than for simplicity (Romero et al., 2008b). On the other hand, there are also an increasing number of mining tools specifically oriented to educational data such as: Mining tool (Zaiiane and Luo, 2007) for association and pattern mining, MultiStar (Silva and Vieira, 2002) for association and classification, EPRules (Romero et al., 2004) for association, KAON (Tane et al., 2004) for clustering and text mining, Synergo/ColAT (Avouris et al., 2005) for statistics and visualization, GISMO (Mazzaa and Milani, 2005) for visualization, TADA-Ed (Merceron and Yacef, 2005) for visualizing and mining, etc. However, most of the current EDM tools are complex for educators to use and their features go well beyond the scope of what an educator may want to do. So, these tools must have a more intuitive interface that is easy to use, with parameter-free data mining algorithms to simplify the configuration and execution, and with collaboration facilities to make their results available to other educators or e-learning designers (Garcia et al., 2009b).

This paper describes an educational data mining tool based on association rule mining and collaborative filtering for the continuous improvement of e-learning courses. The main objective is to make a mining tool in which the information discovered can be shared and scored by different instructors and experts in education. The paper is organized as follows: Section 2 is a background about educational data mining tools; Section 3 describes the proposed mining tool and shows some examples of discovered rules. Finally, future lines of research and conclusions are outlined in Section 4.

## 2. Educational Data Mining Tool Background

Nowadays, EDM researchers have developed a lot of educational data mining tools to help EDM users (teachers/instructors/academic responsibility/course developers/learning providers/etc.) to solve different educational problems and objectives (see Table 1).

**Table 1.** List of EDM tools.

However, all these tools are oriented for use by a single user in order to discover useful knowledge in his own courses. So, they do not allow collaborative usage in order to share all the discovered information with other users who use similar courses (contents, subjects, educational type: elementary and primary education, adult education, higher, tertiary and academic education, special education, etc.). In this way, the information discovered locally by some users could be joined and stored in a common repository of knowledge available for all users in order to solve similar problems when detected.

One of the most commonly used data mining techniques in the above-mentioned tools is association rule discovery (Agrawal et al., 1996). Association rules are one of the most popular ways of representing discovered knowledge and describing a close correlation between frequent items in a database. An  $X \Rightarrow Y$  type association rule expresses a close correlation between items (attribute-value) in a database. There are many association rule discovery algorithms (Zheng et al., 2001) but Apriori is the first and foremost among them (Agrawal et al., 1996).

Most association rule mining algorithms require the user to set at least two thresholds, one of minimum support and the other of minimum confidence. The support  $S$  of a rule is defined as the probability that an entry has of satisfying both  $X$  and  $Y$ . Confidence is defined as the probability an entry has of satisfying  $Y$  when it satisfies  $X$ . Therefore the aim is to find all the

association rules that satisfy certain minimum support and confidence restrictions, with parameters specified by the user. A really important improvement to the Apriori algorithm for use in educational environments is the Predictive Apriori (Scheffer, 2005) because it does not require the user to specify any of these parameters (either the minimum support threshold or confidence values). The algorithm aims to find the  $N$  best association rules, where  $N$  is a fixed number. The Predictive Apriori (PA) algorithm strikes an appropriate balance between support and confidence to maximize the probability of accurately predicting the dataset. In order to achieve this, the PA algorithm, using the Bayesian method, proposes a solution that quantifies the expected predictive accuracy ( $E(c | \hat{c}, s)$ ) of an association rule  $[x \Rightarrow y]$  with given confidence  $\hat{c}$  and the support of the rule's body (the left hand side of the rule) of  $s$ .

### 3. Mining Tool Description and Tutorial

In this paper, a data mining tool with two subsystems is proposed: a client application and a server one (Figure 2). The client application uses an association rule mining tool to uncover interesting relationships found in student usage data through IF-THEN recommendation rules. The server application uses a collaborative recommender system to share and score the rules previously obtained by instructors in similar courses with other instructors and experts in education.

**Figure 2.** Collaborative data mining tool.

As can be seen in Figure 2, the system is based on client-server architecture with  $N$  clients, which applies an association rule mining algorithm locally on student usage data. In fact, the client application uses the Predictive Apriori algorithm. The only parameter is the number of rules to be discovered, which is a more intuitive parameter for a teacher who is not an expert in data mining. The association rules discovered by the client application must be evaluated

to decide if they are relevant or not; therefore the client application uses an evaluation measure (García et al., 2009a) to classify if the rules are expected or unexpected, comparing them to the rules that have been scored and stored in a collaborative rules repository maintained on the server side. Also, the expected rules found are then expressed in a more comprehensible recommendation format of possible solutions for problems detected in the course. The teacher sees the recommendation and can determine if it is relevant or not for him/her in order to apply/use the recommendation. On the other hand, the server application allows the management of the rules repository by using collaborative filtering techniques with knowledge-based techniques (García et al., 2009a). The information in the knowledge base is stored in the form of tuples (rule-problem-recommendation-relevance) which are classified according to a specific course profile. The course profile is represented as a three-dimensional vector related to the following characteristics of his/her course: Topic (the area of knowledge, e.g. Computer Science or Biology); Level (level of the course, e.g. University, High School, Elementary or Special Education); and Difficulty (the difficulty of the course, e.g., Low or High). These similarities between courses are available for other teachers to assess in terms of applicability and relevance. A group of experts propose the first tuples of the rule repository and also vote on those tuples proposed by other experts. On the other hand, teachers could discover new tuples (in the client application) but these must be validated by the experts (in the server application) before being inserted into the rule repository. In our experiments, three experts in the Computer Sciences and Artificial Intelligence area in Cordoba University, Spain have also participated and were responsible for proposing the initial tuples in the repository. And there have also been two other teachers from the same area involved (the authors of the courses themselves).

### 3.1. Client Application

The main feature of client application is its specialization in educational environments. Domain specific attributes, filters and restrictions for the rules, and student usage datasets from the e-learning course have been used to achieve this. The interface for client application has the following four basic panels.

**Pre-processing panel.** Before applying a data mining algorithm, the data has to be pre-processed in order to be adapted to our data model. Figure 3 shows the pre-processed panel of CIECoF (*Continuous Improvement of E-learning Courses Framework*) tool, using red icons to highlight the different steps the user must follow. Next, the main functionality of each step is described, represented as a red square in the figure:

1. **Select Data:** this is the first step, where the teacher has to select the origin of the data to be mined (see Figure 3). There are two different formats available for input data: 1) the Moodle relational database, for teachers that work with Moodle as well as the INDESAHC authoring tool (García et al., 2009), so all our attributes are used directly; or 2) a Weka (Witten and Frank, 2005) ARFF text file, for teachers that use other LMSs and, therefore, other attributes.
2. **Attributes:** if the teacher selects an ARFF file as the data source, the application only shows the teacher the numerical attributes detected in the file. On the other hand, if the teacher selects a Moodle database compatible with CIECoF attributes, the teacher can select from among different tables, such us *course*, *unit*, *lesson*, or *exercise*, among others.
3. **Data summary:** the teacher can consult all the attributes and instances in the selected data, determining the quantity of nominal and numerical attributes.

4. **Selected attribute:** this step allows the teacher to transform the numerical attribute selected in step 2 into a nominal attribute. The objective is to make the rules discovered easier to understand and also to significantly reduce the mining algorithm's running time. The transformation into discrete variables can be seen as a categorisation of attributes that take a small set of values. The basic idea involves partitioning the values of continuous attributes within a small list of intervals. Our process of discretization used three possible nominal values: LOW, MEDIUM and HIGH. Three partition methods have been used: the equal width method, score type method and a manual method, where the teacher sets the limits of the categories manually, specifying the cutting point of each interval.

