УДК 004.94

ОСОБЕННОСТИ ПРИМЕНЕНИЯ АЛГОРИТМОВ ПРОЦЕССНОЙ АНАЛИТИКИ (PROCESS MINING) ДЛЯ АНАЛИЗА ПОВЕДЕНИЯ СТУДЕНТОВ

А.А. Логинова
Аспирантка КГУ, ассистент кафедры информационных систем и технологий КГУ
aloginova255@gmail.com

М.Д. Попов
Магистрант института автоматизированных систем и технологий КГУ
milan070699@gmail.com

А.Р. Денисов
Профессор кафедры информационных систем и технологий КГУ, д.т.н.
iptema@yandex.ru

А. А. Логинова
Является аспиранткой Костромского государственного университета. Работает ассистентом кафедры информационных систем и технологий Костромского государственного университета.

М. Д. Попов
Является магистрантом института автоматизированных систем и технологий Костромского государственного университета.

А. Р. Денисов
Профессор кафедры информационных систем и технологий Костромского государственного университета, доктор технических наук.

Аннотация. Рассматривается проблема анализа действий студентов с целью повышения гибкости образовательных траекторий. Предлагается анализировать деятельность учащихся на основе данных цифровых следов, которые студенты оставляют в системах управления обучением. Одним из способов анализа таких данных названа процессная аналитика (Process Mining). В рамках данной работы рассматриваются особенности алгоритмов Process Mining и особенности применения этих алгоритмов с точки зрения анализа журналов событий в системе управления обучением.

Ключевые слова: процессная аналитика, интеллектуальный анализ образовательных процессов, цифровой след, система управления обучением.

Введение.

При реализации образовательных программ важно обеспечить формирование обязательных компетенций, прописанных во ФГОС. За время обучения в вузе каждый студент должен получить этот набор компетенций, чтобы получить документ об образовании.

Однако зачастую влияние субъективных факторов может привести к тому, что студент выбирает направление подготовки неосознанно, в связи с чем наблюдается отсутствие мотивации и, как следствие, низкие результаты образования. Навыки студента студента также неоднородны: если одни дисциплины даются ему достаточно легко, то с другими могут возникать сложности, препятствующие освоению образовательной программы. Компетенции, получаемые студентом в данном случае, могут не соответствовать требованиям рынка труда, что негативно сказывается на взаимодействии университета с работодателями.

Для решения этой проблемы необходимо иметь возможность анализировать деятельность студентов. В условиях дистанционного и смешанного обучения студенты оставляют множество так называемых цифровых следов – данных, содержащих информацию о процессах учебной деятельности и хранящихся преимущественно в журналах событий систем управления обучением (Learning Management System, LMS). Применяя к ним методы анализа данных, в частности, методы процессной аналитики,

можно определить особенности учащихся, в том числе стиль обучения, личные интересы и предпочтения, результаты обучения и т. д., а также спрогнозировать успеваемость учащегося и предложить рекомендации по процессу обучения, предоставить возможность получения дополнительных компетенций [1]. Кроме того, анализ данных позволит разработать систему формирования компетенций в соответствии с требованиями ФГОС и работодателей-партнеров.

Актуальность.

Оценка деятельности студентов методами процессной аналитики является достаточно перспективным направлением в сфере современного высшего образования. Это обусловлено изменением тенденций в сфере высшего образования. В частности, меняются требования к образовательным результатам: в данный момент рынок труда требует от выпускников вузов наличия определенных компетенций, которые не всегда соответствуют навыкам, приобретаемым студентами в процессе обучения, и которые сложно оценить существующими методами. По этой причине вузам необходимо перестраивать образовательный процесс таким образом, чтобы отвечать предъявляемым требованиям. Существующие системы, анализирующие деятельность студентов, в том числе системы управления обучением, рассматривают компетенции студентов, но не проводят анализ их мотивации и оценку «мягких» навыков [2].

Educational Process Mining

Задача оценки индивидуальных особенностей студентов является достаточно сложной. Ее решение предполагает анализ процессов получения образовательных результатов, сохраненных в логах системы LMS [3]. Для решения таких задач используется группа методов интеллектуального анализа данных, которая получила название процессной аналитики (Process Mining, PM) [4]. Процессная аналитика основана на выделении и формализации повторяющихся последовательностей действий, или паттернов. Выявление паттернов поведения позволит со временем быстрее принимать решения, основываясь на схожих ситуациях в прошлом [5].

