Analyzing and Improving Educational Process Models using Process Mining Techniques

Awatef Hicheur Cairns[1], Billel Gueni[1], Joseph Assu[1], Christian Joubert[1], Nasser Khelifa[2]

[1]ALTRAN Research, [2]ALTRAN Institute
Vélizy-Villacoublay, France
e-mails: {awatef.hicheurcairns, billel.gueni, assu.joseph, christian.joubert, nasser.khelifa}@altran.com

***Abstract*— Educational process mining is an emerging field in the educational data mining discipline, concerned with discovering, analyzing, and improving educational processes as a whole, based on information hidden in educational datasets and event logs. In this paper, we demonstrate the applicability of process mining techniques, implemented in the ProM framework, to monitor and analyze educational processes in the field of professional trainings. Furthermore, we extended the discovered training processes with performance characteristics and decision rules using performance and decision mining techniques.**

Keywords- Educational Process Mining; Conformance Ckecking; Decision Minin; Performance Analysis; ProM.

I. INTRODUCTION

Given the ever changing needs of the job markets, education and training centers are increasingly held accountable for student success. Therefore, education and training centers have to focus on ways to streamline their offers and educational processes in order to achieve the highest level of quality in curriculum contents and managerial decisions. To respond to these requirements, education and training centers promote more flexible and personalized curriculums where students are free to choose the skills they want to develop, the way they want to learn and the time they want to spend. This tendency is reinforced by the emergence of "e-learning", which represents an increasing proportion of the in-company trainings, while addressing ever wider populations [10]. Educational systems support a large volume of data, coming from multiple sources and stored in various formats and at different granularity levels. These data can be analyzed from various levels and perspectives, showing different aspects of educational processes from the view points of the students, the educators or the directors of education centers [10, 11]. Recently, Educational Process mining has emerged as a promising and active research field in Educational Data Mining [11], dedicated to extracting process related-knowledge from educational datasets. The basic idea of process mining [2] is to discover, monitor and improve real processes by extracting knowledge from event logs (recorded by an information system). In this paper, we focus on educational process monitoring and improvement using performance analysis, conformance checking and process model extension techniques. We take as a case study a professional training dataset of a consulting company involved in the training of professionals. This work is motivated by the fact that training mangers aim to gain more insight in employees' training paths and motivation so they can offer more personalized training courses, according to the job market needs. Therefore, our aim is to (1) analyze training processes and their conformance with established curriculum constraints, educators' hypothesis and prerequisites and (2) to enhance training process models with performance indicators such as execution time, bottlenecks and decision points. We use the process mining tool ProM as an execution framework in our study.

The remainder of this paper is organized as follows: Section II reviews related works. Section III summarizes process mining techniques. In Section IV, we present our motivating example and we show the use of ProM's plugins for the analysis and the enhancement of training process models. Finally, Section V concludes the paper.

II. RELATED WORKS

In [8], process model discovery and analysis techniques, were used to investigate the students' behavior during online multiple choice examinations. In [14], the authors use process mining techniques to analyze a collaborative writing process and how the process correlates to the quality of the produced document. In [16], the authors proposed a technique relying on a set of predefined pattern templates to extract pattern-driven education models from students' examination traces (i.e., by searching for local patterns and their further assembling into a global model). In [16, 17], the authors developed the first software prototype for academic curriculum mining, built on the ProM framework. This tool monitors the flow of curriculums in real-time and return warnings to students if prerequisites are not satisfied. Two clustering approaches were proposed in [4], grouping students relying on their obtained marks and their interaction with the Moodle's course. Performance analysis techniques were used to detect bottlenecks in students' registration processes in [3]. Finally, in our previous work [5], we showed how social mining techniques can be used to

 ISBN: 978-1-61208-415-2

examine and assess interactions between training providers and courses. We also proposed a two-step clustering approach for partitioning training processes depending on an employability indicator. In comparison with our pervious works, we focus in this paper mainly on educational process monitoring, using performance and conformance analysis techniques. We also studied the applicability of process model extension techniques such as decision mining which have never used in the context of educational process mining. Our goal is to show the advantages of these techniques for the analysis of professional training processes in particular and also their limitations regarding the size of the analyzed event logs. This study help us understand which are the most relevant process mining techniques to integrate in our interactive and distributed platform, tailored for educational process discovery and analysis, and which is currently under construction.