**Figure 3.** Pre-processing panel.

**Configuration parameters Panel.** The teacher has to set up the configuration parameters of the algorithm, and the restrictions he/she wants to apply to reduce discovered output information (see Figure 4). Next, the main functionality of each step is described, represented as a red square in the figure:

1. **Number of rules:** this is the only parameter that Predictive Apriori (Scheefer, 2005) needs and is a more intuitive parameter for a non-expert in data mining.
2. **Restricting attributes:** the teacher can specify the attributes that do or do not have to appear in the rule antecedent or consequent, in order to improve the comprehensibility of the rules discovered.
3. **Analysis depth:** In order to restrict the search field, a few parameters related to the depth of analysis have also been added. First, the teacher must select the level to carry out the analysis: course, unit, lesson and others tables such as course-unit, course-



lesson, course-exercise, course-forum, unit-exercise, unit-lesson, lesson-exercise among others. Then, the teacher must select a particular course, unit or lesson in order to do rule mining only with specified data at the specified level. The system also allows interesting relationships among attributes in different tables to be identified, for example if the user selects an analysis at *course-unit* level, *unit-exercises*, etc; the temporary table will contain attributes and transactions of more than one table (see Table 2).

**Table 2.** Temporary tables used in the discovery of association rules.

4. **Restricting items:** an item is a pair attribute=value. This option allows the teacher to specify maximum numbers for antecedent and consequent items in the rule. By default, the application sets a value of 1 for the quantity of items in the consequent, which also improves the comprehensibility of the resulting rules.

**Figure 4.** Parameters configuration panel.

**Rules Repository Panel.** The rules repository (see Figure 5) is the knowledge database that provides the background for a subjective analysis of the rules discovered.

**Figure 5.** Rules repository panel.

Next, the main functionality of each step will be described, represented as a red square in Figure 5:

1. **Get rules set:** Since a specific rule and/or specific recommendation that has been discovered in one course does not necessarily have to be valid or applicable to another

different course, the rules in the repository are classified according to the teacher profile: Topic, Level and Difficulty. Before running the algorithm, the teacher downloads the current knowledge database from the server (pressing *Get rules set from server*), according to his/her course profile. The personalisation of the tuples returned by the server is based on these three filtering parameters, along with the type of course to be analysed. So, the teacher only downloads tuples that match each profile.

2. **Rule Set:** This table shows the tuples downloaded from the remote repository in step 1. The information provided by the system for each tuple is: the rule itself (antecedent and consequent), the problem detected by the rule, the associated recommendation or recommendations for its solution, and a parameter called weighted accuracy (*WAcc*) (García et al., 2009) which is a function of two parameters: 1) the voting of experts and teachers, and 2) the average of predictive accuracy of the rule obtained by teachers when they apply the Predictive Apriori algorithm to their own courses. The rules repository is created on the server side, based on the educational considerations of experts and the experience garnered from other similar e-learning courses. In order to allow this exchange of information (tuples) between the application client and the server, a web service has been implemented to download the most recent version of the repository using standard PMML format. A model of association rules has three main parts: 1) attributes of the model; 2) items; and 3) association rules. Each rule stored in the repository should indicate possible problems to be detected in on-line courses, so it will be necessary to describe each tuple through the fields previously mentioned. In order to include new fields in this model, different extensions are added through the element "Extension" of PMML. The definitive structure of each

association rule inside the PMML file is shown in Table 3.

**Table 3.** Structure of each association rule inside the PMML file.

Next, in Table 4, a fragment of the rules repository used in CIECoF is provided.

Specifically, two tuples related to attributes in the *students\_courses* and *students\_units* tables are shown.

**Table 4.** A fragment of the rules repository in PMML format.

3. **Run button:** Finally, after downloading the rule repository and configuring the application parameters or using default values, the teacher executes the association rule algorithm, pressing this button.

**Results Panel.** Once the professor executes the mining algorithm, the rules discovered are classified according to whether they were foreseeable or not (see Figure 6). If they coincide with some of the rules in the repository, in the event of being foreseeable, it is possible to directly show the professor the other fields of the tuple, corresponding to the rule in question: a) the problem detected, b) the possible solutions to this problem through a group of recommendations. There are two types of recommendations:

- a. Active, if a direct modification in course content or structure is involved. Active recommendations can be linked to: modifications in the formulation of the questions or the practical exercises/tasks assigned to the students; changes in previously assigned parameters such as course duration or the level of lesson difficulty; or the elimination of a resource such as a forum or a chat room.
- b. Passive, if they detect a more general problem and point the teacher towards

more specific recommendations.

**Figure 6.** Classification of discovered rules.

The results panel shows the association rules discovered through the application of the Predictive Apriori algorithm in the following fields: rule, problem, recommendation, score and apply button. The main functionality of each step, represented as a red square in the figure is as follows:

1. **Apply button:** For active recommendations, by clicking the *Apply* button, the teacher will be shown the area of the course that the recommendation refers to (see Figure 7) so that he/she can carry out the modification, change, elimination, etc. Each time a teacher applies an active recommendation, he/she implicitly votes for that tuple.
2. **Resource Dialog:** It shows the specific parameters of the didactic resource pertaining to the problem detected. For example, Figure 7 shows that there is a writing error in the wording of the question (20 cm instead of 2 cm) and so the teacher has corrected it and has also added some more information.

**Figure 7.** Results panel.

### 3.2. Server Application

On the server side, a web application has been implemented to manage the knowledge database or repository. In order to access absolutely all the editing options for the repository, a basic profile was created, which is the profile of the experts in the educational domain.

These experts have permission to introduce new tuples into the rule repository (see Figure 8). The tuples in the repository are classified according to the course profile; therefore, before introducing the rules, the expert must select the course profile that the rule corresponds to.

The next section describes the main functionality of each step, represented as a red square in the figure:

1. **Rule parameters:** the teacher can configure the items in the antecedent of the rule, up to three items in the antecedent of the rule, and one item in the consequent.
2. **Problem detected:** the teacher sets the problem/s this rule could detect
3. **Recommendation:** finally, the teacher enunciates the possible recommendation/s to solve the problem that this rule detects.

**Figure 8.** Panel to insert a rule in the repository.

Both experts and teachers participate in the creation of the knowledge base. Initially the knowledge base was empty and experts proposed tuples. In continuation, there is a commentary on how experts and teachers voted.

On one hand, each expert, using the server application, voted for each tuple in the repository according to the approaches specified in Figure 9.

**Figure 9.** Voting Panel.

Expert evaluation has been divided into two groups of evaluation criteria or approaches:

1. The  $A_1$  block: the three options in this block are related to expert evaluation of the selected tuple with respect to its comprehensibility, suitability and the adjustment of the tuple to the profile.
2. The  $A_2$  block: the three options in this block are related to expert decision regarding the selected tuple with respect to its addition to the repository.

By making  $W_1$ ,  $W_2$  the weights assigned by the system administrator to the two groups of options  $A_1$  and  $A_2$ , the total score of a tuple can be calculated according to:

$$NumVotesExpert = W_1 * \bar{A}_1 + W_2 * \bar{A}_2$$

where  $\bar{A}_1$  and  $\bar{A}_2$  are the average score given by experts to each option in the group. The *NumVotesExpert* values are between 0 and 100 and they are distributed, depending on the vote, in the following way: Very Low option (20 points), Low (40 points), Normal (60 points), High (80 points), and Very High (100 points).

On the other hand, as previously mentioned, teachers vote implicitly; that is, if teachers apply one of the recommendations to their course, they are automatically voting for its applicability to this tuple:

$$NumVotesTeacher = 100 * TeacherVote$$

where *TeacherVote* is a binary variable with values of true (1) or false (0) according to whether the teacher votes for the rule or not. The values of *NumVotesExpert* and *NumVotesTeachers* are used to calculate the score of each rule. Each time that a client application updates the repository, the *WAcc* parameter (García et al., 2009) is recalculated and the tuples in the repository are reordered, taking into account this parameter.

