# Proposition d'une architecture utlisant le *trace clustering* pour recommander un parcours d'apprentissage

**Wiem Hachicha** [1,2], **Leila Ghorbel** [1], **Ronan Champagnat** [2], **Mourad Rabah** [2], **Samuel Nowakowski** [3], **Corinne Amel Zayani** [1]

1. *MIRACL - Université de Sfax*
   *Tunis Road Km 10 PB. 242*
   *3021 Sfax, Tunisie*
   *corinne.zayani@fss.usf.tn*
2. *L3i - Université de La Rochelle*
   *Avenue Michel Crépeau*
   *17 042 La Rochelle, France*
   *nom.prenom@univ-lr.fr*
3. *LORIA - Université de Lorraine*
   *Campus Scientifique, 615 rue du jardin-botanique*
   *54 506 Vandœuvre-lès-Nancy, France*
   *samuel.nowakowski@loria.fr*

RÉSUMÉ. *Les systèmes d'informations pédagogiques permettent d'observer les traces d'apprentissage des apprenants et de mener des analyses sur leurs pratiques ou de prédire leur réussite. Dans ces travaux nous étudions comment la fouille de processus, qui permet d'extraire des modèles de comportement des utilisateurs dans un système d'information, peut être utilisée dans un système de recommandation contextuel. Nous nous concentrons plus particulièrement sur le trace clustering qui vise à regrouper des traces possédant des dynamiques proches. Notre apport porte sur la définition d'une architecture pour la recommandation qui utilise le trace clustering. Nous validons notre proposition sur les données collectées d'un cours d'introduction à la programmation d'IHM.*

ABSTRACT. *Educational information systems make it possible to observe learners' learning traces and to carry out analyses of their behaviour. In this work, we study how Process Mining can be used in contextual recommender systems. Our contribution concerns the definition of an architecture for recommendation that uses trace clustering. We validate our proposal on data collected from an introductory course in UI programming.*

MOTS-CLÉS : *SI pédagogique, fouille de processus, recommandation, trace clustering*

KEYWORDS: *Intelligent Tutor System, Process Mining, Recommendation, Trace Clustering*

## 1. Introduction

Les systèmes d'information pédagogiques se sont rapidement développés avec la mise en place d'environnements numériques de travail. La quantité de données produites et de traces laissées par les utilisateurs de ces systèmes offre l'opportunité de fournir des tableaux de bord d'apprentissage et des analyses sur les apprenants (Cordier *et al*., 2013).

La personnalisation des apprentissages est devenue un facteur essentiel pour la réussite des apprenants. Plusieurs projets se sont développés afin de traiter cette problématique. Ces projets s'attaquent au problème de l'offre de formation et de parcours d'apprentissage personnalisé, avec un accompagnement des étudiants dans leur projet de formation et d'insertion professionnelle. Toutefois, peu d'outils sont capables de fournir des indicateurs pertinents afin de recommander des parcours personnalisés ou des actions de remédiation pour les étudiants en difficulté.

La découverte des parcours d'apprentissage reste un défi à relever dans le domaine pédagogique. Toutefois, dans des domaines connexes, on trouve des travaux approchants. En particulier dans le domaine de la fouille de processus des recherches se sont focalisées, à partir de l'analyse des traces d'exécution réelles des utilisateurs (enregistrement des chemins de navigation), sur l'extraction de connaissances sur le parcours de l'utilisateur (Leblay *et al*., 2018).

Le parcours d'apprentissage revient à sélectionner et ordonner les activités à réaliser pour acquérir des connaissances et compétences. Ce parcours d'apprentissage correspond à un scénario pédagogique suivi par un apprenant. Il peut être modélisé par un processus métier. Ce type de processus possède la particularité d'être faiblement structuré. C'est-à-dire que l'utilisateur dispose de degrés de liberté importants. L'objectif est alors de déterminer pour chaque utilisateur ou par groupe d'utilisateurs le comportement adopté lors de l'utilisation du système.

Notre objectif est d'étudier la possibilité de définir une méthodologie de recommandation basée sur les processus extraits des traces utilisateurs et d'implémenter l'architecture logicielle correspondante. Pour atteindre cet objectif, nous proposons de construire un modèle utilisant la fouille de processus à partir des observations recueillies lors des expériences des utilisateurs précédents avec le système d'information pédagogique. Ce modèle représente l'enchaînement des étapes et leurs impacts sur les états du processus global et sera par la suite utilisé pour recommander l'étape la plus appropriée pour guider l'utilisateur ou l'apprenant actuel. Un premier défi, contribution présentée dans le présent article, consiste à classer les différentes trajectoires que nous aurons identifiées au sein de systèmes d'activité sélectionnés dans un contexte d'apprentissage, afin de pouvoir ensuite déterminer, de manière automatisée, à quelle trajectoire type correspond le parcours et le développement d'un apprenant dans un environnement numérique dépourvu de processus métiers bien identifiés.