III. EDUCATIONAL PROCESS MINING

Process mining focuses on the development of a set of intelligent tools and techniques aimed at extracting process-related knowledge from event logs [2]. An *event log* corresponds to a set of process *instances* (i.e., traces) following a business process. Each recorded *event* refers to an *activity* and is related to a particular process instance. An event can have a *timestamp* and a *performer* (i.e., a person or a device executing or initiating an activity). Typical examples of event logs in education may include students' registration procedures and attended courses, student's examination traces, use of pedagogical resources and activity logs in e-learning environments. The three major types of process mining techniques are: (1) *discovery*, (2) *conformance* and (3) *extension*. *Process model discovery* takes an event log and produces a complete process model able to reproduce the behaviour observed in this log. *Conformance checking* aims at monitoring deviations between observed behaviours in event logs and normative process models [12]. *Compliance checking* aims at measuring the adherence of event logs with predefined business rules or Quality of Service (QoS) definitions [1]. *Process model extension* aims to improve a given process model based on information (e.g., time, performance, case attributes, decision rules, etc.) extracted from an event log related to the same process. The ProM Framework is the most complete and powerful process mining tool, with an extendable pluggable architecture, aimed at process discovery and analysis from all perspectives [18]. ProM supports a wide range of techniques for process discovery, conformance analysis and model extension. In practice, however, ProM presents certain issues of flexibility and scalability, which limit its effectiveness in handling large logs from complex industrial applications [9].

IV. CASE STUDY: AUDITING TRAINING PROCESSES USING PROCESS MINING TECHNIQUES

Our motivating example is based on real-world professional training databases from a worldwide consulting company. This company has around 6 000 employees that are free to choose different training courses aligned with their profiles, during their careers. These training courses are provided by internal or external training organizations. The data collected for analysis reports all the 16 260 training courses followed by 3440 employees, during the last three years, performed by 494 training organisations. This data includes the employees' profiles (identifier, function, and number of years of service), their careers (i.e., the jobs/missions they did) and their training paths.

TABLE I. EXAMPLE OF AN EDUCATIONAL EVENT LOG

Matricul	Profil	Training_Id	Training_Label	Training_Orga_Id	StartDate	EndDate
7	CONSULTANT	Tr 850	EXCEL ELEARNING	Org 13	11/07/2011	31/12/2011
8	CONSULTANT	Tr 769	QF TEST	Org 135	26/04/2011	28/04/2011
9	CONSULTANT	Tr 252	INTERCULTURAL WORKING RELATONS : INDIA	Org 135	01/07/2011	01/07/2011
10	CONSULTANT	Tr 260	SELENIUM	Org 135	25/10/2011	26/10/2011
11	CONSULTANT	Tr 812	UML FUNCTIONAL ANALYSIS	Org 135	24/10/2011	27/10/2011
12	CONSULTANT	Tr 774	DESIGN PATTERNS AND APPLICATION C++	Org 135	08/12/2011	09/12/2011
13	CONSULTANT	Tr 1923	SQL BASIC	Org 135	03/04/2012	05/04/2012
14	CONSULTANT	Tr 813	C++ ADVANCED	Org 135	04/04/2012	06/04/2012
15	CONSULTANT	Tr 2014	XML BASIC AND XPATH	Org 135	10/04/2012	11/04/2012
14	CONSULTANT	Tr 1282	DESIGN PATTERNS AND APPLICATION IN C++	Org 135	13/09/2012	14/09/2012
....	....		....		.....	

A. Dotted Chart Analysis

The *dotted chart* shows the spread of events over time by plotting a dot for each event in an event log thus allowing to gain some insight in the underlying process, its performance and some interesting patterns [15]. The chart has two orthogonal dimensions: time and component types. The time is measured along the horizontal axis of the chart. The component types (e.g., instance, originator, task, event type, etc.) are shown along the vertical axis. Figure 1 illustrates the output of the dot chart analysis (implemented in ProM 6.4 as a plugin) of the training log example using process instances as component type. In this chart, every row corresponds to a particular case of the training process, i.e., all the trainings followed by one employee during the last three years.