В сфере решения задач образования говорят об аналитике образовательных процессов (Educational Process Mining, EPM). С помощью методов EPM можно выявить типовые паттерны поведения студентов при получении различных образовательных результатов, определить соответствие поведения студента ранее выявленным паттернам, выявить социальные связи между студентами и т. п. [6–9].

В настоящее время существует несколько методов анализа данных из систем управления обучением, одним из которых является интеллектуальный анализ образовательных процессов (Educational Process Mining). Его целью является поиск поведенческих паттернов, типичных для определенных групп учащихся [10].

Особенности алгоритмов Process Mining

Основным источником данных для процессной аналитики является журнал событий, или журнал рабочего процесса [11]. Он представляет собой электронную таблицу, таблицу базы данных или файл, содержащий записи о последовательности событий. Каждое событие представляет собой строку в журнале событий и содержит данные о действиях, задачах, временных отметках. Формально журнал рабочего процесса определяется следующим образом. Пусть T – набор задач. Тогда $\sigma \in T*$ – трассировка рабочего процесса, а $W \in P(T*)$ – журнал рабочего процесса. Здесь $P(T*)$ – множество мощности $T*$, т. е. $W \subseteq T*$.

В общем случае процессная аналитика начинается с обнаружения процессов [12, 13]. Обнаружение подразумевает формирование модели процесса из журнала событий. Результатом является модель процессов, способная воспроизвести поведение, наблюдаемое в журнале событий.

Для процессной аналитики преимущественно применяются алгоритмы, основанные на классической модели сетей Петри. При использовании сетей Петри для исследования журналов рабочего процесса задачи моделируются переходами, а причинно-следственные зависимости – позициями и дугами.

Формально сеть места/перехода представляет собой кортеж (P, T, F), где:

P – конечное множество мест,

T – конечное множество переходов ($P \cap T = \emptyset$), и

$F \subseteq (P \times T) \cup (T \times P)$ – множество направленных дуг, называемое отношением потока.

Выполнением сети Петри управляют количество и распределение меток. Разметка (или маркировка) – это размещение по позициям сети Петри меток. Маркированная сеть Петри – это пара (N, s), где N = (P, T, F) – сеть Петри, s – мультимножество над P, обозначающее разметку сети.

Сеть Петри выполняется посредством запусков переходов. Переход запускается удалением меток из его входных позиций и образованием новых меток в выходных позициях. Переход $t \in T$ запускается, если он разрешен. Переход разрешен (обозначается $(N, s)[t\rangle$) тогда и только тогда, когда каждая из его входных позиций имеет число меток не меньшее, чем число дуг из позиции в переход, то есть $\bullet t \leq s$.

Сеть Петри, которая моделирует управление рабочим процессом, называют сетью рабочего процесса (Workflow Net, WF-сеть) [14]. Формально WF-сеть определяется следующим образом. Пусть $N = (P, T, F)$ – сеть Петри, а t- – идентификатор, не принадлежащий $P \cup T$. N является сетью рабочего процесса (WF-сетью) тогда и только тогда, когда выполняются условия:

– создание объекта: P содержит входную позицию i такую, что $\bullet i = \emptyset$,

– завершение объекта: P содержит выходную позицию o такую, что

$o\bullet = \emptyset$,

– связность: $N^{-} = (P, T \cup \{t^{-}\}, F \cup \{(o, t^{-}), (t^{-}, i)\})$ сильно связна.

Одним из таких алгоритмов анализа данных, основанным на сетях Петри, является альфа-алгоритм Ван Дер Аалста (Alpha Miner). Альфа-алгоритм представляет собой метод, использующий отношения зависимости между событиями для поиска модели рабочего процесса. Для данного алгоритма на основе журнала определяются так называемые *отношения порядка* . Пусть W – журнал рабочего процесса над T, то есть $W \in P(T*)$. Пусть $a, b \in T$. Тогда можно определить следующие отношения порядка:

1. $a >_W b$ тогда и только тогда, когда существует трассировка $\sigma = t_1 t_2 t_3 \ldots t_{n-1}$ такая, что: $\sigma \in W$, $t_i = a$ и $t_{i+1} = b$, где $i \in \{1, ..., n-2\}$. Отношение $>_W$ описывает, какие задачи появлялись последовательно (одна задача непосредственно следовала за другой в рамках одного прецедента).
2. $a \rightarrow_W b$ тогда и только тогда, когда $a >_W b$ и $b \not>_W a$. Отношение $\rightarrow_W$ означает прямую причинно-следственную связь (одна задача следовала за другой, но не наоборот).
3. $a \#_W b$ тогда и только тогда, когда $a \not>_W b$ и $b \not>_W a$. Отношение $\#_W$ дает пары переходов, которые никогда не следуют друг за другом напрямую, и прямых причинно-следственных связей нет.
4. $a \|_W b$ тогда и только тогда, когда $a >_W b$ и $b >_W a$. Отношение $\|_W$ предполагает потенциальный параллелизм. Если два действия могут следовать друг за другом непосредственно в любом порядке, то они, вероятно, параллельны.

Альфа-алгоритм способен на основе полного журнала рабочего процесса вывести соответствующую модель рабочего процесса. Пусть W – журнал рабочего процесса над T. Тогда алгоритм $\alpha(W)$ определяется следующим образом.

1. $T_w = \{t \in T \mid \exists_{\sigma \in W} t \in \sigma\}$
2. $T_I = \{t \in T \mid \exists_{\sigma \in W} t = first(\sigma)\}$
3. $T_O = \{t \in T \mid \exists_{\sigma \in W} t = last(\sigma)\}$
4. $X_W = \{(A, B) \mid A \subseteq T_W \wedge B \subseteq T_W \wedge \forall_{a \in A} \forall_{b \in B} a \rightarrow_W b \wedge \forall_{a1, a2 \in A} a_1 \#_W a_2 \wedge \forall_{b1, b2 \in B} b_1 \#_W b_2\}$
5. $Y_W = \{(A, B) \in X_W \mid \forall_{(A', B') \in X_W} A \subseteq A' \wedge B \subseteq B' \Rightarrow (A, B) = (A', B')\}$
6. $P_W = \{p_{(A, B)} \mid (A, B) \in Y_W\} \cup \{i_W, o_W\}$
7. $F_W = \{(a, p_{(A, B)}) \mid (A, B) \in Y_W \wedge a \in A\} \cup \{(p_{(A, B)}, b) \mid (A, B) \in Y_W \wedge b \in B\} \cup \{(i_W, t) \mid t \in T_I\} \cup \{(t, o_W) \mid T \in T_O\}$
8. $\alpha(W) = (P_W, T_W, F_W)$ [14]

Алгоритм строит сеть (P_W, T_W, F_W). Набор переходов T_W на 1 этапе алгоритма можно получить, просмотрев журнал. Можно найти все начальные переходы T_I и все конечные переходы T_O.

Добавляются места источника i_W и места стока o_W, а также места вида $p_{(A,B)}$. Для такого места нижний индекс относится к набору входных и выходных переходов, т.е. $\bullet p_{(A,B)} = A$ и $p_{(A,B)}\bullet = B$. Место добавляется между a и b тогда и только тогда, когда $a \rightarrow_W b$.

Некоторые из этих мест должны быть объединены в случае ИЛИ-разделений/соединений. Для этого строятся отношения X_W и Y_W. $(A, B) \in X_W$, если существует причинно-следственное отношение между каждым элементом A и каждым элементом B, и элементы A и B не встречаются рядом друг с другом.

Отношение Y_W выводится из X_W путем включения только наибольших элементов относительно включения множества.

Несмотря на эффективность работы альфа-алгоритма, он имеет ряд ограничений, которые не позволяют применять его в некоторых сферах. Так, он предполагает, что журнал полный, то есть, если действие может непосредственно следовать за другим действием, журнал должен содержать пример такого поведения. Также в журнале не должно быть шума. Однако на практике журналы редко бывают полными или свободными от шума. При работе с шумом первостепенное значение имеет частота, с

которой встречается та или иная трассировка, но альфа-алгоритм не учитывает частоту трассировок в журнале [14].

Поэтому рекомендуется рассмотреть более совершенный алгоритм, такой, как эвристический алгоритм Вейтерса (Heuristic Miner). Данный алгоритм учитывает частоты отношений между задачами. Его отличительной особенностью является то, что при построении модели учитываются частотные характеристики событий в журнале [15, 16]. Алгоритм был разработан с использованием метрики, основанной на частоте, поэтому он менее чувствителен к шуму и неполноте журналов.

