Clustering for improving Educational Process Mining

Alejandro Bogarín, Cristóbal Romero
Department of Computer Science
University of Cordoba, Spain
i02bovea@uco.es, cromero@uco.es

Rebeca Cerezo, Miguel Sánchez-Santillán
Department of Psychology
University of Oviedo, Spain
cerezorebeca@uniovi.es,
melsanchezsantillan@gmail.com

ABSTRACT
In this paper, we propose to use clustering to improve educational process mining. We want to improve both the performance and comprehensibility of the models obtained. We have used data from 84 undergraduate students who followed an online course using Moodle 2.0. We propose to group students firstly starting from data about Moodle's usage summary and/or the students' final marks in the course. Then, we propose to use data from Moodle's logs about each cluster/group of students separately in order to be able to obtain more specific and accurate models of students' behaviour. The results show that the fitness of the specific models is greater than the general model obtained using all the data, and the comprehensibility of the models can be also improved in some cases.

Categories and Subject Descriptors
H.2.8 [Database Applications]: Data Mining; K.3.1 [Computer Uses in Education]: Computer-assisted instruction (CAI), Computer-managed instruction (CMI);

General Terms
Algorithms, performance, experimentation.

Keywords
Clustering, process mining, educational data mining, learning analytics.

1. INTRODUCTION
Educational Data Mining (EDM) and Learning Analytics and Knowledge (LAK) [10] study data and analytics in education, teaching, and learning, suggesting educational priorities and undertaking high-quality research into the models, methods, technologies, and impact of analytics. One of the current promising techniques in EDM and LAK is Educational Process Mining (EPM). The basic idea of process mining is to extract knowledge from event logs recorded by an information system. EPM [12] aims to (i) construct complete and compact educational process models that are able to reproduce all observed behaviour, (ii) to check whether the modeled behaviour matches the observed behaviour, and (iii) to project information extracted from the logs onto the model, to make unexpressed knowledge explicit and to facilitate better understanding of the process.

LAK '14, March 24 - 28 2014, Indianapolis, IN, USA

http://dx.doi.org/10.1145/2567574.2567604

The results of EPM can be used to get a better understanding of the underlying educational processes, to generate recommendations and advice to students, to provide feedback to either students, teachers or/and researchers, to detect learning difficulties early, to help students with specific learning disabilities, to improve management of learning objects, etc.; but crucially, helping solve the difficulties that students of different ages show when learn in highly cognitively and metacognitively demanding learning environments like hypermedia or Computer Based Learning Environments [3]. However, we have found two problems when using EPM: 1) the model obtained cannot fit well to the general students' behaviour and 2) the model obtained can be too large and complex for use or analysis by an instructor. In order to resolve these problems, we propose to use clustering to improve both the fitness and comprehensibility of the obtained models by EPM.

This paper is organized as follow. Section 2 is a background about related works. Section 3 describes the proposed approach. Section 4 describes the datasets used. Section 5 describes the experiments and finally, section 6 shows the conclusion and future works.

2. BACKGROUND
Most of the traditional data mining techniques focus on data dependencies or simple patterns and do not focus on the process as a whole and do not provide visual representation of the complete educational process ready to be analyzed [6]. However, EPM techniques aim to extract process-related knowledge from event logs recorded by an information system [13]. In fact, nowadays there are an increasing number of examples about the applicability of process mining in education.

Process mining has been used to extract knowledge from a particular type of an educational information system, considering (oversimplified) educational processes reflecting student behaviour only in terms of their examination scores [13]. Several process mining techniques such as Petri nets, Heuristic and Fuzzy miner have been used to analyse assessment data from recently organized online multiple choice tests [6]. Petri nets analysis has been used to find learning paths or to optimise the path that a student must follow in order to reach a degree or a qualification [4]. Bottleneck mining and petri net simulation have been used to detect discrepancies between the flows prescribed in a student's registration model and the actual process instances [2]. Heuristic Mining has been used to analyze students' writing activities to improve not only the quality of the written document but more importantly the writing skills of those involved [11]. Heuristic Mining has been also used to investigate the processes in students' registration of Thailand's universities for adaptive process simplification in education [1]. Heuristic mining has been

also used to analyse learning paths in LMS and track learning behaviour in relation to respective learning styles, which must be identified in advance. Finally, we want to notice that we haven't found any work that uses clustering techniques together with EPM.