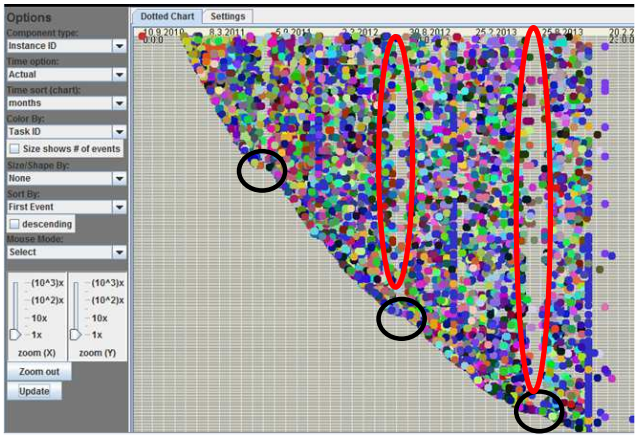


Figure 1. Dotted chart showing all events of the training log

Each training course is represented by a dot. All the instances (one per trainee) are sorted by the first events of trainings i.e., trainings are sorted by the first date of their occurrence. Figure 1 shows that each year, there are few training courses scheduled around the last three months (circled in black), probably because the entire budget is consumed around the end of November. Also, almost no training is scheduled during the summer (circled in red).

B. Trace Alignement

Let us note that in a dotted chart, common patterns among different cases are not clearly visible. For analysing common patterns between the training paths of the company's employees of our case study, we follow the technique presented in [6] and implemented as plugins in ProM 6, where the first step is based on the *`Guide Tree Miner'* plugin and the second steps are handled by the *`Trace Alignment with Guide Tree'* plugin (both implemented in ProM 6.4). The guide tree miner plugin requires an event log as input. This plugin implements the agglomerative hierarchical clustering algorithm for the generation of the guide tree and *k* event logs, one for each cluster. The *`Trace Alignment with Guide Tree'* plugin takes as input a guide tree generated by the guide tree miner plugin and generate an alignment of traces for each cluster. In our example, we have split the training event log into sixteen clusters. Figure 2 depicts the trace alignment for one of the 16 clusters where every row corresponds to a process instance and time increases from left to right. The horizontal position is based on logical time rather than real timestamps. When two rows have the same activity name in the same column, the corresponding events are aligned. The left panel depicts the employee identifier (i.e., process instance identifier) and identifiers with a grey background indicate traces that have identical duplicates. In Figure 2, there are 8 employees who follow the same training path that the employee with the identifier 29784. Also, the traces are sorted based on the training d0 (i.e., Internal Tool Delivriz). Consequently, training traces with d0 in column 3 have the first priority in the ordering.

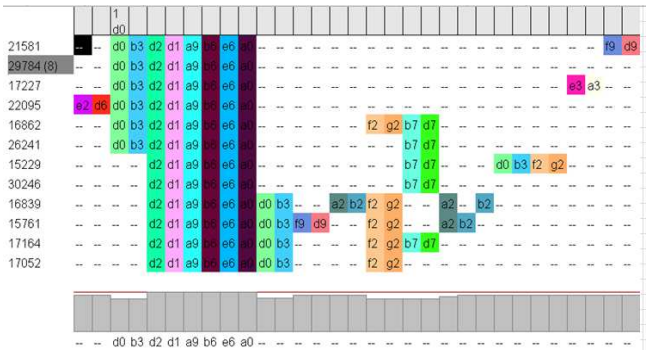


Figure 2. Trace alignment of traces in the training event log example for one of the clusters.

The bottom panel depicts the consensus sequence for the alignment (i.e., the major activity in each column and can be considered as a back-bone sequence for the process). The consensus sequence for this example d0b3d2d1a9b6e6a0 corresponds to a well conserved training pattern "Internal Tool Delivriz, Project Management Module 1, Project Management Module 2 and Project Management Module 3". Figure 2 shows also that the training "Internal Tool Delivriz" (encoded b3) is a concurrent activity. Concurrent activity manifests in mutually exclusive traces across different columns in the alignment. Trace alignment allows us to see such similarities between training traces inducing interesting training patterns. Given the great heterogeneity in training traces, only few clusters produced by the guide tree miner allow us to obtain a good trace alignment. In fact, the challenge of trace alignment is to find an alignment that is as simple and informative as possible when we analyse event log with such heterogeneity as the ones encountered in the education domain. To achieve this, some trivial pre-processing techniques such as the filtering of traces based on their length may help produce a better alignment.

