Tell the student: evidence-based advantages of prerequisites (Discussion Paper)*

Marco Cameranesi[1], Claudia Diamantini[1], Laura Genga[2], and Domenico Potena[1]

[1] Dipartimento di Ingegneria dell'Informazione, Università Politecnica delle Marche, via Brecce Bianche, 60131 Ancona, Italy
[2] Department of Mathematics and Computer Science, Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands
{m.cameranesi,c.diamantini,d.potena}@univpm.it, {l.genga}@tue.nl

Abstract. When bachelor and master degrees were introduced in the Italian system, the Engineering Faculty of Università Politecnica delle Marche dropped mandatory prerequisites between modules from its degrees. Since then, we periodically witness intense debates among professors, who think this led to unstructured learning practices with consequent performance issues, and students who are concerned of an overly constrained course of study. This paper briefly describes an ex-post analysis of the carriers of students enrolled in the bachelor degree in Computer and Automation Engineering based on process mining techniques, the resulting evidence about typical patterns followed by students, and sketches possible actions that can be put in practice to support students without constraining their course of study.

1 Introduction

Digital technologies in learning received much attention in the last years. In particular, educational data mining is a research field by itself [3]. Leveraged by e-learning environments and MOOC, data mining techniques can be exploited on an increasing amount of digital resources related to almost every step of the learning process, to perform a wide variety of tasks: prediction of future enrollments, abandonments or outcomes; latent knowledge estimation; mining of relations between certain students characteristics and their learning style; clustering of schools, courses or students, just to name a few.

In this paper we briefly describe the application of process mining techniques to the carriers of students enrolled in the bachelor degree in Computer and Automation Engineering, with the aim to understand the typical paths followed by students in taking exams, and their relations with final outcomes. The work is motivated by the intense debate among professors and students that arise from time to time on mandatory prerequisites. Mandatory prerequisites among

* This work has been partially funded by the ITEA2 project M2MGrid (No. 13011).

modules have been dropped from courses of the Faculty of Engineering of the Università Politecnica delle Marche. While somebody would like to re-introduce them in order to limit unstructured learning practices, students are concerned of an overly constrained course of study. The work is part of the monitoring activities in charge of the Faculty committee named "Commissione Paritetica" [5], in which one of the authors is involved. Similar applications of Process Mining techniques exist in the literature. They are mainly devoted to exploit event logs generated by Learning Management Systems like Moodle to discover the workflow of learning activities performed inside a unit of learning, or teaching module (e.g. reading documents, doing assignments, accessing the forum, etc.) [10, 15], or to model patterns of examinee behaviors during assessments [12]. The work in [9] shares some similarities with the present work, in that it tries to find correlations between outcome variables and the sequence of modules, by segmenting traces on the basis of key performance indicators.

The rest of the paper is organized as follows: in Section 2 we provide a brief introduction to Process Mining and techniques used in the paper. Section 3 describes data, their processing and the results of mining activities. Section 4 introduces some reflection and draw future work.

2 Background

Process Mining encompasses a wide range of techniques for the analysis of business processes from run-time information recorded in event logs [1].

A *process* consists of a set of activities that have to be performed by some actors to reach a certain goal. Welfare grant approval is an example of business process, that involves activities like `application_submission`, the analysis of documentation, possibly split in `financial_check` and `check_for_illness_conditions` performed by two different units in parallel, and a final `approval` decision. A *process schema* is a general schema representing the overall control-flow of activities. The main elements of a process schema are activities, and control-flow operators like sequence, parallelization and merging, alternative choices, and loops. Many different process modeling formalisms exist, among them Petri Nets is the best known theoretical model. A *process instance* is a specific execution of the process. In our example, each single welfare grant application represents a process instance. Process instances are recorded by information systems in *event logs* in the form of *traces*, i.e. sequences of events generated by the execution of activities in their strict order of occurrence. The core information recorded in an event log covers the unique id of the event occurred, the associated activity, the time of occurrence and the instance, or case, it belongs to. The name of the actor executing the activity, or its role, can be also reported. Depending on the information system, other information can be recorded, like for instance the data processed by the activity, the case category and so forth. This kind of information is often referred to as data attributes and this kind of logs are also called data annotated logs.