3. PROPOSED APPROACH

In this paper, we proposed an approach that uses clustering for improving educational process mining (see Figure 1).

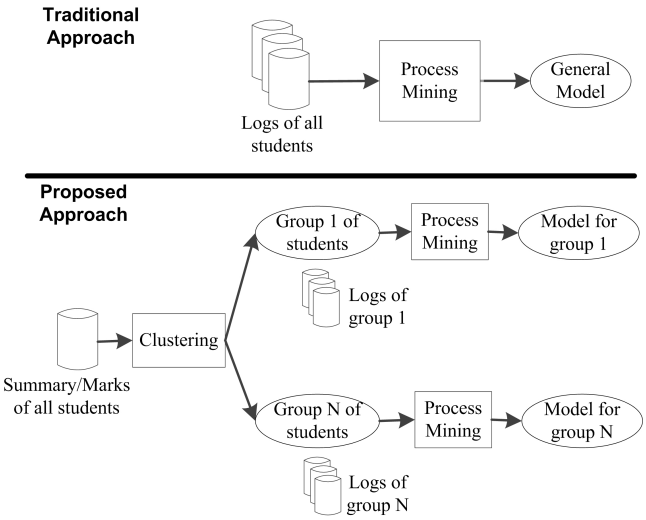

Figure 1. Proposed approach versus the traditional approach.

The traditional approach uses all event log data to reveal a process model of student's behaviour.

On the other hand, this proposed approach firstly applies clustering in order to group students with similar marks or characteristics. It then applies process mining to discover more specific models of student behaviour. We propose two different clustering/grouping approaches:

- **Manual**: To group students directly using only the students' marks obtained in the final exam of the course.
- **Automatic**: To group students using a clustering algorithm over the student's interaction with the Moodle's course.

4. DESCRIPTION OF THE DATA USED

The datasets used in this research were gathered from a Moodle 2.0 course used by 84 undergraduate university students taking the Psychology degree in a university in the north of Spain. The study was conducted during two different semesters in the years 2011-2012 and 2012-2013. The experiment took the form of an assignment in the curriculum of a 3rd year compulsory subject. Students were asked to participate in an eLearning/training programme about "learning to learn", related to the subject's topic, completed entirely out of teaching hours. The programme was made up of 11 different units that were sent to the students on a weekly basis, but each of them was able to work on it during a 15 day period. Each unit was composed of three different types of contents:

- **Declarative knowledge level**: Theoretical contents, description, information, and how-to put the "learn to learn" strategy or strategies of the week into practice.
- **Procedural knowledge level**: Practical tasks where the students had to put their declarative knowledge into practice.
- **Conditional knowledge level**: Discussion forums where the students had to discuss about how they had or would use the strategy or strategies of the week in different contexts.

Students get an extra point in their final subject grade if they complete at least 80% of the assignments. Compulsory assignments for each unit were to solve the practical task and to post at least one comment on each unit forum. Suggested assignments for each unit were to understand the theoretical contents, put them in practice in the task, and share their experience about the week's topic in the forum. Based on the data obtained after students worked on the entire programme, we used three different sources of information.

On the one hand, we used a summary about the interaction of each student in Moodle calculated from Moodle's log and different tables of the database (Table 1).

Table 1. Variables of a student in our Moodle summary file.