C. Performance Analysis

Performance analysis with Petri net plugin of ProM can extract the Key Performance Indicators from an event log, summarizing them in an intuitive way, and graphically present them on a Petri net describing the process under consideration. When calculating the different performance metrics, all process instances in the input event log are replayed in the input Petri net. We apply this plugin on our training courses dataset in order (1) to predict learners' training path duration, (2) to predict necessary resources (i.e., trainers) at any moment within the training process and (2) to improve the training process by analysing the bottlenecks. Identifying bottlenecks in training paths can be used to better schedule training sessions to reduce the waiting time of trainees. Let us note that given the great heterogeneity in training traces, we can't use the *performance analysis* plugin on our complete training event log [9]. To get past this limitation, we apply the performance analysis plugin on only a subset of the training dataset example containing learning paths with only the six most frequent trainings. The result of the analysis is partly shown in Figure 3 displaying the bottlenecks (red coloured places) and the routing probabilities on each arc. As shown, 6% percent of the employees chose to take the training management project-M1 while 10% followed the training Information Technology Infrastructure Library (ITIL) Foundation. The average waiting time between two successive trainings is 45 days and the maximal waiting time is one year. Moreover, the average duration of a training path is 78 days while the longest training path lasted more than one year.

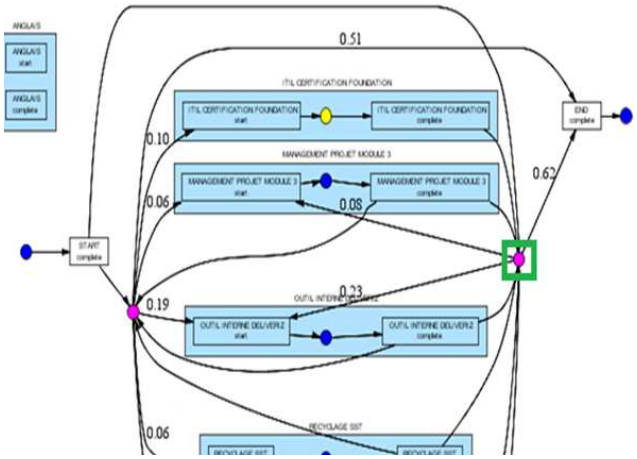


Figure 3. Results of applying the Performance analysis with Petri net plugin

D. *Conformance Analysis*

In what follows, we propose to use ProM's *Conformance Checker* and *LTL Checker* plugins in a training tailored procedure (see Figure 4) to check whether training paths, as they are really followed by employees, are conform to established constraints in the training curricular.

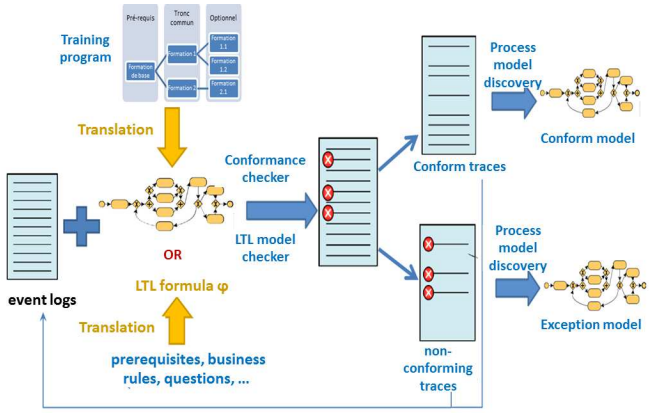


Figure 4. The proposed procedure for conformance analysis of training paths

The conforming and non-conforming traces produced by these two kinds of analysis can be used to extract conform training process models and exception training process models underlying these traces (respectively).

1) *Linear Temporal Logic (LTL) Analysis*

Training advisors and directors of training organisms often need to check (off-line or on-line) whether trainees' paths conform to established career paths, trainings' prerequisites or business rules. For this purpose, we use the *LTL Checker* plugin of ProM that allows us to check whether an event log satisfies a given set of properties expressed in terms of LTL logic [1]. There is a set of predefined formulas in the LTL model checker plugin of ProM. It is also possible to tailor the LTL checker plugin to express specific types of constraints encountered in the educational domain. All these properties can be easily coded using the LTL language of the plugin and imported as a LTL file into the user interface. In what follows, we want to check if the rule "*Project Management-Module-1* training must be taken before a *Project Management-Module* 3 can be taken" was always respected (prerequisite check). We define this property as an LTL formula as follows:

```
formula c2_is_a_prerequisite_of_c1
c1: ate.WorkflowModelElement,
c2: ate.WorkflowModelElement) :=
{<h2> Is the training C2 a prerequisite for the
training C1? </h2>}
(<>(activity==c2)/\(activity!=c2 _U
activity==c1));
```

Fig 5 shows the result displayed when this property is checked.