In Process Mining, two major tasks can be recognized:

– Process Discovery: synthesize a process schema from event logs.
– Conformance Checking: analyze the event log to check whether it fits a given process. If the model is a normative or prescriptive (hence a-priori correct) model, conformance checking is interpreted as a way to measure the adherence of real process executions to those prescribed or planned, however it can also be seen as a way to evaluate the accuracy of a model mined by a Process Discovery technique in capturing the content of the event log.

The basic idea of Process Discovery techniques is to exploit event logs to reconstruct causal relations among events that are then used to build a model of the process. In order to deal with variability and noise in real-life logs, *filtering approaches* have been introduced. The infrequent Inductive Miner (iIM) algorithms [11] adopted in this paper belongs to the class of filtering, block-based techniques, that infer a model on the basis of a process tree built on frequent behaviors only. The iIM algorithm adopts an iterative procedure. First, it generates a *directly-follows graph*, i.e. a graph where each node corresponds to an event, and an arc exists between two nodes A and B if event A appears immediately before event B in at least one trace. The number of occurrences of each arc in the log is recorded. The graph is then analyzed to find a *cut*, i.e. a split of activities into disjoint sets such that all the activities in a set are in the same flow relation with all the activities of the other subsets. Sequence, exclusive choice, parallel and loop operators each have their own characteristic cut. If a cut is found then the corresponding operator is chosen as a node of the process tree, and the log is in turn partitioned into sets that reflect the partition choice for the nodes of the graph. The procedure is repeated until each sublog contains only a single event. A user-defined *noise threshold* $0 \leq k \leq 1$ allows to control filtering. For instance, an arc can be removed from the directly-follows graph if the ratio between its number of occurrences and the occurrences of the strongest outgoing arc from the same node is less than k.

Conformance Checking is based on the notion of *replay*: each trace is simulated on the process model one event at a time. When the current event cannot be found at a given step, an *alignment* procedure is adopted to continue the replay procedure, by either skipping the event in the log or the activity in the process schema. Besides, a measure of the fitness of the trace to the model is calculated [13]. In the present work we will adopt the cost-based fitness measures defined in [2].

3 Mining of Students' Careers

The major questions we like to answer can be stated as follows: "Do students take (exams of) modules in the suggested order?" and "Which are *best practices* and *worst practices* followed by students?". The order among teaching modules is related to prerequisites suggested in modules programs, so that module A should be taken before module B if A's contents are fundamental or useful to understand B's contents. Of course there is only a partial order among modules, since courses may deal with different, unrelated topics. Furthermore, best/worst

practices are defined with respect to careers performances, as expressed by the final grade mark and length of the period of study.

With these questions in mind, we exploited process mining as a descriptive mining technique to obtain a synthetic view of the paths followed by students from enrollment to bachelor degree, in the form of process schemas. The methodology can be summarized as follows: after some data pre-processing, we obtain a set of traces representing students' careers. Traces are segmented into three disjoint groups depending on the final grade mark and length of the period of study. Then, for each group, a process model is synthesized in the form of Petri Net by the iIM algorithm. Accuracy of the models is assessed by measuring its fitness over the entire set of traces in the group. In the following we provide some details of these steps and of the results obtained.

3.1 Data Selection and Preparation

Data are mainly taken from ESSE3, a software suite commonly adopted by Universities to manage students' careers and in particular registration to exam sessions and record of grades. We focused the analysis on the latest educational system, as defined by D.M. 270/2004, considering students enrolled from Academic year 2009/10 on. This provides us with a set of homogenous curricula. Furthermore only complete and correct curricula are considered, discarding students who have not taken the degree yet, or students with exams confirmed from previous degrees (these exams do not appear as database records, determining oversimplified and short curricula). Further cleaning procedures were necessary in order to deal with inconsistencies and null values generated during database population at the time of ESSE3 software adoption. At the end, the careers of 187 graduated students were available for analysis. In order to reduce noise, a further step was to delete from curricula the diverse modules freely chosen by students from other courses of Università Politecnica delle Marche.