Figure 10 shows the list of tuples classified according to the course profile. The fields in this list are:

1. **Description of the rule:** it shows its links to each tuple in the repository, so that, before voting, the expert can consult the specific parameter of the tuple.
2. **The score:** it shows the *WAcc* parameter of the tuple.
3. **Voting options:** it shows the expert a link to the 'Voting Panel' (Figure 9) if the expert has not voted for the rule; otherwise the expert can modify his/her vote using the hand icon.

**Figure 10.** List of tuples.

### 3.3. Examples of discovered rules

This section concentrates on a description of the meaning and the possible use of several rules regarding our courses that were discovered using CIECoF. The student's usage data used is from an adaptive web-based course based on subjects included in the ECDL (European Computer Driving Licence) executed by 90 students from 3 towns in the Province of Cordoba (aimed at increasing the technological literacy of women in rural areas). The course topics were: basic concepts of information technology, using the computer and managing files, word processing, spreadsheets, databases/filing systems, presentation and drawing, and information network services.

It is important to highlight that a single rule can have several interpretations. Therefore the system will always show those recommendations related to the detection of a possible problem, and it is the teacher him/herself who actually decides what recommendations to use.

Semantically our discovered rules are expressed in the following pattern:

**IF** *Time|Score|Participation* **AND** ... **THEN** *Time|Score|Participation* (*supp. = acc. =*)

Where *Time*, *Score* and *Participation* are thereby generic attributes referring to: the reading time dedicated to the course, units, lessons and exercises (HIGH, MEDIUM and LOW values); information on students' scores in the test and activities questions (HIGH, MEDIUM and LOW values); and lastly, participation refers to how the students have used collaborative resources like the forum and chat (HIGH, MEDIUM and LOW values). Based on the rules discovered, the teacher can decide which of the relationships expressed are desirable or undesirable, and whether or not to apply the recommendation in order to strengthen or weaken the relationship (namely changing or modifying the contents, structure and adaptation of the course, etc.).

Next, we describe some examples of the general patterns found in rules of interest offering

the teacher useful information about how to improve a course. We also describe some of their possible interpretations. It is important to highlight that a single rule can have several interpretations. Therefore the system will always show all the recommendations related to a detected problem, and it is the teacher him/herself who actually decides what recommendations to us.

**Examples of expected rules:**

**IF** *u\_time [5] = HIGH AND u\_attempts [5] = LOW THEN* *u\_final\_score[5] = HIGH* (supp. = 0.81, acc.= 0.72)

**Meaning/Detected Problem:**

If the time used to complete unit 5 is high and the number of attempts to overcome the topic is low, then the final note of the topic is high. This can mean that although the students needed a long time to complete this topic, it was carried out in a few attempts and in the end, the score was high, so possibly the time needed to complete the unit was not estimated correctly.

**Action:**

The duration of this topic was lengthened.

**IF** *u\_assignment\_score [11] = LOW THEN* *u\_final\_score[5] = HIGH* (supp. = 0.65, acc = 0.72)

**Meaning/Detected Problem:**

This rule means that if students got a low score in task 11 that corresponded to unit 5, then the final score obtained in unit 5 was high. This rule can indicate that there could be a problem in this task, for example the wording of this task could be incorrect or ambiguous, giving place to several interpretations.

**Action:**



The wording of the task was modified because it was ambiguous and did not specify all the data needed to complete the task correctly.

**Examples of unexpected rules:**

**IF** *u\_time [1] = HIGH* **THEN** *u\_forum\_read[2] = LOW* (supp. = 0.58, acc.= 0.68)

**Meaning/Detected Problem:**

If the time used in completing unit 1 is high, then the quantity of messages read in the forum of the same unit is low. This was an unexpected rule and it can have several interpretations. On one hand, the students demonstrated self-sufficiency when dedicating more time to the resolution of unit 1 without needing to use the forum associated to that unit, so maybe the forum was not necessary, or, on the other hand, the unit is complicated and the students have had problems in its resolution, although they have not used the forum. In these cases it is recommended to see if other more specific problems have been detected at this unit level.

**Action:**

Other more specific problems and recommendations were consulted at unit level, detecting problems with respect to that forum

**IF** *u\_forum\_read [3] = HIGH* **THEN** *u\_assignment\_score[9] = HIGH* (supp. = 0.78, acc.= 0.67)

**Meaning/Detected Problem:**

This rule means that, if the quantity of messages read in forum 3 belonging to unit 3 is high, then the score of task 9, corresponding to the same unit, is also high. At first sight this relationship can seem obvious or logical, but other similar rules found in other topics led to the discovery that most of the units with some assigned task involved a medium or high participation in the forum .

**Action:**

New tasks were created in those units with no tasks or with a low number of tasks.

**4. Conclusions**

This paper has demonstrated a data mining tool that uses association rule mining and collaborative filtering in order to make recommendations to instructors about how to improve e-learning courses. This tool enables the sharing and scoring of rules discovered by other teachers in similar courses. The use of the tool is described and some examples of the rules discovered are given. The system proposed validates didactic models for e-learning, so that once the teacher establishes the values that he/she considers to be high, medium and low in the use of each didactic course resource, the system will identify anything that contradicts the model as being a problem. This allows the teacher to analyze and to improve initial beliefs based on real interactions between the students and the e-learning course.

Currently, the mining tool has only been used by a group of instructors and experts involved in the development of the tool itself. In fact, the general opinion of these experts and teachers who participated in the experiments has been very good, showing a high level of interest and motivation. Therefore in the future the tool will have to be tested with several groups of external instructors and experts to determine its usability with external users. In our experiments, the same weight was given to the voting of experts as to that of teachers, and so analyzing the relevance of this teacher and expert voting could also be an interesting topic for future research.

**Acknowledgement**

The authors gratefully acknowledge the financial support provided by the Spanish department of Research under TIN2008-06681-C06-03 and P08-TIC-3720 Projects.

## References

- Agrawal, R., H. Mannila, R. Srikant, H. Toivonen and A. Verkamo. (1996). Fast discovery of association rules. *Advances in Knowledge Discovery and Data Mining*, Menlo Park, CA: AAAI Press, 307-328.
- Andrejko, A., Barla, M., Bielikova, M., Tvarozek, M. (2007). User Characteristics Acquisition from Logs with Semantics. In *International Conference on Information System Implementation and Modeling*, Czech Republic, 103-110.
- Avouris, N., Komis, V., Fiotakis, G., Margaritis, M., Voyiatzaki, E. (2005). Why logging of fingertip actions is not enough for analysis of learning activities. In *Workshop on Usage analysis in learning systems*, AIED Conference, Amsterdam, 1-8.
- Baker, R., Yacef, K. (2010). The State of Educational Data Mining in 2009: A Review and Future Visions. *Journal of Educational Data Mining*, 3-17.
- Bari, M. Benzater, B. (2005). Retrieving data from pdf interactive multimedia productions. In *International Conference on Human System Learning*, 321–330.
- Becker, K., Vanzin, M. Ruiz, D. (2005). Ontology-based filtering mechanisms for web usage patterns retrieval. In *6th International Conference on E-Commerce and Web Technologies*, 267–277.
- Bellaachia, A., Vommina, E. (2006). MINEL: A framework for mining e-learning logs. In *Fifth IASTED International Conference on Web-based Education*, Mexico, 259-263.
- Ben-naim, D., Bain, M., Marcus, N. (2009). A User-Driven and Data-Driven Approach for Supporting Teachers in Reflection and Adaptation of Adaptive Tutorials. In *International Conference on Educational Data Mining*, Cordoba, Spain, 21-30.
- Castro, F., Vellido, A., Nebot, A. Mugica, F. (2007). Applying Data Mining Techniques to e-Learning Problems. In: Jain, L.C., Tedman, R. and Tedman, D. (eds.) *Evolution of*

Teaching and Learning Paradigms in Intelligent Environment. *Studies in Computational Intelligence*, 62, Springer-Verlag, 183-221.

