

Anwendung von Process Mining zur kontinuierlichen Lernpfadidentifikation in Lernmanagementsystemen

Lars Quakulinski¹, Sven Judel², Miriam Wagner³ und Ulrik Schroeder²

Abstract: Learning Analytics Anwendungen, die mittels Process Mining die Lernpfade von Studierenden identifizieren, machen dies meist nach Kursabschluss auf den vollständigen Daten. Von diesen gewonnenen Kenntnissen und eventuell folgenden Kursanpassungen profitieren jedoch frühestens die Teilnehmenden der nächsten Kursdurchführung. Lehrenden bereits während der Kursdurchführung Einsichten zu geben bietet die Möglichkeit frühzeitig auf eventuelle Probleme zu reagieren. Studierende können ihren eigenen Lernpfad reflektieren und bei Bedarf anpassen. In diesem Beitrag wird eine Anwendung vorgestellt welche einmal täglich die als xAPI Statements gesammelten Daten der letzten 24 Stunden aus dem Lernmanagementsystem Moodle analysiert und die Lernpfade der einzelnen Kurse erweitert. Um eine skalierende Lösung bereitzustellen, werden Techniken des Streaming Process Minings angewandt.

Keywords: Educational Process Mining, Streaming Process Mining, Learning Analytics, xAPI

1 Einleitung

Learning Analytics beschreibt die Sammlung, Messung, Analyse von sowie das Berichten über Interaktionen von Lernenden und deren Kontext mit dem Ziel das Lernen zu verstehen und zu optimieren [Co11]. Datenquellen für solche Analysen sind unter anderem Event Logs aus Lernmanagementsysteme wie Moodle. *Educational Process Mining* (EPM) bezeichnet die Analyse solcher Event Logs um z. B. Einsichten in das Lernverhalten von Studierenden zu gewinnen [SK22]. Bisherige Arbeiten analysieren meist jedoch erst am Ende eines Kurses den vollständigen Datensatz [JZR19]. Um Lehrenden und Lernenden die Vorteile des EPMs schon während einer Kursdurchführung zugänglich zu machen, braucht es eine Infrastruktur, welche die Analysen periodisch durchführt und die Ergebnisse kontinuierlich bereitstellt. Für größeren Universitäten mit mehreren 10.000 Studierenden muss eine gute Skalierung der Infrastruktur gegeben sein. Im Folgenden werden zunächst die theoretischen Grundlagen von EPM und eine geeignete Analyseinfrastruktur vorgestellt sowie die Auswahl eines Algorithmus¹ für die Lernpfaderkennung begründet. Abschließend wird das Dashboard, in dem die Analyseergebnisse dargestellt werden, vorgestellt und fortführende Arbeiten diskutiert.

¹ RWTH Aachen, lars.quakulinski@rwth-aachen.de

² RWTH Aachen, Lerntechnologien, {judel, schroeder}@cs.rwth-aachen.de, <https://orcid.org/0000-0002-0424-0402>, <https://orcid.org/0000-0002-5178-8497>

³ RWTH Aachen, PADS, wagner@pads.rwth-aachen.de, <https://orcid.org/0000-0002-6941-037X>

2 Process Mining und Learning Analytics

Educational Process Mining (EPM) wurde vielfach erfolgreich auf Event Daten aus Lehr- und Lernsituationen, oft aus Moodle, angewandt [SK22]. Dabei werden unter anderem geplante Lernpfade mit real genommenen abgeglichen [Ä16] oder das Verhalten während Moodle Quizzes analysiert [JZR19]. Als Modelle für Lernpfade werden im EPM meist Directly Follows Graphen (DFGs) oder Petri Netze verwendet [SK22]. In den bisherigen Arbeiten werden die Analysen jedoch erst nach Abschluss eines Kurses, auf den vollständigen Event Logs durchgeführt. Es gab somit keine Rückmeldungen während der Kursdurchführung, sodass Teilnehmende nicht von eventuellen Kursanpassungen profitieren. Auch Lehrende können nicht rechtzeitig intervenieren, wenn Probleme schon früher erkennbar sind. Um dies zu ermöglichen muss zunächst eine Infrastruktur für periodische Datenverarbeitungen und kontinuierliche Ergebnisbereitstellung geschaffen werden. Tools wie Celonis⁴ oder Disco⁵ eignen sich aufgrund ihrer kommerziellen oder Closed Source Umsetzung nicht, da die eigenen Anwendung Open Source umgesetzt werden soll.