The final dataset is a table with the following attributes: student ID, module name, exam date, exam mark, duration of the period of study (calculated as the difference between the final exam date and the enrollment date, conventionally fixed to the 1st of November of the Academic year of enrollment). The dataset can be interpreted as a very simple event log, where each student is a case, and exam passing is the set of events with their timestamp.

Additional data attributes related to exam mark and duration are exploited to segment the event log. As a matter of fact, a plain application of process mining techniques to the whole event log did not lead to meaningful process schemas. This can be expected, since the many optional choices and the lack of mandatory prerequisites make the course of study very flexibly customized by students according to their needs and contingencies. In order to deal with this kind of unstructured processes, different methodologies have been devised, among which trace clustering and its process cube variant are pre-processing techniques that are related to the approach followed in this study. In both approaches the idea is to generate different groups of traces with homogeneous characteristics such that better models can be obtained for each group. While

in trace clustering similarity is based on the sequence of events inside traces (inner or structural trace similarity) [8], the process cube approach segment traces according to common values of data attributes, independently from the trace structure. Data attributes can be organized into hierarchies, thus implementing a multidimensional cube of traces [4]. In order to understand more clearly whether prerequisites have an impact on final performance, we decided to enrich the log with exam mark and duration, and exploit them to segment students' careers into three groups: high performance, medium, and low performance. Table 1 shows the definition of groups and careers distribution within each group. As one can note from the table, the definition of thresholds is such that careers are roughly equally distributed in each group.

Table 1. Data attributes, thresholds and careers distribution. The fraction of year in the second column allows to take into account the last graduation session of each academic year, that usually takes place in February. Dark gray cell: high performance careers. Light grey cells: low performance careers. White cells: medium careers.

		Avg. Mark		
		$m \geq 27/30$	$25 \leq m < 27$	$m < 25$
Study	$d \leq 3.3$	27.27%	12.83%	3.74%
Duration	$3.3 < d \leq 5.3$	7.49%	15.51%	16.58%
(years)	$d > 5.3$	0.53%	4.81%	11.23%

3.2 Experiments

The ProM framework[3] has been adopted to perform experiments, in particular the infrequent Inductive Miner (iIM) and Replay a Log on Petri Net for Conformance Analysis (Replay) plugins. For each group of careers, iIM has been launched with different noise thresholds and the different models obtained have been evaluated on the basis of the fitness value calculated by Replay. In what follows we discuss the best Petri Net models generated for high performance and low performance careers. The former is shown in Figure 1. It is obtained with noise threshold equal to 0.3 and has a fitness equal to 96%. Circles and boxes represents places and transition respectively. In particular white boxes corresponds to events in the log (i.e., exams in our case study), while black boxes are invisible transitions, they do not correspond to real events and are generated by iIM in order to obtain a sound model. We can appreciate in the initial part of the model a fairly structured flow, magnified in Figure 2, followed by a more irregular flow. The process starts with three parallel events, corresponding to exams taken for modules of Algebra Lineare e Geometria, Fisica 1, Analisi Matematica 1. This means that, although there is not a strict precedence relation among these exams in the log, students typically take all the three before moving to

[3] www.promtools.org/prom6

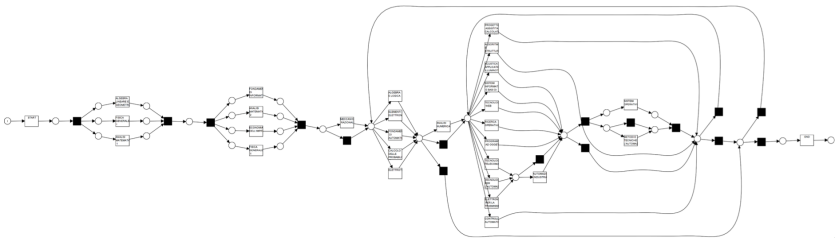

Fig. 1. Process model of best performing students.