Clementine (2010). <<http://www.spss.com/clementine/>>.

Damez, M., Marsala, C., Dang, T. Bouchon-meunier, B. (2005). Fuzzy decision tree for user modeling from human-computer interactions. In *International Conference on Human System Learning*, 287–302.

DBMiner (2010). <<http://www.dbminer.com>>

Gaudioso, E., Montero, M., Talavera, L., Hernandez-del-Olmo, F. (2009). Supporting teachers in collaborative student modeling: a framework and an implementation. In *Expert System with Applications*, 36, 2260-2265.

Garcia, E., Romero, C., Ventura, S., Castro, C. (2009a). An architecture for making recommendations to courseware authors using association rule mining and collaborative filtering. *User Modeling and User-Adapted Interaction: The Journal of Personalization Research*, 19, 99–132.

Garcia, E., Romero, C., Ventura, S., Castro, C. (2009b). Collaborative Data Mining Tool for Education. In *International Conference on Educational Data Mining*, Cordoba, Spain, 299-306.

Hershkovitz, A., Nachmias, R. (2008). Developing a log-based motivation measuring tool. In *1st International Conference on Educational Data Mining*, Montreal, 226-233.

Hurley, T., Weibelzahl, S. (2007). Using MotSaRT to support on-line teachers in student motivation. In *European Conference on Technology-Enhanced Learning*, Crete, Greece, 101-111.

- Hwang, G.J., Tsai, P.S., Tsai, C.C., Tseng, J.C.R. (2008). A novel approach for assisting teachers in analyzing student web-searching behaviors. *Journal of Computer & Education*, 51, 926-938.
- Jovanovic, J., Gasevic, D., Brooks, C., Devedzic, V., Hatala, M. (2007). LOCO-Analyst: A tool for raising teacher's awareness in online learning environments. In European Conference on Technology-Enhanced Learning, Crete, 112-126.
- Jong, B.S., Chan, T.Y., Wu, Y.L. (2007). Learning log explorer in e-learning diagnosis. *Journal of IEEE Transactions on Education*, 50,3, 216-228.
- Juan, A., Daradoumis, T., Faulin, J., Xhafa, F. (2009). SAMOS: a model for monitoring students' and groups' activities in collaborative e-learning. *International Journal of Learning Technology*, 4,1-2, 53-72.
- Keel. (2010). <<http://www.keel.es/>>.
- Koedinger, K., Cunningham, K., Skogsholm A., Leber, B. (2008). An open repository and analysis tools for fine-grained, longitudinal learner data. In 1st International Conference on Educational Data Mining, Montreal, 157-166.
- Kosba, E.M., Dimitrova, V., Boyle, R. (2005). Using Student and Group Models to Support Teachers in Web-Based Distance Education. In International Conference on User Modeling, Edinburgh, 124-133.
- Lau, R., Chung, A., Song, D., Huang, Q. (2007). Towards fuzzy domain ontology based concept map generation for e-learning. In International Conference on Web-based Learning, Edinburgh, 90-101.
- Mazza, R., Milani, C. (2004). GISMO: a Graphical Interactive Student Monitoring Tool for Course Management Systems, In International Conference on Technology Enhanced Learning, Milan, 1-8.

- Merceron, A., Yacef, K. (2005). Educational Data Mining: a Case Study. In International Conference on Artificial Intelligence in Education, Amsterdam, The Netherlands, 1-8.
- Miner (2010). <<http://www-306.ibm.com/software/data/iminer/>>.
- Mostow, J., Beck, J., Cen, H., Cuneo, A., Gouvea, E., Heiner, C. (2005). An educational data mining tool to browse tutor-student interactions: Time will tell! In Proceedings of the Workshop on Educational Data Mining, 15–22.
- Mostow, J., Beck, J. (2006). Some useful tactics to modify, map and mine data from intelligent tutors. *Journal Natural Language Engineering*, 12 ,2, 195-208.
- Nagata, R. Takeda, K., Suda, K., Kakegawa, J., Morihiro, K. (2009). Edu-mining for book recommendation for pupils. In International Conference on Educational Data Mining, Cordoba, Spain, 91-100.
- Psaromiligkos, Y., Orfanidou, M. Kytagiias, C., Zafiri, E. (2009). Mining log data for the analysis of learners' behaviour in web-based learning management systems. *Journal of Operational Research*, 1-14.
- RMiner (2010). <<http://www.rapidminer.com>>.
- Romero, C., Ventura, S., De Bra, P. (2004). Knowledge discovery with genetic programming for providing feedback to courseware author. *User Modeling and User-Adapted Interaction: The Journal of Personalization Research*, 14, 5, 425–464.
- Romero, C., Ventura, S.. (2007). Educational Data Mining: a Survey from 1995 to 2005. *Journal of Expert Systems with Applications*, 1,33, 135-146.
- Romero, C., Ventura, S., Salcines, E. (2008a). Data mining in course management systems: Moodle case study and tutorial. *Journal of Computer & Education*, 51(1), 368-384.

- Romero, C., Gutierrez, S., Freire, M., Ventura, S. (2008b). Mining and visualizing visited trails in web-based educational systems. In International Conference on Educational Data Mining, Montreal, Canada, 182-185.
- Romero, C., Ventura, S., Zafra, A., de Bra, P. (2010). Applying Web Usage Mining for Personalizing Hyperlinks in Web-based Adaptive Educational Systems. *Journal of Computers & Education*, 53, 828-840.
- Retalis, S., Papasalouros, A., Psaromilogkos, Y., Siscos, S., Kargidis, T. (2006). Towards networked learning analytics – A concept and a tool. In Fifth International Conference on Networked Learning, 1-8.
- Scheffer, T. (2005) Finding Association Rules That Trade Support Optimally against Confidence. *Intelligent Data Analysis*, 9(4), 381-395.
- Selmoune, N., Alimazighi, Z. (2008). A decisional tool for quality improvement in higher education. In International Conference on Information and Communication Technologies, Damascus, Syria, 1-6.
- Shen, R., Yang, F., Han, P. (2002). Data analysis center based on e-learning platform. In Workshop The Internet Challenge: Technology and Applications, Berlin, Germany, 19-28.
- Shen, R., Han, P., Yang, F., Yang, Q., Huang, J. (2003). Data mining and case-based reasoning for distance learning. *Journal of Distance Education Technologies*, 1, 3, 46-58.
- Silva, D., Vieira, M. (2002). Using data warehouse and data mining resources for ongoing assessment in distance learning. In IEEE International Conference on Advanced Learning Technologies. Kazan, Russia, 40–45.
- Singley, M.K., Lam, R.B. (2005). The Classroom Sentinel: supporting data-driven decision-making in the classroom. In 13th World Wide Web Conference, Chiba, Japan, 315-322.

- Tane, J., Schmitz, C. Stumme, G. (2004). Semantic resource management for the web: An elearning application. In Proceedings of the WWW Conference. New York, USA, 1-10.
- Ventura, S., Romero, C., Hervás, C. (2008). Analyzing rule evaluation measures with educational datasets: a framework to help the teacher. In International Conference on Educational Data Mining, Montreal, Canada, 177-181.
- Vialardi, C., Bravo, J., Ortigosa, A. (2008). Improving AEH courses through log analysis. In *Journal of Universal Computers Science*, 14 (17), 2777-1798.
- Weka (2010). <<http://www.cs.waikato.ac.nz/ml/weka/>>.
- Witten, I. H., Frank, E. (2005). *Data Mining. Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan y Kaufmann.
- Yoo, J., Yoo, S., Lance, C., Hankins, J. (2006). Student progress monitoring tool using treeview. In Technical Symposium on Computer Science Education, ACM-SIGCESE, 373 - 377.
- Zaïane, o., Luo, J. (2001). Web usage mining for a better web-based learning environment. In Proceedings of Conference on Advanced Technology for Education. Banff, Alberta, 60–64.
- Zheng, Z., R. Kohavi and L. Mason. (2001). Real world performance of association rule. Sixth ACM SIGKDD International Conference on Knowledge Discovery & Data Mining 2(2), 86-98.
- Zinn, C., Scheuer, O. (2006). Getting to know your students in distance-learning contexts. In 1st European Conference on Technology Enhanced Learning, 437-451.