Notre approche suggère une voie corrective, si nécessaire. Dans (Ho *et al*., 2016), les auteurs ont introduit une méthode pour amener les utilisateurs à prendre les bonnes

décisions en fonction de certaines informations extraites d'un modèle de données qui décrit les activités possibles à réaliser pour atteindre l'objectif et leur impact. Ce modèle de données est une entrée de la méthode, dans le sens où il est construit a priori par des experts du domaine. Dans notre cas, notre approche calcule le modèle de données à partir des informations disponibles avant de recommander l'étape suivante.

Dans la section suivante nous présentons le domaine de la fouille de processus. Nous décrivons le principe, les contraintes, les techniques de fouilles et les critères de qualité pour estimer la pertinence du modèle extrait. Nous terminons cette section par une présentation du *trace clustering*.

Nous faisons, ensuite, un état de l'art de l'utilisation de la fouille de processus dans le domaine de l'éducation. Puis nous proposons notre architecture qui s'appuie sur le *trace clustering* pour classer les parcours d'apprentissage. Nous validons cette architecture sur les données d'un cours universitaire de programmation des IHM.

## 2. Fouille de processus

La fouille de processus est utilisée dans de nombreux domaines, notamment pour la modélisation des comportements des utlisateurs qui recherchent des informations dans une bibliothèque numérique (Trabelsi *et al*., 2019a ; 2019b), dans le domaine médical afin de cartographier les processus healthcare (Pika *et al*., 2019), ou dans les réseaux sociaux pour modéliser les parcours utilisateurs (Li, De Carvalho, 2019)... L'objectif de la fouille de processus est de découvrir, superviser et améliorer des processus métier existants en extrayant de la connaissance à partir des journaux d'événements facilement disponibles dans les systèmes d'information actuels.

Chaque événement dans un tel journal fait référence à une activité (une étape bien définie d'un processus) et est lié à un cas particulier (une instance de processus). Les techniques de fouille de processus utilisent des informations supplémentaires telles que la ressource (une personne ou un équipement) qui exécute ou lance l'activité, l'horodatage de l'événement ou des éléments de données enregistrés avec l'événement (par exemple, la quantité commandée).

Les techniques de fouille de processus sont apparues au cours des années 1990. (Cook, Wolf, 1998) et (Agrawal *et al*., 1998) ont proposé des algorithmes de découverte de modèle de processus afin de pouvoir analyser une organisation ou comparer des exécutions de processus métier à partir des traces observées dans le sysètme d'information. (Van der Aalst, 2016) a popularisé la fouille de processus et développé de nombreux algorithmes.

Une première utilisation de la fouille de processus consiste à découvrir un modèle de processus. Un algorithme de découverte analyse un journal d'événements et construit un modèle à partir des relations de précédences entre activités. Parmi les algorithmes de découverte nous pouvons citer Inductive Miner, Fuzzy Miner, Heuristic Miner et Alpha Miner. La sortie de ces algorithmes est un modèle tel que les réseaux de Petri, BPMN...

Une deuxième utilisation de la fouille de processus est la vérification de la conformité. Il s'agit ici de comparer un modèle existant avec les journaux. L'objectif est de déterminer si la réalité enregistrée dans les journaux d'événements est conforme vis-à-vis du modèle et réciproquement (le modèle est conforme vis-à-vis de journaux observés).

### *2.1. Journaux d'événements*

Un journal d'événements contient un ensemble d'événements. Chaque événement correspond à la réalisation d'une activité. Les informations minimales afin d'extraire des connaissances en utilisant la fouille de processus sont : un identifiant permettant de rattacher l'activité à un processus (CaseID), un identifiant de l'activité (Activity) et un horodatage (timestamp) (Van der Aalst, 2016).

Chaque variant d'un processus est décrit par une séquence d'activités (par exemple $\langle Activity1, Activity2, Activity1\rangle$). L'ensemble du journal d'événements peut s'écrire sous la forme $L = [\langle Activity1,\ Activity2,\ Activity1\rangle^2,\ \langle Activity1,\ Activity2\ , Activity3\rangle^1, \langle Activity1, Activity1\rangle^3]$ où la multiplicité des séquences donne le nombre de fois que ce variant est présent dans le journal.

### *2.2. Algorithmes de découverte*

Les algorithmes de découverte visent à extraire des modèles de processus à partir des informations contenues dans les journaux d'événements. Ils se concentrent sur les aspects contrôle. Les techniques de découvertes sont variées, mais toutes sont bâties sur les relations entre les activités dans les journaux d'événements.