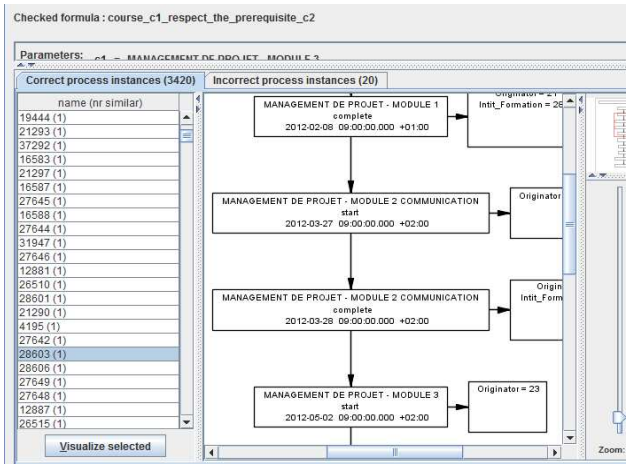


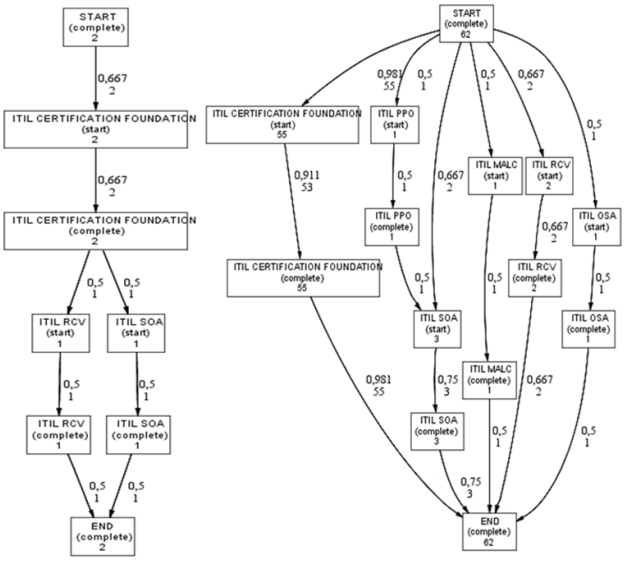
Figure 5. The results returned by the LTL Checker plugin of ProM 5.3 while verifying the Project Management prerequisite

We can see that there are 3420 trainees that satisfy this property and 20 trainees who took the training Project Management–M3 while they didn't take the Project Management -M1 training before.

2) *Conformance Checking*

The *Conformance Checker* plugin supports analysis of the model fitness, precision and structure via log replay, state space analysis, and structural analysis [12]. In what follows, we apply the conformance checker on the training dataset example to verify if the ITIL training program (expressed as a curriculum pattern) is always respected. Our goal is to extract the real ITIL process model as it is followed by trainees during the last three years. We fist apply a filter plugin of ProM on the training log to keep only the traces containing ITIL courses. In a second step, we apply the conformance checker plugin of ProM taking as input the filtered training log and the ITIL training program modelled

as a Petri net. In this program, the training course ITIL Certification Foundation is mandatory. There is also a set of optional training courses: ITIL PPO (Planning, Protection & Optimization), ITIL OSA (Operational Analysis Support) and ITIL MALC (Managing Across the LyfeCycle), ITIL RCV (Release, Control and Validation) and ITIL SOA (Service Offerings & Agreement). From the conforming and non-conforming traces produced by the conformance checking plugin, we mine the process models underlying these two types of traces using the *Heuristic Miner* plugin of ProM (see Figure 6). In Figure 6, the numbers in the boxes indicate the frequencies of the training courses. The decimal numbers along the arcs show the probabilities of transitions between two training courses and the natural numbers present the number of times this order of trainings occur.



(a) Conform training model

(b) Non-conforming training model

Figure 6. ITIL process model mined from the training event log example

The obtained result showed that trainees following a learning path conform to the training ITIL program never took the optional courses ITIL PPO, ITIL OSA and ITIL MALC. Moreover, there are also some trainees who took the same courses as the ITIL program but not in the same order. These results may help training advisors in reviewing the original ITIL training program. They may study the usefulness of some optional courses and propose also new variants of the ITIL training curriculum.