В отличие от альфа-алгоритма, журнал рабочего процесса должен включать данные о временной отметке.

Эвристический алгоритм начинается с построения графа зависимостей. Чтобы проверить отношение зависимости между двумя событиями A и B (обозначается $A \Rightarrow_W B$), используется так называемая метрика, основанная на частоте.

Пусть W – журнал событий над T и a, b ∈ T. Тогда $|a >_W b|$ – количество раз, когда $a >_W b$ встречается в W, и

$$a \Rightarrow_W b = \left(\frac{|a >_W b| - |b >_W a|}{|a >_W b| + |b >_W a| + 1}\right) \quad (1)$$

Значение $a \Rightarrow_W b$ всегда находится в диапазоне от -1 до 1. Высокое значение $A \Rightarrow_W B$ свидетельствует о том, что существует зависимость между задачами A и B.

Действия в системе взаимосвязаны: так, каждая неисходная деятельность должна иметь хотя бы одну другую деятельность, являющуюся ее причиной, и каждая неконечная деятельность должна иметь зависимую деятельность. Используя эту информацию в так называемой «эвристике, связанной со всеми действиями», можно выбрать лучшего кандидата с наивысшей оценкой $A \Rightarrow_W B$. Эта эвристика помогает найти надежные причинно-следственные связи, даже если журнал событий содержит шум, и построить корректный граф зависимостей.

Эвристический алгоритм является наиболее подходящим для исследования журнала событий системы управления обучением, поскольку, как уже было сказано, он адаптирован для работы с журналами, содержащими шум.

Применение алгоритмов Process Mining к журналу событий системы дистанционного обучения

Для установления и выявления паттернов используются данные из системы управления курсами Moodle, на которой базируется система дистанционного обучения Костромского государственного университета (СДО КГУ). Данная система содержит собственную базу данных, которая фиксирует действие в системе, его временную отметку, на каком уровне это сделано и есть ли обратная связь от системы или преподавателя [5].

Исследован журнал событий СДО КГУ, представляющий собой таблицу из 23 полей. Ключевыми из них являются следующие:

1. Уровень выполнения действия (курс, задание, форма или ядро).
2. Время фиксации действия (в миллисекундах).
3. Идентификатор студента – обезличенный код в базе данных, по которому можно идентифицировать принадлежность данной записи студенту.
4. Реакция системы – обратная связь от системы или преподавателя на активность студента.

Эти данные позволяют получить и проанализировать информацию о загружаемых и изменяемых заданиях студентов, выяснить, какие задания студент не выполнил, или в какой момент его активность в системе снизилась.

Для анализа данных об активности студентов была использована библиотека pm4py языка программирования Python. С ее помощью исследован журнал событий системы дистанционного обучения университета. На примере данных одного из дистанционных курсов была протестирована работа альфа-алгоритма и эвристического алгоритма. С помощью данных алгоритмов построены сети, визуализирующие действия, выполненные тем или иным студентом при изучении курса. Для визуализации использовались средства библиотеки pm4py: для альфа-алгоритма использовался визуализатор сети Петри pm4py.visualization.petrinet; для эвристического алгоритма сеть строится с помощью объекта визуализатора из pm4py.visualization.heuristics_net.

В ходе исследования выявлено, что альфа-алгоритм обрабатывает многие процессы не так точно, как эвристический алгоритм. В частности, сети альфа-алгоритма часто оставляют «мертвые» части, не

обнаруживая их связи с другими действиями. Это можно заметить на рисунках 2 и 3, где изображены сети одного и того же прецедента.