Table 1. List of EDM tools.

<b>Tool</b>	<b>Objective</b>	<b>Reference</b>
WUM tool	To extract patterns useful for evaluating on-line courses.	(Zaïane and Luo, 2001)
MultiStar	To aid in the assessment of distance learning.	(Silva and Vieira, 2002)
Data Analysis Center	To analyze students' patterns and organize web-based contents efficiently.	(Shen et al., 2002)
Assistance tool	To provide a tool for students to look for the materials they need.	(Shen et al., 2003)
EPRules	To discover prediction rules to provide feedback for courseware authors.	(Romero et al., 2004)
KAON	To find and organize the resources available on the web in a decentralized way.	(Tane et al., 2004)
GISMO/CourseVis	To visualize what is happening in distance learning classes.	(Mazza and Milani, 2004)
TADA-ED	To help teachers to discover relevant patterns in students' online exercises.	(Merceron and Yacef, 2005)
O3R	To retrieve and interpret sequential navigation patterns.	(Becker et al., 2005)
Synergo/ColAT	To analyze and produce interpretative views of learning activities.	(Avouris et al., 2005)
LISTEN tool	To explore huge student-tutor interaction logs.	(Mostow et al., 2005)
TAFPA	To provide helpful analysis of the cognitive process.	(Damez et al., 2005)
iPDF_Analyzer	To help predict interactive properties in the multimedia presentations produced by students.	(Bari and Benzater, 2005)
Classroom Sentinel	To detect patterns and deliver alerts to the teacher.	(Singley and Lam, 2005)
Teacher ADVisor	To generate advice for course instructors.	(Kosba et al., 2005)
Teacher Tool	To analyze and visualize usage-	(Zinn and Scheuer, 2006)

		tracking data.	
CoSyLMSAnalytics		To evaluate the learner's progress and produce evaluation reports.	(Retalis et al., 2006)
Monitoring tool		To trace deficiencies in student comprehension back to individual concepts.	(Yoo et al., 2006)
MINEL		To analyze the navigational behavior and the performance of the learner.	(Bellaachia and Vommina, 2006)
Simulog		To validate the evaluation of adaptive systems by user profile simulation.	(Bravo and Ortigosa, 2006)
LOCO-Analyst		To provide teachers with feedback on the learning process.	(Jovanovic et al., 2007)
LogAnalyzer tool		To estimate user characteristics from user logs through semantics.	(Andrejko et al., 2007)
Learning Explorer	log	To diagnose student learning behaviors.	(Jong et al., 2007)
MotSaRT		To help the on-line teacher with student motivation.	(Hurley and Weibelzahl, 2007)
Measuring tool		To measure the motivation of online learners.	(Hershkovitz and Nachmias, 2008)
DataShop		To store and analyze click-stream data, fine-grained longitudinal data generated by educational systems.	(Koedinger et al., 2008)
Visualizing trails		To mine and visualize the trails visited in web-based educational systems.	(Romero et al., 2008)
Measures Tool		To analyze rule evaluation measures.	(Ventura et al., 2008)
Solution Trace Graph		To visualize and analyze student interactions.	(Ben-Naim et al., 2008)
Decisional Tool		To discover factors contributing to students' success and failure rates.	(Selmoune and Alimazighi, 2008)
Concept Map Generation Tool	Map	To automatically construct concept maps for e-learning.	(Lau et al., 2007)
Author Assistant		To assist in the evaluation of adaptive courses.	(Vialardi et al., 2008)
Meta-Analyzer		To analyze student learning behavior in the use of search	(Hwang et al., 2008)

		engines.	
CIECoF		To make recommendations to courseware authors about how to improve courses.	(Garcia et al., 2009b)
SAMOS		Student activity monitoring using overview spreadsheets.	(Juan et al., 2009)
PDinamet		To support teachers in collaborative student modeling.	(Gaudioso et al., 2009)
E-learning analysis	data	To analyze learner behavior in learning management systems.	(Psaromiligkos et al., 2009)
Refinement		To confirm or reject hypotheses concerning the best way to use adaptive tutorials.	(Ben-Naim et al., 2009)
Suggestion Tool			
Edu-mining		To recommend books to pupils.	(Nagata et al., 2009)
AHA! Mining Tool		To recommend the best links for a student to visit next.	(Romero et al., 2010)

---

**Table 2.** Temporary tables used in the discovery of association rules.

<b>Table</b>	<b>Attributes</b>
<i>Students - Courses</i>	<i>c_score, c_time</i>
<i>Students - Units</i>	<i>u_time, u_initial_score, u_final_score, u_attempts</i>
<i>Students - Lessons</i>	<i>l_visited, l_finished, l_time</i>
<i>Students - Exercises</i>	<i>e_score, e_time</i>
<i>Students - Forums</i>	<i>forum_read, forum_post</i>
<i>Students - Quizzes</i>	<i>quiz_time, quiz_score</i>
<i>Students - Tasks</i>	<i>assignment_score</i>
<i>Students - Chats</i>	<i>chat_messages</i>
<i>Students -&lt;Table1&gt;-...&lt;TableN&gt;</i>	<i>&lt;attributes table1&gt;, ..., &lt;attributes table N&gt;</i>

**Table 3.** Structure of each association rule inside the PMML file.

```
<AssociationRule support=" " confidence=" " antecedent=" " consequent=" " >  
  <Extension name="ProblemDetected" value=" " />  
  <Extension name="Recommendation" value=" " />  
  <Extension name="WAcc" value=" " />  
</AssociationRule>
```

Table 4. A fragment of the rules repository in PMML format.

```

<?xml version="1.0" encoding="utf-8"?>
<PMML          version="3.1"          xmlns="http://www.dmg.org/PMML-3_1"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Header copyright="CIECoF">
    <Application name="Rule repository" version="1.0" />
    <Annotation> Example of PMML file </Annotation>
  </Header>
  <DataDictionary numberOfFields="2">
    </DataField>
    <DataField name="c_time" optype="categorical" dataType="string">
      <Value value="LOW" property="valid" />
      <Value value="MEDDLE" property="valid" />
      <Value value="HIGH" property="valid" />
    </DataField>
    <DataField name="c_score" optype="categorical" dataType="string">
      <Value value="LOW" property="valid" />
      <Value value="MIDDLE" property="valid" />
      <Value value="HIGH" property="valid" />
    </DataField>
  </DataDictionary>
  ...
  <AssociationModel      functionName="associationRules"      algorithmName="PredictiveApriori"
numberOfRules="15">
    <MiningSchema>
      <MiningField name="c_time" />
      <MiningField name="c_score" />
      <MiningField name="u_time" />
      <MiningField name="u_initial_score" />
      <MiningField name="u_final_score" />
      <MiningField name="u_attempts" />
    </MiningSchema>
    <Item id="1" value="u_time=HIGH" />
    <Item id="2" value="u_attempts=LOW" />
    <Item id="3" value="u_initial_score=HIGH" />
    <Item id="4" value="u_final_score=MIDDLE" />
    <Item id="5" value="c_score=HIGH" />
    <Itemset id="1" numberOfItems="1">
      <ItemRef itemRef="3" />
    </Itemset>
    <Itemset id="2" numberOfItems="1">
      <ItemRef itemRef="5" />
    </Itemset>
    <Itemset id="3" numberOfItems="2">
      <ItemRef itemRef="1" />
      <ItemRef itemRef="2" />
    </Itemset>
    <Itemset id="4" numberOfItems="1">
      <ItemRef itemRef="4" />
    </Itemset>
    <AssociationRule> antecedent="2" consequent="1"
      <Extension name="ProblemDetected" value=" If the score in the chapter is medium/normal and
the final socre in the course is high, then you can set that there are problems in the chapter." />
      <Extension name="Recommendation" value="To see/check other recommendations about the
chapter." />
      <Extension name="WAcc" value="0.74" />
    </AssociationRule>
  </AssociationModel>
  ...
</PMML>