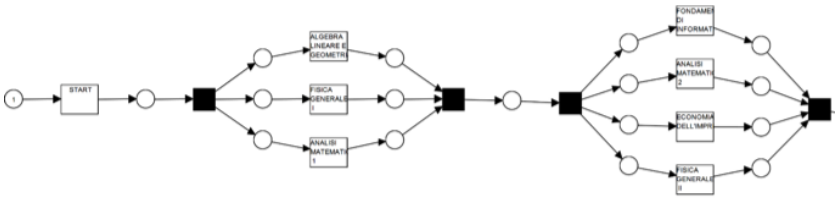

Fig. 2. Process model of best performing students: initial part.

other exams. The three courses are those scheduled during the first semester of the first year. Similarly, the next four parallel events correspond to modules scheduled during the second semester of the first year. Hence, we can see that students with the best careers basically follow the ideal order in taking exams of the first year. Subsequently, the schema shows a less structured flow. This is expected, since free choice modules are introduced from the second year on, due to the presence of two different curricula offered by the course (Computer Engineering and Automation Engineering respectively). Nevertheless, we can still appreciate that the first three groups of events all correspond to modules taught in the second year.

The model is even more interesting when contrasted with that generated from low performing careers, shown in Figure 3. Here, the only regularity is that students choose to take their first exam among six courses taught in the first year (Fisica 2 being the one missed in the list), and that in a number of cases the sequence of exam for Automazione Industriale, Progettazione Assistita da Calcolatore, and Laboratorio di Automazione is followed. No other regularity can be appreciated, suggesting the lack of any criteria in exams flow, and enlightening striking differences in the behaviors of students in the two groups.

4 Discussion

The one presented is a preliminary analysis of students careers, and conclusions must be taken cautiously, in particular for what concerns the cause-effect relationship between high performance and structured course of study: is it that

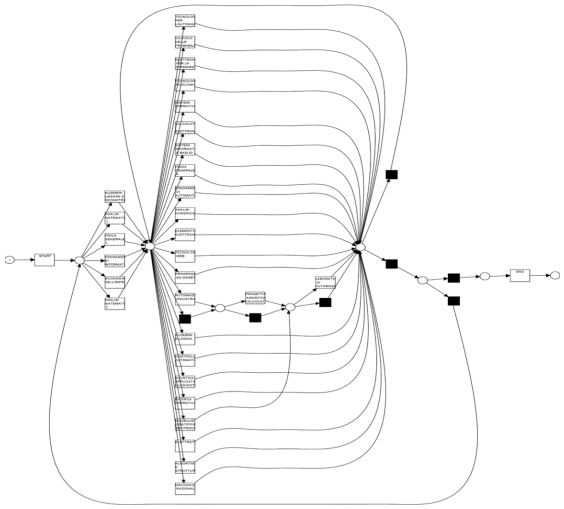

Fig. 3. Process model of worst performing students.

following the "right" order of exams leads students to high performance, or is it the other way around, i.e. students with a natural propensity to computer and automation engineering (which in principle lead to high performance somewhat independently from other factors) have a "structured mindset" and thus follow the order proposed in course regulations? Nevertheless, the evidence of such striking differences between the two groups led us to disseminate results among students, both inside different Faculty committees and during the degree introductory activities, and they have been welcomed by students and professors as the first evidence-based discussion on the advantages of prerequisites. The results also stimulated the board of degree in Computer and Automation Engineering to spread among students information about modules' prerequisites, and to develop actions able to increase awareness on the potential benefits of a thoughtful schedule of exams.