Name	Description	Extraction Method under Moodle nomenclature
Time Theory	Total time spent on theoretical contents	Time between viewing resource and next different action
Time Tasks	Total time spent on practical tasks	Time between viewing quiz/ attempting quiz/ continuing quiz attempt/ closing quiz attempt and next different action
Time Forums	Total time spent reviewing forums	Time between viewing forum and the next different action
Days Theory	How long students wait to check the content	Date resource was viewed since content became available on Moodle (in days)
Days Tasks	How long students wait to check the task	Date task was viewed since task became available on Moodle (in days)
Days "hand in"	Taking time in hand in the task	Date quiz attempt closed since task became available on Moodle (in days)
Number of Words in forums	Number of words in forum posts	Extracting number of words added to forum discussion OR forum replay words added
Number of sentences in forums	Number of sentences in forum posts	Extracting number of sentences added to forum discussion OR forum replay

		sentences added

We saved/converted all this information into an .ARFF (Attribute-Relation File Format) file to be able to apply clustering algorithms provided by Weka [16].

On the other hand, we also used the log file provided by Moodle directly (see Table 2).

Table 2. Variables of a Modle log files.

Attribute	Description
Course	The name of the course
IP Address	The IP of the device used to access
Time	The date they accessed it
Full Name	The name of the student
Action	The action that student has done
Information	More information about the action

We preprocessed this log file (see Table 2) thus: we only used the last four variables, that is, we did not use the name of the course (it is the same for all records) or the IP address (it is irrelevant for our purposes). We turned the students' names into IDs (Identifiers) to maintain their privacy. We filtered the possible actions in our log file. So, from 39 actions that Moodle stored in our log file, we only used the next 20 actions that are related to the students activities in the course: upload assignment, view assignment, view course, view folder, add forum discussion, add forum post, update forum post, view forum discussion, view forum, view page, submit questionnaire, view questionnaire, attempt quiz, close quiz attempt, continue quiz attempt, review quiz, view quiz, view quiz summary, view resource, and view url. We then transformed this log file into a MXML (Minimal XML) file using the proMimport framework in order to can use ProM [14] for later process mining.

Finally, we used the students' final marks, or the scores they obtained in the final exam at the end of the course. This is a file provided by the instructor that contains both each student's ID and his or her final mark (a numeric value on a 10-point scale). We turned this continuous value into a categorical value using the traditional academic grading in Spain: 0-4.9: fail and 5-10: pass.

5. EXPERIMENTS

We carried out several experiments to test our proposal. In the first, we used all the log data about the 84 students. In the second, we divided firstly the original log file into two datasets: one that contains the 68 students who passed and other with the 16 students who failed. And in the last experiment, we have used the Expectation-Maximization (EM) clustering algorithm provided by Weka [16] to group together students of similar characteristics when using Moodle (see Table 1). We used this algorithm because it is a well-known clustering algorithm that does not require the user to specify the number of clusters to find. In our case, we obtained three clusters with the following distribution of students:

- **Cluster 0**: 23 students (22 pass and 1 fail).
- **Cluster 1:** 41 students (39 pass and 2 fail).
- **Cluster 2**: 20 students (13 fail and 7 pass).

Clustering algorithms provide a high interpretable result model by means of the values of each cluster centroid (Table 3). The centroid represents the most typical case/student or prototype in a cluster, which does not necessarily describe any given case in that cluster.

Table 3. Values (mean±std.dev.) of centroids of each cluster.