```

Figure

[Click here to download high resolution image](#)

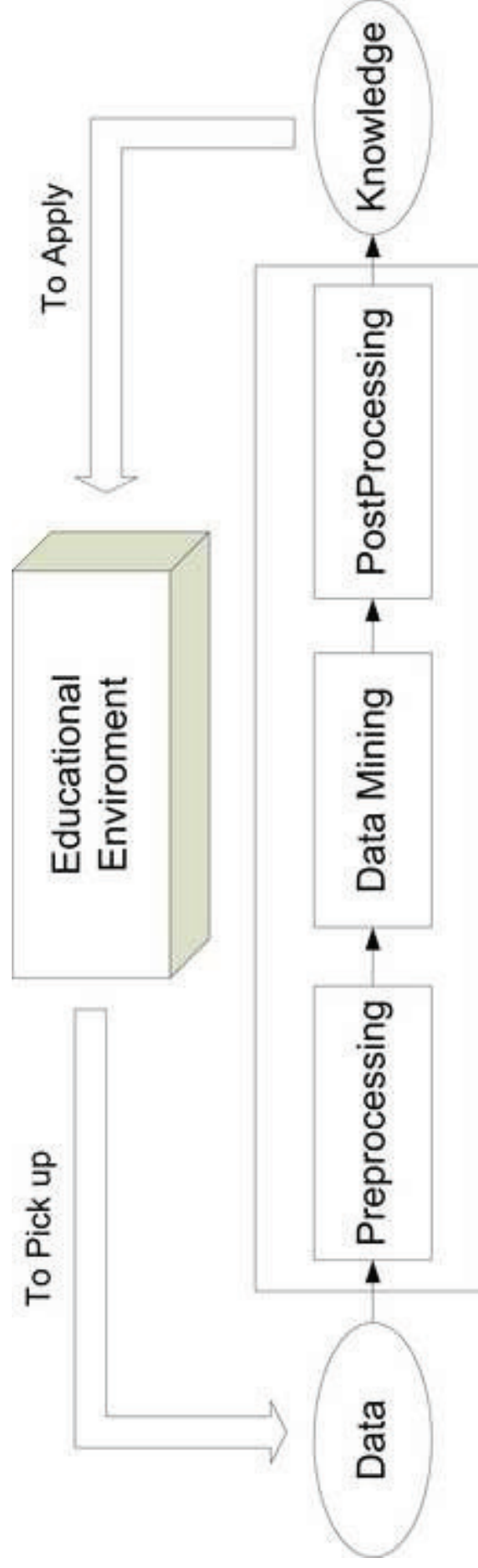
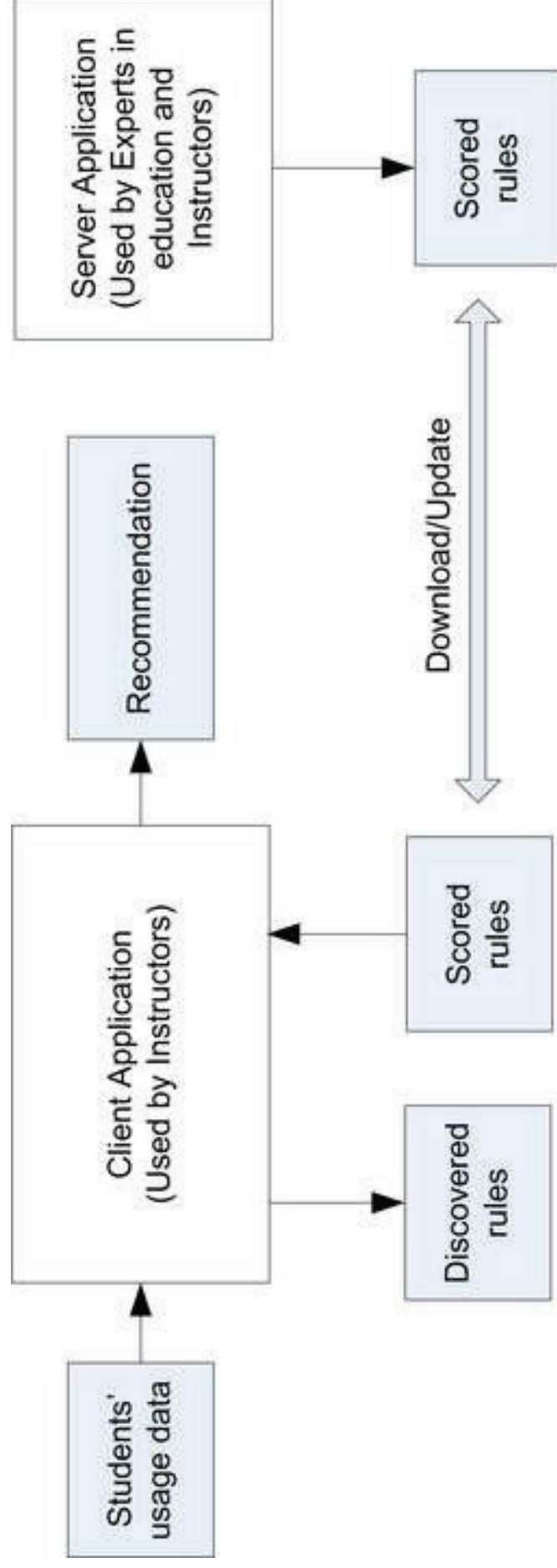


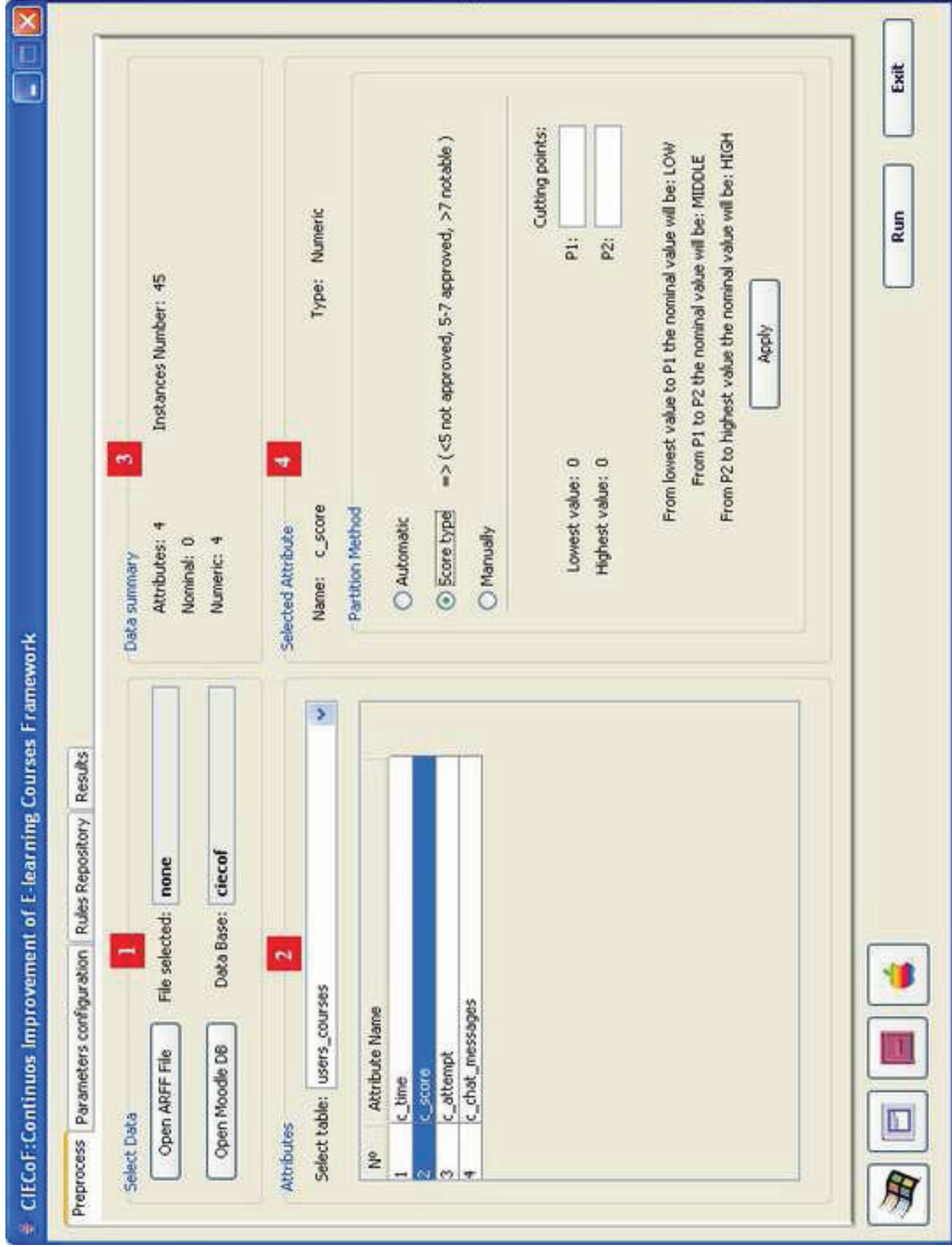
Figure  
[Click here to download high resolution image](#)