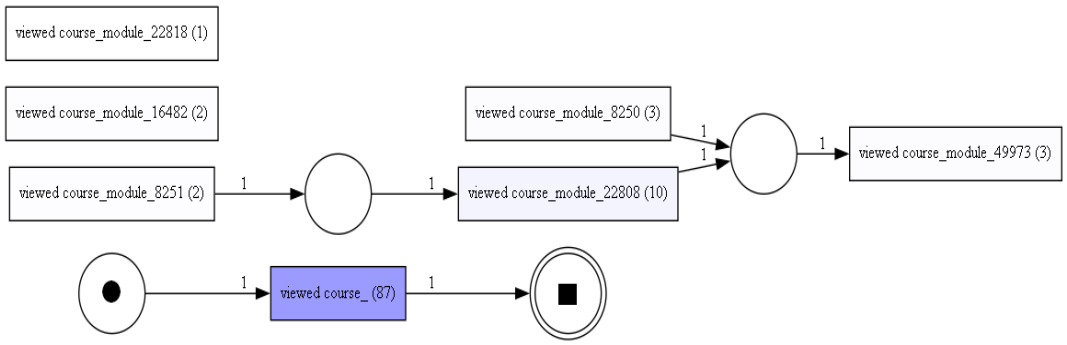

Рисунок 1. Сеть, построенная альфа-алгоритмом, для пользователя с id 1462

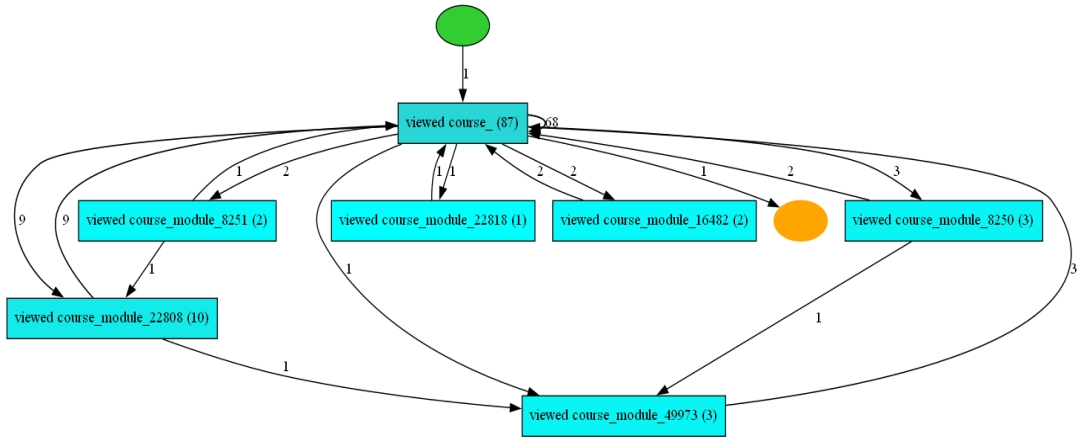

Рисунок 2. Сеть, построенная эвристическим алгоритмом, для пользователя с id 1462

Визуализация сетей позволяет сделать вывод о том, что эвристический алгоритм находит причинно-следственные связи более точно, чем альфа-алгоритм. Возможно, это обусловлено способностью эвристического алгоритма работать с журналами событий, содержащими шум. Данная особенность может являться основанием для выбора эвристического алгоритма для дальнейшего анализа данных в рамках создания системы анализа цифрового следа обучающихся.

Заключение.

Для построения моделей поведения студентов на основе их журналов событий в Moodle выбран эвристический алгоритм, поскольку он в большей степени подходит для исследования журнала событий LMS благодаря своей адаптированности к работе с журналами, содержащими шум. Сети, построенные эвристическим алгоритмом, могут быть обработаны и проанализированы для определения общих закономерностей в поведении студентов.

Выявление поведенческих паттернов студентов позволит в дальнейшем быстрее принимать решения, основываясь на схожих ситуациях в прошлом. Инструмент анализа поведенческих паттернов можно обеспечить пользовательским интерфейсом для более удобного использования. Также следует рассмотреть внедрение иных механизмов анализа данных процесса обучения для получения иной статистики по состоянию обучения.

Список литературы

[1]. Курбацкий, В. Н. Цифровой след в образовательном пространстве как основа трансформации современного университета // «Вышэйшая школа»: навукова-метадычны і публіцыстычны часопіс. № 5. – Минск, 2019. – с. 40-45

[2] Логинова А.А., Денисов А.Р. Актуальные аспекты применения технологии анализа цифрового следа для формирования индивидуального цифрового профиля студента // Преподавание информационных технологий в

Российской Федерации : сборник научных трудов; материалы Девятнадцатой открытой Всеросс. конф. – ООО "1С-Паблишинг", Москва, 2022). – с 17-18.