Attribute	Cluster 0	Cluster 1	Cluster 2
Time Theory	5.9±3.2	7.1±2.6	3.9±1.5
Time Tasks	14.1±2.7	11.3±6.2	5.3±1.9
Time Forums	8.7±7.0	12.4±5.4	7.6±4.5
Days Theory	6.0±2.4	1.6±6.9	8.4±2.6
Days Tasks	3.5±1.0	1.8±0.8	6.8±2.3
Days "hand in"	4.8±0.9	3.0±1.2	9.3±2.2
Number of words in forums	7.5±6.5	9.2±3.5	5.3±3.4
Number of sentences in forums	92.9±23.1	107.8±40.6	78.1±39.4

As shown by the mean values in the different variables, students in clusters 0 and 1 (in which most students pass) obtain higher values than cluster 2 (in which most students fail) in times (theory, task and forums) and numbers (of words and sentences in forums), but lower values in days (theory, tasks and hand in). Cluster 0 gives priority to the procedural level of knowledge, corresponding to the scores at the *Time tasks* and *Days Tasks.* The students comprising that cluster also seem to show an achievement or strategic approach based on the prioritisation of the actions related to the *compulsory assignments*. In contrast, students belonging to Cluster 1 are presumably adopting a more dedicated approach to learning: note that the scores are good whether or not the variables are related with *compulsory or suggested* assignments. Finally, Cluster 2, comprised of students who normally fail, shows maladaptive profiles and a more infrequent use of Moodle.

We then applied process mining over the different datasets (all, pass/mark and clusters) in order to discover students' process models and workflows in each one of these logs. We used ProM, [14] a generic tool for implementing process mining tools in a standard environment. In fact, we applied Heuristic Miner, [1] one of the robust algorithms to investigate the processes in users' behaviour. Heuristic Miner can be used to express the main behaviour registered in an event log. It focuses on the control flow perspective and generates a process model in the form of a Heuristics Net for the given event log. Therefore, the Heuristic Miner algorithm was designed to make use of a frequency based metric and so it is less sensitive to noise and the incompleteness of logs. We used the default threshold parameters of Heuristic Miner algorithm provided by ProM and the fitness as a quality measure (see Table 4). Fitness is a quality measure indicating the gap between the behaviour actually observed in the log and the

behaviour described by the process model. It gives the extent to which the log traces can be associated with execution paths specified by the process model. If model has a poor fitness value, this indicates that the mined process model does not successfully parse most of the log traces. This may be due to the presence of noise, resulting in dangling activities and missing connections. It is also possible that the parameter settings do not notice all connections.

Table 4. Fitness of the obtained models

Dataset	Fitness
All students	0.8333
Pass students	0.9117
Fail students	0.9375
Cluster 0 students	0.9130
Cluster 1 students	0.9024
Cluster 2 students	0.9000

As we can see in Table 4, the lowest fitness was obtained when using all data in which 70 of 84 students fit to the obtained model, that is, 83.33% of all students. On the other hand, all the other models (obtained using both manual and automatic clustering) obtained a fitness value greater than 90% in all the cases. And the highest fitness value was obtained when using data from students who failed, where 15 of 16 students fit to the obtained model, that is, 93.75% of students who failed. So, in this case/experiment we can see that these specific models obtained using manual and automatic grouping/clustering performed/fitted better than the general model obtained from all students.

Next, some information about the level of complexity or size of each one of the obtained models (Table 5) and two examples of obtained models (Figure 2 and 3) are described. The model discovered by Heuristic Miner algorithm is a heuristic net, drawn as a directed cycle graph, which represents the most frequent behaviours of the students of the dataset used.

We have used two typical measures from graph theory (the total number of nodes and the total number of links) in order to see the level of complexity of the models.

Table 5. Complexity/Size of the obtained models

Dataset	N.Nodes	N.Links
All students	32	70
Pass students	113	244
Fail students	12	24
Cluster 0 students	61	121
Cluster 1 students	59	110
Cluster 2 students	38	84

As we can see in Table 5, the smaller or more comprehensible model was obtained with students who failed followed by all students and cluster 2 students. On the other hand, the other three models are much bigger and complex. We think that the reasons for this may be that:

- In the dataset "all students", the students show different behaviour and only have some common actions because there are mixed different type of students (pass and fail students).
- In the datasets "students who failed and cluster 2 students", the students show only some common behavioural patterns because this type of student participates/interacts little with Moodle.
- In the datasets "students who pass, cluster 0 and cluster 1", students show much more common behavioural patterns because these types of students are more active users of Moodle