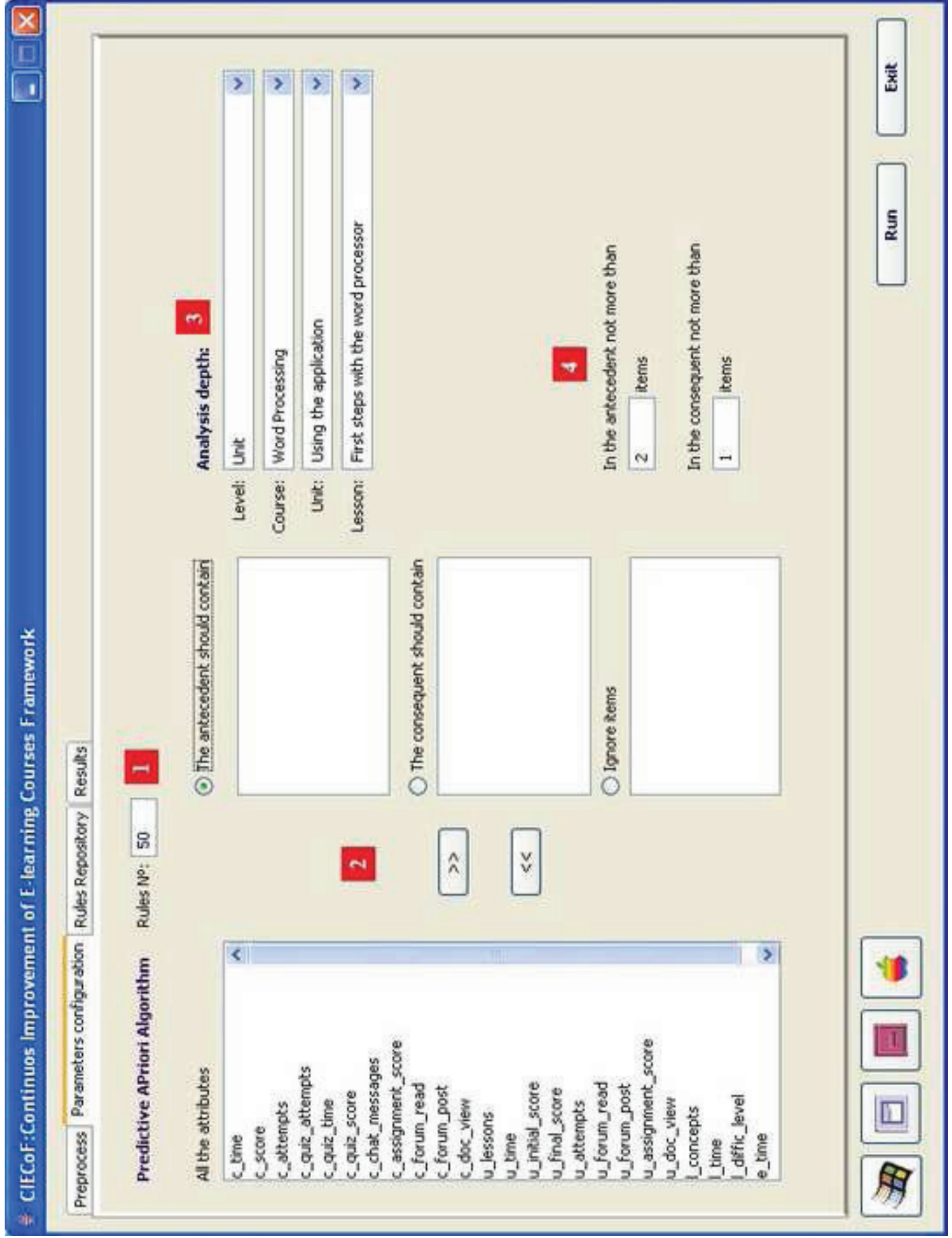
Figure

Click here to download high resolution image



Figure

Click here to download high resolution image

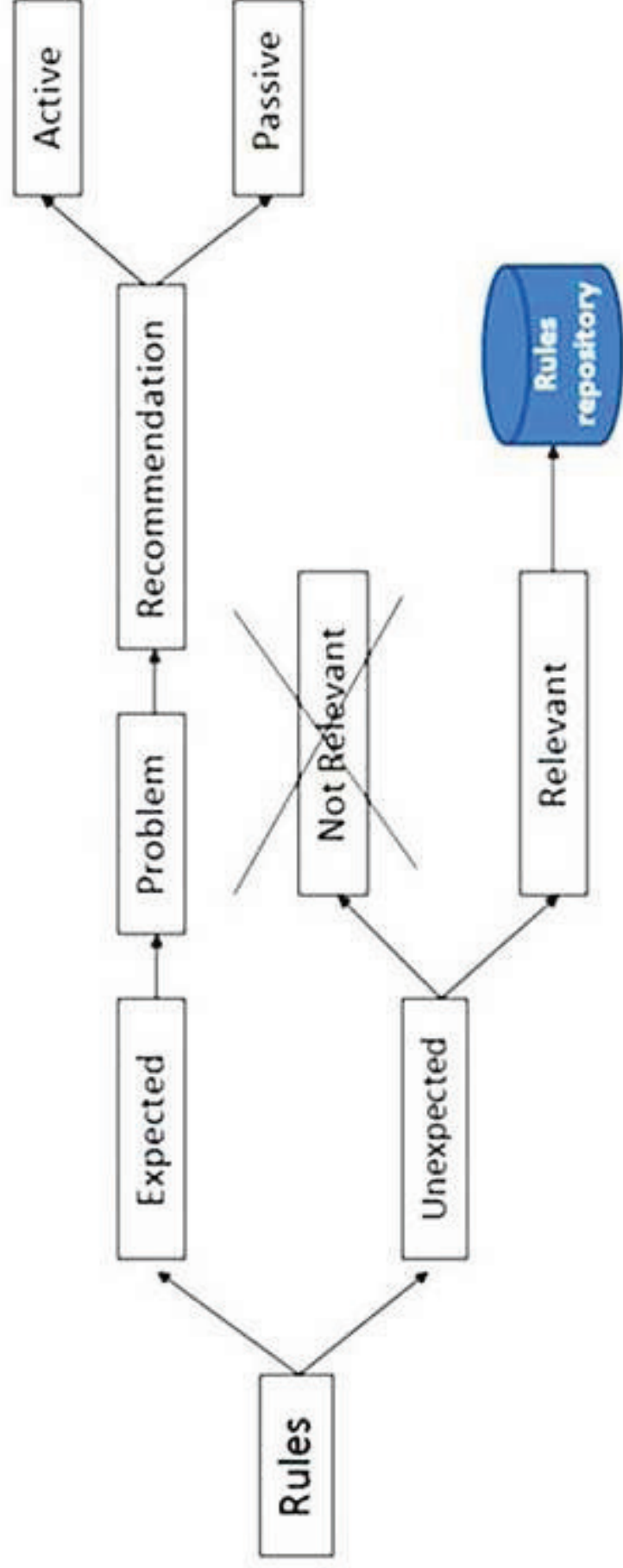


Figure

Click here to download high resolution image



Figure  
[Click here to download high resolution image](#)



Figure

Click here to download high resolution image

CECoF: Continuous Improvement of E-learning Courses Framework

Preprocess Parameters configuration Rules Repository Results

Rule	Problem detected	Recommendation	Score
e_time[2]=HIGH => e_score[2]=LOW acc=0.989	1. The exercise wording could be incorrect or ambiguous. 2. The exercise hyperlink could be broken.	1. Verify the exercise hyperlink. 2. Modify the exercise wording. 3. Eliminate the exercise	50
u_final_score[1]=LOW => c_score=MEDIUM acc=0.975		ns at unit level	50
u_doc_view[13]=LOW => u_final_score[2]=MEDIUM acc=0.907		ok of the doc is link	40
u_final_score[2]=MEDIUM => c_score=MEDIUM acc=0.879			-
e_time[25]=HIGH => e_score[25]=LOW acc=0.822		2. Modify the exercise	50
u_forum_post[2]=LOW => u_final_score[1]=HIGH acc=0.825		2. Eliminate the	50
u_time[1]=HIGH => u_time[3]=LOW acc=0.813			-
u_initial_score[3]=HIGH => u_final_score[2]=LOW acc=0.780		estions	50
L_concepts=LOW, L_diffc_level=LOW => L_time=HIGH acc=0.752		e lesson difficulty	40
u_assignment_score[3]=LOW => u_final_score[3]=HIGH acc=0.722			50

**2** **Modify or delete exercise**

Current exercise wording:  
Change the margin of the document to 20 cm.

New exercise wording:  
Change the margin of the document to 2 cm but using the options of the main menu.

Verify hyperlink Delete exercise Modify Cancel

Page: 1 of 2 Previous Next

Run Exit

Figure  
Click here to download high resolution image

The screenshot shows a Microsoft Internet Explorer browser window displaying the CIECoF (Continuous Improvement of E-learning Courses Framework) interface. The browser's address bar shows the URL: `http://pvirtual.uco.es/CiecofWS/insertaregla.html`. The page title is "Continuos Improvement of E-learning Courses Framework" and the subtitle is "Repositorio de Reglas Colaborativo". The interface is for "COURSE TYPE: Computer Science - University - Basic".

The main area contains a rule editor with the following sections:

- ANTECEDENT 1**: `e_score` = BAJO
- ANTECEDENT 2**: [ ] = [ ]
- ANTECEDENT 3**: [ ] = [ ]
- CONSEQUENT**: `c_score` = ALTO

Below the rule editor is an "Insert Rule" button. To the right, there are two informational boxes:

- Problem detected 2**: Problemas con el ejercicio. El enunciado puede ser incorrecto o ambiguo.
- Recomendation 3**: Modificar el enunciado o eliminar el ejercicio.

At the bottom right, the user is identified as "User: Carlos de Castro Lozano" and there is a ">> Exit" link.

## Tuple evaluation

Rule:  $e\_time = HIGH \Rightarrow e\_score = LOW$

Rule profile: COMPUTER SCIENCE, UNIVERSITY, BASIC

Problem detected: The exercise wording could be incorrect or ambiguous.  
The exercise hyperlink could be broken.