Analyseinfrastruktur: Eine geeignete Infrastruktur stellt Excalibur LA dar. Verschiedene Systeme können Event Daten als xAPI Statements an diese schicken, sodass sie in einem Learning Record Store (LRS) speichert und mit Analytics Engines verarbeitet werden. Die Ergebnisse werden kontinuierlich über eine REST API bereitgestellt. [JS22] Für Moodle existiert bereits ein Plugin, welches Moodle Events sowie clientseitige Interaktionen in xAPI Statements überführen und verschicken kann [JSS22], welche alle für die angedachten Analysen benötigten Informationen enthalten. Ein Dashboard zur Darstellung der Ergebnisse in Moodle ist ebenfalls gegeben, muss jedoch um eine geeignete Visualisierung für die Lernpfade erweitert werden. Die Existenz dieser Tools sowie deren Erweiterbarkeit und Skalierbarkeit stellen eine gute Grundlage für periodisches EPM dar.

Streaming Process Mining: Trotz der skalierenden Infrastruktur hat EPM auf dem Event Logs aller Kurse Grenzen. Bisherige Erfahrungen an der RWTH Aachen mit mehr als 47.000 Studierenden zeigen, dass etwa 2 Millionen xAPI Statements pro Tag von Moodle erwartbar sind. Das Abrufen der Statements einer Woche dauert bereits über 13 Stunden. Die Datenmenge stellt auch für den Arbeitsspeicher eine Herausforderung dar, da in einem Semester mindestens 347,6 GB an Statements zu verarbeiten sind. Die Literatur nennt einige Techniken zum Umgang mit solchen Mengen wobei die meisten, wie z. B. Dekompositionstechniken [LFVdA18], immer noch den vollständigen Datensatz benötigen. Einzig *Streaming Process Mining* Techniken benötigen nicht die Gesamtheit an Daten [Bu22]. Hierbei wird der Event Log erst in eine abstrakte, deutlich kompaktere, Repräsentationsform, wie z. B. ein DFG [VVV18], überführt, auf dessen Basis anschließend Prozess Modelle, wie das Petri Netz, konstruiert werden können. Beim Verarbeiten neuer Events ist es ausreichend, die abstrakte Repräsentation zu aktualisieren und darauf ein neues Petri Netz zu identifizieren.

⁴ <https://www.celonis.com/>, abgerufen am 18.06.2023

⁵ <https://fluxicon.com/disco/>, abgerufen am 18.06.2023

3 Streaming Process Mining in Excalibur

Während der Fuzzy Miner simplere Modelle [Sa21], was Lesen und Interpretieren erleichtert, erzeugen kann, schnitt der Inductive Miner (IM) im Vergleich mit unter anderem dem Heuristic und Inductive Miner am besten ab [BCR18]. Die nur auf dem DFG operierende Variante des IM [VdA16] ist für Streaming Process Mining geeignet, sodass diese für Excalibur gewählt wurde. Da Analytics Engines in Python implementiert werden, wurden zunächst bestehende PM Packages evaluiert. Das Open Source Package PM4Py⁶ unterstützt den Inductive Miner auf DFGs und gibt das resultierende Modell maschinenlesbar aus, sodass es sich am besten für den Anwendungsfall eignet. Um die Darstellungen von DFG und Petri Netz im Dashboard zu ermöglichen, wird die Detektion der Lernpfade durch zwei Engines umgesetzt. Entsprechend konstruiert und speichert die erste Engine, der *DFG Discoverer*, für jeden Kurs den DFG als abstrakte Repräsentation des Event Logs eines Tages und reichert zusätzlich einen kumulierten DFG über alle bisherigen Events des jeweiligen Kurses an. Primär sollen die Lernpfade die Daten der gesamten Kursdauer widerspiegeln. Um aber auch Verhaltensänderungen zwischen verschiedenen Zeiträumen erkennen zu können, soll es möglich sein, die Lernpfade zeitlich einzuschränken. Wird später ein kleinerer Zeitraum gefiltert, können die DFGs der ausgewählten Tage verbunden und darauf ein neues Petri Netz erstellt werden. Die zweite Engine, der *LP Discoverer*, generiert aus diesem kumulierten DFG die Lernpfade in Form eines Petri Netzes.