Finally, two examples of obtained models (heuristic nets) are described. The first example shows the heuristic net obtained when using all students (Figure 2) and the second when using fail students (Figure 3). In our heuristic nets the square boxes represent the actions of the students when interacting with Moodle's interface, and the arcs/links represent dependences/relations between actions

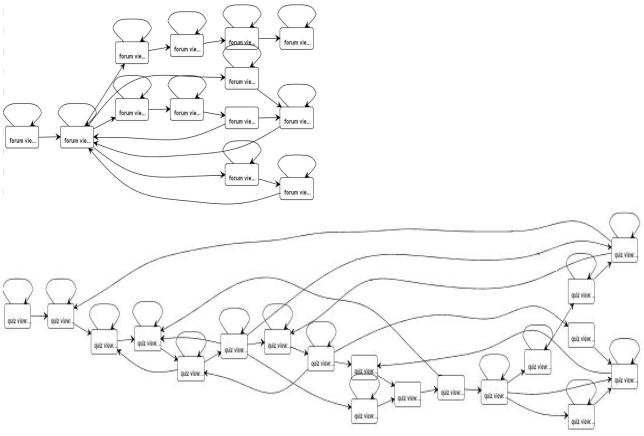

Figure 2. Heuristic net of all students.

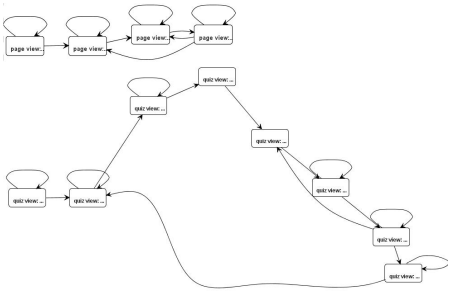

Figure 3. Heuristic net of fail students.

On the one hand, Figure 2 shows two subnets that follow most of the student of the course. The upper subnet consists of some view forum actions about the most viewed forums in the course. And the lower subnet consists of some view quiz actions about the most viewed quizzed in the course.

On the other hand, Figure 3 shows two subnets that follow most of the students who fail the course. The upper subnet consists of some page view actions about the most viewed pages. These pages are about general information on the course. The lower subnet consists of some view quiz actions about the most viewed

quizzed. In this case/dataset, the number of quizzes is much lower than in the previous case (see lower part of Figure 2) and not in the same exact order of visiting.

In summary, both Figures 2 and 3, show the most typical behaviour of the students in each case/dataset. But, it is interesting to see that the heuristic net of students who failed is smaller than the heuristic net of all students. From an educational and practical point of view (to be able to use this information for providing feedback to instructors about student learning), it could easily be used to point out new students at risk of failing the course. For example, instructors only have to check if new students follow the same specific routes/behavioural patterns that shown by the heuristic net of students who failed. That is, if they visit the same pages, view the same quizzes, and in the same order as previous students who failed.

6. CONCLUSIONS

In the present work, we propose to use clustering to improve educational process mining and, at the same time, optimise both the performance/fitness and comprehensibility/size of the model obtained. In particular, the comprehensibility of the model is a core goal in education due to the transferral of basic knowledge that it entails. Making graphs, models or visual representation more accessible or at least, accessible, to teachers and students, makes these results very useful for monitoring the learning process and providing feedback, one of our future goals being to do it in real time. Furthermore, Moodle does not provide specific visualization tools of students' usage data that let the different agents of the learning process understand these large amounts of raw data and become aware of what is happening in distance learning, apart from extending the use of the results to Adaptive Hypermedia Learning Environments in which it is very useful to prompt students or recommend learning paths, shortenings, etc., in order to enhance the learning experience in a more strategic way.