Recommendations: 1. Verify the exercise hyperlink  
2. Modify the exercise wording  
3. Eliminate the exercise

### Expert evaluation

1	Evaluation criterions $A_1$	Very Low	Low	Normal	High	Very High
	1. The rule comprehensibility is	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
	2. The rule suitability is	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
	3. The adjusting of the rule to the selected profile in the knowledge database is	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

### Expert decision

2	Evaluation criterions $A_2$	Very Low	Low	Normal	High	Very High
	1. My recommendation about to add this rule to the repository is	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
	2. My confidence in this decision is	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
	3. My experience as expert in this rule profile is	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Vote for the rule

Figure  
Click here to download high resolution image

CIECoF - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

Dirección <http://pvirtual.uco.es/CiecofWS/reglas.jsp?numpagina=1>

Vinculos >>

UNIVERSIDAD  
CORDOBA

Continuos Improvement of E-learning Courses Framework  
(CIECoF)

COURSE TYPE: Computer Science - University - Basic

1 2 3

Description of the Rule	Score	Voting
<u>c_score=HIGH=&gt;u_final_score=LOW</u>	50	<a href="#">Vote</a> <input checked="" type="checkbox"/>
<u>e_time=HIGH=&gt;e_score=LOW</u>	50	<a href="#">Vote</a> <input checked="" type="checkbox"/>
<u>forum_read=LOW forum_post=LOW=&gt;u_final_score=HIGH</u>	50	<a href="#">Vote</a> <input checked="" type="checkbox"/>
<u>e_score=LOW=&gt;u_final_score=HIGH</u>	50	<a href="#">Vote</a> <input type="checkbox"/>
<u>e_score=LOW=&gt;e_time=HIGH</u>	50	<a href="#">Vote</a> <input checked="" type="checkbox"/>
<u>forum_read=LOW forum_post=LOW=&gt;c_score=HIGH</u>	50	<a href="#">Vote</a> <input type="checkbox"/>
<u>score_assignment=LOW=&gt;c_score=HIGH</u>	45	<a href="#">Vote</a> <input type="checkbox"/>
<u>score_assignment=LOW=&gt;u_final_score=HIGH</u>	45	<a href="#">Vote</a> <input type="checkbox"/>
<u>score_quiz=LOW=&gt;u_final_score=HIGH</u>	45	<a href="#">Vote</a> <input type="checkbox"/>
<u>u_initial_score=HIGH u_final_score=LOW=&gt;c_score=HIGH</u>	45	<a href="#">Vote</a> <input checked="" type="checkbox"/>
<u>score_quiz=LOW=&gt;c_score=HIGH</u>	40	<a href="#">Vote</a> <input type="checkbox"/>
<u>u_time=HIGH u_attempts=LOW=&gt;u_final_score=HIGH</u>	40	<a href="#">Vote</a> <input type="checkbox"/>
<u>l_concepts=LOW l_diffic_level=LOW=&gt;l_time=HIGH</u>	40	<a href="#">Vote</a> <input type="checkbox"/>
<u>l_concepts=LOW l_diffic_level=HIGH=&gt;l_time=LOW</u>	40	<a href="#">Vote</a> <input type="checkbox"/>
<u>c_chat_messages=HIGH=&gt;c_score=LOW</u>	35	<a href="#">Vote</a> <input checked="" type="checkbox"/>
<u>l_concepts=LOW l_diffic_level=LOW=&gt;l_time=HIGH</u>	30	<a href="#">Vote</a> <input checked="" type="checkbox"/>

User: Enrique Garcia Salcines

Page 1 of 2

>> Exit