E. Decision Mining in Educational Processes

In a (business) process model, a decision point corresponds to a point where the process is split into alternative paths (e.g., a place with multiple outgoing arcs in a Petri net) [13]. Decision mining (i.e., decision point analysis) aims at the detection of data dependencies that affect the routing of a case. Starting from a process model and a corresponding event log, decision points are identified and data attributes of this log are analysed to determine how case data influence the choices made in the process based on past process executions [13]. In what follows, we analyse choices in training processes to find out which properties of trainee's profile might lead him/her to take certain training paths. To achieve this, we carry out a decision point analysis using the *Decision Miner* plugin of the ProM framework. This later analysis the choice constructs of Petri net process models using the well-known concept of decision trees. The decision miner plugin of ProM takes as an input a Petri net and a corresponding event log. Given the great number of distinct traces in the training log example, we can't use this plugin on the complete training event log. We have first to pre-process this log to reduce its size in order to facilitate the mining of the underlying Petri net model. We started by filtering the training log example, using the *Simple Heuristic Filter* plugin of ProM. We obtained a reduced training log containing employee's training paths referring to the six most frequent training courses in the log. Then, we extract the training process model (represented as a Petri net) corresponding to the filtered training log using the *Alpha Miner* plugin of ProM (see Figure 7).

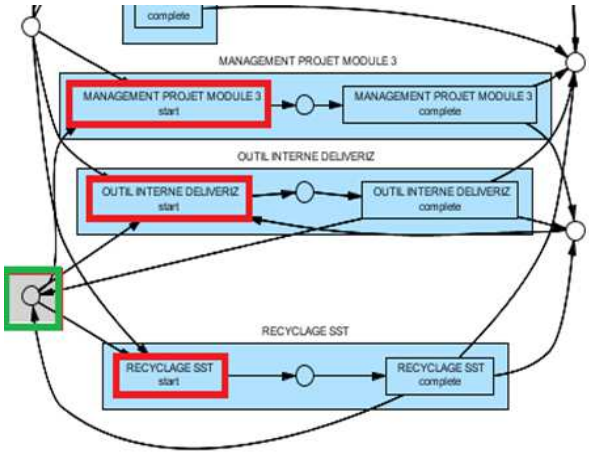


Figure 7. A fragement of the Petri net model underlying the filtered training log containing the six most frequent training courses

The generated Petri net model is then introduced along the filtered training log as inputs in the *Decision Miner* plugin. We choose to study the decision point p_0 (in a green square in Figure 7) which appears after the training course "ITIL certification foundation" to explain the employees' choices between three alternative training paths (Project Management Module 3, Internal Tool Deliviriz or Recycling test). We rely in this analysis on the two case attributes describing an employee profile (i.e., function, number of years of service). Figure 8 shows the decision tree result for the decision point p_0, from which we can now infer the following rule. The training course "Internal Tool

Deliveriz" is chosen by an employee if he/she has been in the company less than 47 months or more than 53 months.

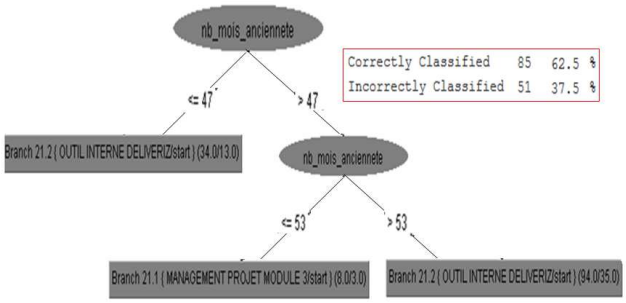


Figure 8. Decision tree result for analysis decision point p0

An employee is more likely to choose the training "Project Management-Module3" if he/she has been in the company between 47 and 53 months. The discovered rule has an accuracy of 62%.

V. CONCLUSION

The aim of our research is to develop an interactive and distributed platform tailored for educational process discovery and analysis. In this paper, we showed how conformance checking, performance analysis and process models enhancement techniques can be used to monitor and improve educational processes in the field of professional trainings. However, performance analysis with Petri net, conformance checking and decision mining, as they are actually implemented in ProM, can't handle heterogeneous and large scale event logs encountered in the professional training field [7, 9]. The adoption of filtering, abstraction or clustering techniques may help reducing the complexity of the discovered process models, and hence the application of advanced analysis techniques [7]. To enhance the usability of our platform, we have also to work on designing an intuitive graphical interface for non-experts that automatically sets parameters and suggests suitable types of analysis.