Um einen Plattformwechsel zu vermeiden, werden DFG und Petri Netz in einem in Moodle integriertes Dashboard dargestellt. Abbildung 1 zeigt beide Visualisierungen. Über ein Toggle-Element können Nutzende selbst bestimmen, was ihnen angezeigt wird und sowohl das Zeitintervall als auch, im Falle des DFGs, die Kantengewichte filtern.

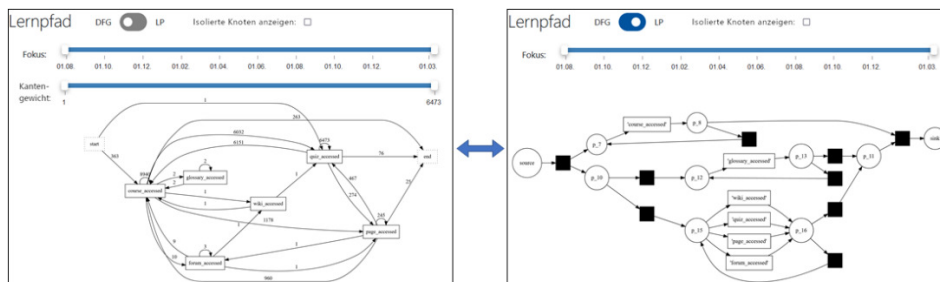


Abb. 1: Darstellung von DFG und Petri Netz sowie der Interaktionsmöglichkeit im Dashboard

Die dargestellten Modelle wurden aus Realdaten⁷ eines Kurses mit 365 Teilnehmenden ermittelt. Dabei wurden die Zugriffe auf Kursinhalte, basierend auf Objekttyp und Verb, betrachtet, was zu einer übersichtlichen Darstellung führt. Im DFG (links) ist anhand der

⁶ <https://pm4py.fit.fraunhofer.de/>, abgerufen am 18.06.2023

⁷ Die Autoren bedanken sich bei Niels Seidel von der FernUniversität Hagen für die Aufbereitung und Bereitstellung dieser Daten.

Kantengewichte erkennbar, dass Glossar, Wiki und Forum kaum genutzt wurden, Textseiten und Quizze hingegen deutlich öfter. Da die Hauptseite des Kurses (*course_accessed*) häufig zur Navigation zwischen Kursinhalten genutzt wird, sind die hohen Werte erwartbar. Das Petri Netz (rechts) erlaubt Einsichten in die Parallelitäten der Zugriffe. Während jedes Mal der Kurs aufgerufen wird, ist zu erkennen, dass entweder das Glossar oder einer der anderen vier Objekttypen aufgerufen wurde. Dabei ist keine Navigation von diesen vier Objekttypen und dem Glossar möglich. Diese Einsichten sind noch limitiert, bereiten aber eine Grundlage für weitere Analysen und Interpretationen (siehe Abschnitt 4).

Abbildung 2 zeigt die gemittelte Laufzeit der einzelnen Analyseschritte, die auf 2,4 Millionen Statement mehrfach ausgeführt wurden. Dabei sind die ersten vier Schritte (blaue Balken) Teil der DFG und die unteren beiden (violette Balken) Teil der LP Discoverer Engine. Die horizontale Linie trennt die unteren drei Schritte ab, da diese sowohl in den Engines auch in den Endpoints ausgeführt werden um den angeforderten, zeitlich gefilterten DFG und das zugehörige Petri Netz zu konstruieren und auszuliefern. Das Gruppieren der Statements und deren Mapping auf Events brauchen die meiste Zeit. Die anschließende Konstruktion der DFGs erfolgt sehr schnell. Gemessen wurde dies für Aktivitäten, die aus Objekttyp und Verb zusammengesetzt wurden. Eine geringe Laufzeit ist speziell in den letzten drei Schritten wichtig, um den Dashboard-Nutzenden schnell die angeforderten, gefilterten Daten auszuliefern und eine gute Nutzbarkeit zu unterstützen. Die abgebildeten Messwerte wurden für die Operationen auf den DFGs aller 3000 Kurse erhoben. Im Durchschnitt dauerte die Petri Netz Konstruktion also nur 2,4 Millisekunden.