[3] Learning Management System. Большой обзор LMS-систем: виды, поставщики и реальный кейс внедрения. [Электронный ресурс]. – URL: https://vc.ru/education/218817-bolshoy-obzor-lms-sistem-vidy-postavshchiki-i-realnyy-keys-vnedreniya (дата обращения: 12.09.2022).

[4] Process Mining: знакомство [Электронный ресурс]. – URL: https://habr.com/ru/post/244879/ (дата обращения: 12.09.2022).

[5] Попов М. Д., Логинова А. А., Денисов А. Р. Инструмент выявления паттернов поведения студентов КГУ на основе алгоритмов PROCESS MINING // Технологии и качество. 2022. No 3(57). С. 34–38. https: doi 10.34216/2587-6147-2022-3-57-34-38.

[6] W. Hachicha. Using Process Mining for Learning Resource Recommendation: A Moodle Case Study / W. Hachicha, L. Ghorbel, R. Champagnat, C. A. Zayani, I. Amous // Procedia Computer Science. 2021. No 192. P. 853–862.

[7] Van der Aalst, W. Process mining: Data science in action. Berlin : Heidelberg : Springer-Verlag, 2016. 477 p.

[8] Bogarín A., Cerezo R., Romero C. A survey on educational process mining // Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery. 2018. Vol. 8, no 1. P. 1230–1247.

[9] Bogarín A., Cerezo R., Romero C. Discovering learning processes using Inductive Miner: A case study with Learning Management Systems (LMSs) // Psicothema. 2018. Vol. 30, no 3. P. 322–329.

[10] Galina Deeva, Jochen De Weerdt. Understanding automated feedback in learning processes by mining local patterns. // Business Process Management Workshops. 2018. pp. 56-68 DOI:10.1007/978-3-030-11641-5_5

[11] Wil M.P. van der Aalst, Shengnan Guo, Pierre Gorissen. Comparative Process Mining in Education: An Approach Based on Process Cubes // IFIP International Federation for Information Processing. – 2015. – p. 110-134. DOI:10.1007/978-3-662-46436-6_6

[12] Poohridate Arpasat, Nucharee Premchaiswadi, Parham Porouhan, Wichian Premchaiswadi. Applying Process Mining to Analyze the Behavior of Learners in Online Courses // International Journal of Information and Education Technology, Vol. 11, No. 10. – 2021. – p. 436-443

[13] Awatef Hicheur Cairns, Billel Gueni, Mehdi Fhima, Andrew Cairns, Stéphane David. Process Mining in the Education Domain // International Journal on Advances in Intelligent Systems, vol 8 no 1 & 2. – 2015. – p. 219-232

[14] W.M.P. van der Aalst, A.J.M.M. Weijters, L. Maruster. Workflow Mining: Discovering Process Models from Event Logs // IEEE Transactions on Knowledge and Data Engineering. – 2004. – p.1-42

[15] А. А. Мицюк, И. С. Шугуров. Синтез моделей процессов по журналам событий с шумом, Модел. и анализ информ. систем, 2014, том 21, номер 4. – с. 181–198

[16] A.J.M.M. Weijters, W.M.P. van der Aalst, A.K. Alves de Medeiros. Process Mining with the Heuristics Miner-algorithm // Cirp Annals-manufacturing Technology, 2006. – p.1-35

FEATURES OF APPLYING PROCESS MINING ALGORITHMS FOR STUDENT BEHAVIOR ANALYSIS

A.A. Loginova
Postgraduate student of KSU, assistant of the Department of Information Systems and Technologies of KSU

M.D. Popov
Master student of the Institute of Automated Systems and Technologies, KSU

A.R. Denisov
Professor of the Department of Information Systems and Technologies of KSU, Doctor of Technical Sciences

Department of Information Systems and Technologies
Institute of Automated Systems and Technologies,
Kostroma State University, Russian Federation

Abstract. The problem of analyzing the actions of students in order to implement the functional trajectories of education is considered. It is proposed to analyze the activities of students based on the data of digital investigators who participate in learning management processes. One of the operations of analyzing such data is called process analytics (Process Mining). In the context of this work, the features of Process Mining algorithms and features of the application of algorithms from the point of view of analyzing event logs in a learning management system are considered.

Keywords: Process Mining, Educational Process Mining, digital footprint, Learning Management System