In the future, we want to do more experiments in order to test our proposed approach with other types of courses from different fields. We also want to explore other ways to group students before process mining. For example, grouping students based on the triangulation of the different sources of information used at this work and adding, if possible, self-report data from students' metacognitive behaviour.

7. ACKNOWLEDGMENTS

This work was supported by the Regional Government of Andalusia and the Spanish Ministry of Science and Technology projects, P08-TIC-3720, TIN-2011-22408 and EDU2010-16231, and FEDER funds.

8. REFERENCES

[1] Ayutaya, N. S. N., Palungsuntikul, P., & Premchaiswadi, W. 2012. Heuristic mining: Adaptive process simplification in education. *International Conference on ICT and Knowledge Engineering*. 221-227.

[2] Anuwatvisit, S., Tunggkasthan, A., Premchaiswadi, W. 2012. Bootleneck mining and petri net simulation in education situations. *Conference on ICT and Knowledge Engineering.* 244-251.

[3] Azevedo, R., Behnagh, R., Duffy, M., Harley, J. M., & Trevors G. J. 2012. Metacognition and self-regulated learning in student-centered learning environments. *Theoretical foun- dations of student-center learning environments*. Erlbaum, Mahwah, NJ, 2nd edition, 216–260.

[4] Campos-Rebelo, R., Costa, A., Gomes, L. 2012. Finding learning paths using petri nets modeling applicable to e-learning platforms. *International Federation for Information Processing*, 151-160.

[5] Holzhüter, M., Frosch-Wilke, D., Klein, U. 2013. Exploiting learner models using data mining for e-learning: a rule base approach. *Intelligent and Adaptive ELS*. Springer. 77-105.

[6] Pechenizkiy, M., Trcka, N., Vasilyeva, E., van der Aalst, W. M., & De Bra, P. 2009. Process Mining Online Assessment Data. *Educational Dada Mining Conference*, Cordoba, Spain, 279-288.

[7] Perera, D., Kay, J., Koprinska, I., Yacef, K. and Zaiane, O. 2009. Clustering and Sequential Pattern Mining of Online Collaborative Learning Data. *IEEE Transactions on Knowledge and Data Engineering*. 21(6): 759-772.

[8] Romero, C., Ventura, S. Zafra, A. and De Bra, P. 2009. Applying Web usage mining for personalizing hyperlinks in Web-based adaptive educational systems. *Computer&Education*, 53, 828-840.

[9] Romero, C., Lopez, M.I., Luna, J.M., and Ventura, S. 2013. Predicting students' final performance from participation in online discussion forums. *Computers&Education*. 68,458–472.

[10] Siemens, G., Baker, R.S.J.d.. 2012. Learning Analytics and Educational Data Mining: Towards Communication and Collaboration. *International Conference on Learning Analytics and Knowledge*. 1-3.

[11] Southavilay, V., Yacef, K., Calvo, R.A. 2010. Process mining to support student's collaborative writing. *Educational Data Mining Conference*, 257-266.

[12] Trcka, N., Pechenizkiy, M. 2009. From Local Patterns to Global Models: Towards Domain Driven Educational Process Mining. *International Conference on Intelligent Systems Design and Applications*, Milan, Italy, 1114-1119.

[13] Trcka., N. Pechenizkiy, M. van der Aalst, W. 2011. Process mining from educational data. *Educational Data Mining Handbook*. CRC Press.

[14] Van der Aalst, W. 2011. Process Mining: Discovery, Conformance and Enhancement of Business Processes. Springer.

[15] Weijters, A. J. M. M., van der Aalst, W. M., & De Medeiros, A. A. 2006. Process mining with the heuristics miner-algorithm. Technische Universiteit Eindhoven, Tech. Rep. WP, 166.

[16] Witten, I.H., Eibe, F., Hall, M.A. 2001. Data Mining, Practical Machine Learning Tools and Techniques. Third Edition. Morgan Kaufman Publishers.