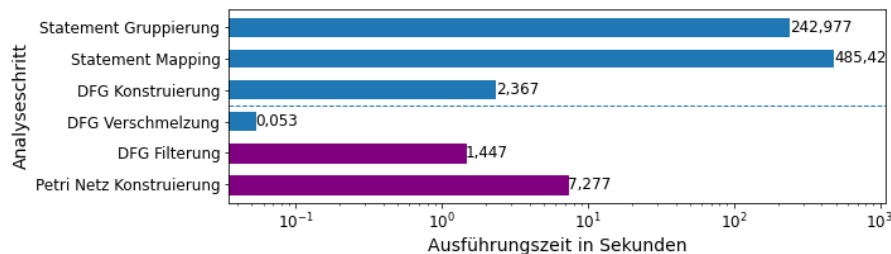


Abb. 2: Benötigte Zeit für die einzelnen Analyseschritte.

Neben der Laufzeit wurde die Qualität der resultierenden Petri Netze evaluiert. Als Aktivitäten wurden sowohl die Kombination von Objekttyp und Verb (OT-V), als auch Objektinstanz und Verb (OI-V) genutzt. Für beide Aktivitätsdefinitionen ist die Fitness mit 98,03% (OT-V) und 98% (OI-V) sehr hoch. Die DFG Variante und die optionalen Filter führen zum Verlust der perfekten Fitness [VdA16]. Die Präzision ist mit 22,69% für OT-V und 1,19% für OI-V vergleichsweise gering. Laufzeit und Modellqualität sind vielversprechend. Negativ fällt nur die Präzision bei der Verwendung von OI-V als Aktivität auf. Dem Dashboard können Ergebnisse in einer angemessenen Zeit ausgeliefert werden. Auch die zeitintensiven Analyseschritte verarbeiten große Datenmengen in nur wenigen Minuten, was für die nachts ausgeführten Analysen als angemessen angesehen wird.

4 Diskussion und weitere Arbeiten

Die umgesetzten Engines und Darstellungsmöglichkeit sind eine gute Grundlage für weitere Arbeiten. Zur Beantwortung konkreter Fragen von Lernenden und Lehrenden können fokussierte Engines entwickelt werden, die sich z. B. auf bestimmte Objekttypen konzentrieren, ähnlich den Quiz-Analysen von [JZR19]. Weitere Engines können auch die vorgeschlagene Kombination von EPM Ergebnissen mit denen anderer Analysen umsetzen. Zusätzlich können nicht nur tägliche DFGs für Kurse, sondern auch für Studierende jeden Kurses erstellt werden. Dies eröffnet das Potential, Studierenden ihren eigenen Pfad aufzuzeigen und individuelle Rückmeldungen zu geben. Manche Rückmeldungen, wie z. B. das Einhalten angedachter Lernpfade, wie in [A16], erfordern zudem initialen Input durch Lehrende. Die Anwendung von *Streaming Conformance Checking* ermöglicht Fitness und Präzision ebenfalls periodisch zu bestimmen und kontinuierlich im Dashboard anzuzeigen. Dies schafft mehr Transparenz und erlaubt die dynamische Filterung des DFGs nach den häufigsten Aktivitäten und Pfaden. Dies kann zwar bereits mit der aktuellen Implementierung realisiert werden, allerdings kann durch die fehlenden Fitnesswerte die Qualität der Modelle nicht eingeschätzt werden. Korrektes und verständliches Feedback könnte die Akzeptanz von EPM erhöhen. Lernenden die Vorteile des EPM zu vermitteln könnte ihre Bereitschaft, der notwendigen pseudonymen Datensammlung zuzustimmen, erhöhen. Die Engines sind so umgesetzt, dass xAPI Statements beliebiger Plattformen analysiert werden können. Studienverlaufsdaten wie z. B. die Belegungsreihenfolge von Modulen, wie in [Wa23], ebenfalls mit dem IM, skizziert, können untersucht werden. Einzig die DFG Discoverer Engine muss dahingehend angepasst werden, dass sie nach Studiengang oder Kohorte statt Kurs gruppiert.