ACKNOWLEDGMENT

This work is done by Altran Research and Altran Institut in the context of the project PERICLES (http://e-pericles.org/).

REFERENCES

[1] W. M. P. van der Aalst, H. T. de Beer, and B. F. van Dongen, "Process Mining and Verification of Properties: An Approach Based on Temporal Logic," In OTM Conferences, R. Meersman et al., editors, LNCS, 3760 (1):, pp. 130-147, 2005.

[2] W. M. P. van der Aalst et al, "Process mining manifesto," In Business Process Management (BPM) 2011 Workshops Proceedings, pp. 169–194, 2011.

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

[4] A. Bogarín, C. Romero, R. Cerezo and M. Sánchez-Santillán, "Clustering for improving educational process mining,". In Proceedings of the Fourth International Conference on Learning Analytics And Knowledge. ACM, New York, NY, USA, pp. 11-15, 2014.

[5] A. Hicheur Cairns et al., "Towards Custom-Designed Professional Training Contents and Curriculums through Educational Process Mining," IMMM14, pp. 53-58, Jul. 2014, Paris, France.

[6] R. P. Jagadeesh Chandra Bose and W. M. P. van der Aalst, "Process diagnostics using trace alignment: Opportunities, issues, and challenges," Inf. Syst. 37, 2, pp. 117-141, Apr. 2012.

[7] J. Munoz-Gama, J. Carmona, and W. M. P. van der Aalst, "Conformance Checking in the Large: Partitioning and Topology," The 11th International Conference on Business Process Management (BPM 13), pp. 130–145, Aug. 2013, doi:10.1007/978-3-642-40176-3_11.

[8] M. Pechenizkiy, N. Trčka, E. Vasilyeva, W. P. M. van der Aalst, and P. De Bra, "Process Mining Online Assessment Data," The 2nd International Conference on Educational Data Mining (EDM 2009), pp. 279–288, Jul. 2009.

[9] M. Reichert, "Visualizing Large Business Process Models: Challenges, Techniques, Applications," In 1st Int'l Workshop on Theory and Applications of Process Visualization Presented at the BPM 2012, Tallin, pp. 725-736, 2012.

[10] C. Romero, S. Ventura, and E. Garcia, "Data Mining in Course Management Systems: Moodle Case Study and Tutorial," E. Computers & Education, 51(1), pp. 368-384, 2008.

[11] M. Romero and C. Ventura, "Data mining in education," The Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, vol 3, pp. 12–27, Feb. 2013.

[12] A. Rozinat and W. M. P. van der Aalst, "Conformance checking of processes based on monitoring real behavior," Inf. Syst. 33, 1, pp. 64-95, Mar. 2008.

[13] A. Rozinat and W. M. P. van der Aalst, "Decision mining in prom," In Proceedings of the 4th international conference on Business Process Management (BPM'06), Schahram Dustdar, José Luiz Fiadeiro, and Amit P. Sheth (Eds.), Springer-Verlag, Berlin, Heidelberg, pp. 420-425. 2006.

[14] V. Southavilay, K. Yacef, and R. A. Calvo, "Process mining to support students' collaborative writing," The 3rd International Conference on Educatinal Data Mining (EDM 2010), pp. 257-266, Jun. 2010.

[15] M. Song and W. M. P. van der Aalst, "Supporting Process Mining by Showing Events at a Glance," Seventeenth Annual Workshop on Information Technologies and Systems (WITS'07), In K. Chari, A. Kumar editors, Montreal, Canada, pp. 139–145, Dec. 2007.

[16] N. Trčka and M. Pechenizkiy "From Local Patterns to Global Models: Towards Domain Driven Educational Process Mining," In ISDA 2009, pp. 1114–1119, Dec. 2009, doi:10.1109/ISDA.2009.159.

[17] N. Trčka, M. Pechenizkiy, and W. P. M. van der Aalst, "Process Mining from Educational Data (Chapter 9)," Handbook of Educational Data Mining, CRC Press, pp. 123–142, 2010, doi: 10.1201/b10274-11.

[18] B. van Dongen, H. Verbeek, A. Weijters, and W. P. M. van der Aalst, "The ProM framework: a new era in process mining tool support," The 26th International Conference (ICATPN 2005) LNCS Vol. 3536, pp. 444–454, Jun. 2005, doi:10.1007/11494744_25.

 ISBN: 978-1-61208-415-2