Several further analyses can be performed to support and refine results: first of all, we started replicating the methodology to other degrees, with encouraging results. Second, more refined clustering techniques, based both on finer categories and on structural trace similarity may allow us to discover more precise models. Third, we plan to apply different mining techniques: on the one hand we will consider *local* process mining techniques, tackling unstructuredness of processes by discovering simpler and more precise models of just parts of the process, instead of start-to-end models [6, 7, 14], on the other hand we plan to compare results with those of more traditional data mining techniques, in order to enlighten the advantages of a process perspective of student careers. Finally, information on student's background in terms of secondary school diploma and grade can be exploited to check cause-effect hypotheses. If further analyses will confirm the robustness of the approach, we plan the design of a semi-automatic

advisory system, based on mined models, that can inform students when their careers diverge significantly from best practices and/or show some other worrying signs, thus moving from a descriptive to a predictive scenario.

Acknowledgements

We thank the Commissione Paritetica of the Faculty of Engineering, Università Politecnica delle Marche for his support in the development of the analysis presented in this paper. We also wish to thank Dott. Ing. Matteo Marzioli for the work done in carrying out experiments.

References

1. van der Aalst, W.: Process Mining: Discovery, Conformance and Enhancement of Business Processes. Springer (2011)
2. Adriansyah, A., van Dongen, B., van der Aalst, W.: Conformance Checking Using Cost-Based Fitness Analysis. In: 15th IEEE EDOC Conf. pp. 55–64 (2011)
3. Baker, R.S., Inventado, P.S.: Educational Data Mining and Learning Analytics, pp. 61–75 (2014)
4. Bolt, A., van der Aalst, W.: Multidimensional process mining using process cubes. In: Int. Conf. on Enterprise, Business-Process and Information Systems Modeling. pp. 102–116 (2015)
5. Commissione Paritetica per la didattica e il diritto allo studio: Relazione Annuale 2016, Facoltà di Ingegneria, Università Politecnica delle Marche (December 2016)
6. Diamantini, C., Genga, L., Potena, D., van der Aalst, W.: Building instance graphs for highly variable processes. Expert Systems with Applications 59, 101–118 (2016)
7. Diamantini, C., Genga, L., Potena, D., Storti, E.: Pattern discovery from innovation processes. In: Int. Conf. on Collab. Techn. and Sys. pp. 457–464 (2013)
8. Greco, G., Guzzo, A., Pontieri, L., Sacca, D.: Discovering expressive process models by clustering log traces. IEEE Trans. Know. and Data Eng. 18(8), 1010–1027 (2006)
9. Hicheur Cairns, A., Gueni, B., et al.: Custom-Designed Professional Training Contents and Curriculums through Educational Process Mining. In: Fourth Int. Conf. on Advances in Information Mining and Management. pp. 53–58 (2014)
10. Kelly, N., Montenegro, M., et al.: Combining event- and variable-centred approaches to institution-facing learning analytics at the unit of study level. Int. Jour. of Information and Learning Technology 34(1), 63–78 (2017)
11. Leemans, S.J.J., Fahland, D., van der Aalst, W.: Discovering block-structured process models from event logs containing infrequent behaviour. In: Business Process Management Workshops, LNBIP 171. pp. 66–78 (2014)
12. Papamitsiou, Z., Economides, A.A.: Process mining of interactions during computer-based testing for detecting and modelling guessing behavior. In: Third Int. Conf. on Learning and Collaboration Technologies. pp. 437–449 (2016)
13. Rozinat, A., van der Aalst, W.: Conformance checking of processes based on monitoring real behavior. Inf. Sys. 33(1), 64–95 (2008)
14. Tax, N., Sidorova, N., Haakma, R., van der Aalst, W.: Mining local process models. Journal of Innovation in Digital Ecosystems 3(2), 183 – 196 (2016)
15. Vidal, J.C., Vázquez-Barreiros, B., Lama, M., Mucientes, M.: Recompiling learning processes from event logs. Knowledge-Based Systems 100, 160 – 174 (2016)