5 Zusammenfassung

Periodisches Educational Process Mining (EPM), z. B. zur Erkennung von Lernpfaden, kann Einsichten in und Feedback zum Lernverhalten der Studierenden bieten. Während viele Arbeiten Anwendungsfälle und Vorteile von EPM durch die Analyse von Event Logs nach Abschluss eines Kurses demonstrieren, wurden in dieser Arbeit eine Möglichkeit zur Bereitstellung von Ergebnissen noch während der Kursdurchführung vorgestellt. Analytics Engines für die Learning Analytics Infrastruktur Excalibur LA analysieren xAPI Statements aus Moodle und erzeugen Directly Follows Graphen sowie Petri Netze. Zur Verarbeitung großer Datenmengen in angemessener Zeit, werden Techniken des Streaming Process Minings angewandt. Die Laufzeit der Analysen ist gut und auch die Operationen, die z. B. beim zeitlichen Filtern der Ergebnisse ausgeführt werden, sind performant. Zwar kann keine perfekte Fitness der Ergebnisse mehr garantiert werden, jedoch sind 98% im gegebenen Kontext ein guter Wert. Die geringe Präzision wiederum muss verbessert werden. Die Analyse von Realdaten zeigt, dass die Visualisierungen von DFGs und Petri Netzen abhängig von der Wahl der Aktivitätserstellung schnell komplex werden und damit schwierig zu lesen sind. Hier könnte ein Fokus auf ausgewählte Aktivitäten

mehr Nutzen bringen. Die vorgestellten Engines sind leicht für solche Spezialisierungen anpassbar und auf die Analysen von Logs anderer Plattformen übertragbar. Zusätzlich können weitere Engines geschrieben und in die vorgestellte Infrastruktur integriert werden, welche Ergebnisse des EPMs und anderer Analysen zusammenführt um Lehrenden und Lernenden weitere Rückmeldungen zu geben.

Literaturverzeichnis

- [BCR18] Bogarín, A.; Cerezo, R.; Romero, C.: Discovering learning processes using Inductive Miner: A case study with Learning Management Systems (LMSs). *Psicothema*, (30.3):322–329, 2018.
- [Bu22] Burattin, A.: Streaming process mining. *Process Mining Handbook*, 349, 2022.
- [Co11] Conole, G. et al.: *Proceedings of the 1st International Conference on Learning Analytics and Knowledge*. 2011.
- [JS22] Judel, S.; Schroeder, U.: EXCALIBUR LA - An Extendable and Scalable Infrastructure Build for Learning Analytics. In: *2022 International Conference on Advanced Learning Technologies (ICALT)*. IEEE, S. 155–157, 2022.
- [JSS22] Judel, S.; Schnell, E.; Schroeder, U.: Performantes xAPI Logging in Moodle. In: *20. Fachtagung Bildungstechnologien (DELFI)*, S. 159–164, 2022.
- [JZR19] Juhaňák, L.; Zounek, J.; Rohlíková, L.: Using process mining to analyze students' quiz-taking behavior patterns in a learning management system. *Computers in Human Behavior*, 92:496–506, 2019.
- [LFV18] Leemans, S.; Fahland, D.; Van der Aalst, W.: Scalable process discovery and conformance checking. *Software & Systems Modeling*, 17:599–631, 2018.
- [Sa21] Saint, J. et al: Using process mining to analyse self-regulated learning: a systematic analysis of four algorithms. In: *LAK21: 11th International Learning Analytics and Knowledge Conference*. ACM, S. 333–343, 2021.
- [SK22] Sypsas, A.; Kalles, D.: Reviewing Process Mining Applications and Techniques in Education. *Artificial Intelligence & Applications*, 13(1):83–102, 2022.
- [VdA16] Van der Aalst, W.: *Process Mining: Data Science in Action*. Springer, 2016.
- [VVV18] van Zelst, S.; van Dongen, B.; van der Aalst, W.: Event Stream-Based Process Discovery using Abstract Representations. *Knowledge and Information Systems*, 54(2):407–435, 2018.
- [Wa23] Wagner, M. et a.: A Combined Approach of Process Mining and Rule-Based AI for Study Planning and Monitoring in Higher Education. In (Montali, M.; Senderovich, A.; Weidlich, M., Hrsg.): *Process Mining Workshops*, S. 513–525. Springer, 2023.
- [Á16] Álvarez, P. et al.: Alignment of teacher's plan and students' use of LMS resources. Analysis of Moodle logs. In: *2016 15th International Conference on Information Technology Based Higher Education and Training (ITHET)*. S. 1–8, 2016.