Les algorithmes observent en particulier les relations de causalité entre deux activités. Ils se basent sur l'intuition que si une activité se trouve toujours après une autre, c'est qu'il y a certainement une relation de causalité entre les deux. Entre deux acitvités, $a_1$ et $a_2$, quatre relations sont considérées :

1. *Succession directe*, $a_1 > a_2$ s'il existe un variant tel que $a_1$ est immédiatement suivie par $a_2$ ;
2. *Causalité*, $a_1 \rightarrow a_2$, si $a_1 > a_2$ et $a_2 \not> a_1$ ;
3. *Parallèle*, $a_1 \| a_2$, si $a_1 > a_2$ et $a_2 > a_1$ ;
4. *Choix*, $a_1 \# a_2$, si $a_1 \not> a_2$ et $a_2 \not> a_1$.

L'algorithme $\alpha$ (Aalst *et al.*, 2004) construit un raisonnement sur ces quatre relations pour déduire un réseau de Petri qui modélise les processus. D'autres algorithmes, tel *Heuristic Miner* (Weijters, Ribeiro, 2011) se basent sur le graphe des successions directes observées dans les journaux. *Heuristic Miner* vise à résoudre le problème de log contenant du bruit et obtenus à partir de processus faiblement structurés. Cet alo-

grithme utilise une métrique basée sur la fréquence afin de déterminer la confiance dans la relation de causalité entre deux acitvités, $a_1$ et $a_2$ calculée comme suit :

$$a_1 \Rightarrow a_2 = \left( \frac{|a_1 > a_2| - |a_2 > a_1|}{|a_1 > a_2| + |a_2 > a_1| + 1} \right)$$

Une autre stratégie de découverte, utilisée par *Fuzzy Miner* (Van der Aalst, 2016), consiste à travailler sur la visualisation d'un graphe en s'inspirant de ce qui est fait pour les cartes routières et en donnant des règles de zoom et d'agrégations de chemin en fonction de leur popularité.

### *2.3. Qualité des modèles découverts*

L'objectif de la fouille de processus est de découvrir des modèles qui caractérisent la dynamique d'un système d'information à partir des traces. Plusieurs mesures ont été proposées afin de caractériser la qualité du modèle découvert. Ces mesures font appel aux notions d'*overfitting*, le fait qu'un modèle ne représente que les variants présents dans les logs (un nouveau variant ne sera pas représenté par le modèle), et d'*underfitting*, le fait qu'un modèle représente potentiellement beaucoup (trop) de variants.

Quatre mesures sont définies par la communauté (Buijs *et al.*, 2012) :

– *Fitness* : représente la capacité à expliquer les processus observés ;

– *Generalisation* : caractérise le fait que le modèle est capable de prendre en compte de nouveaux variants ;

– *Precision* : caractérise le fait que le modèle n'est pas trop général, c'est-à-dire qu'il ne prend pas en compte tous les variants ;

– *Simplicity* : caractérise la complexité et les spécificitées du modèle.

Ces mesures visent à déterminer la capacité d'un modèle à expliquer et généraliser les dynamiques observées dans les journaux. Un bon modèle doit trouver un équilibre entre ces mesures.

Pour chaque mesure plusieurs critères ont été proposés. Nous utiliserons ceux définis par (Adriansyah, 2014) qui se basent sur la notion d'alignement entre une trace observée dans les log et une trace d'exécution du modèle. Pour chaque trace on totalise le nombre de transformations à faire pour passer de l'une à l'autre. La *Fitness* est calculée comme suit :

$$F(L, M) = 1 - \frac{\delta(\lambda^M_{opt}(L))}{\delta(\lambda^M_{worst}(L))}$$

Où $\delta$ est la fonction de coût, $\lambda^M_{worst}(L)$ est le cas le plus défavorable où il n'y a aucune synchronisation possible entre la trace et le modèle. $\lambda^M_{opt}(L)$ représente les coûts obtenus pour chaque alignement optimal.

La précision est calculée par :

$$P(T, M) = \frac{1}{|E|} \sum_{e \in E} \frac{en_T(e)}{en_M(e)}$$

Où $E$ est l'ensemble des événements dans les logs $T$, $A$ l'ensemble des activités, $en_T(e) \subseteq A$ est l'ensemble des activités présentes dans les traces et $en_M(e) \subseteq A$ l'ensemble des activités présentes dans le modèle.

### *2.4. Trace clustering*

La fouille de processus est apparue dans le domaine des processus métiers. Ces processus sont généralement bien définis et cadrés au niveau du système d'information. Or, dans le cas de système d'information pédagogique, une partie du processus métier est entre les mains de l'apprenant. Nous avons donc des processus faiblement ou partiellement structurés. L'utilisation des algortimes de découverte dans un tel contexte amène à des modèles complexes difficilement exploitables.

Les travaux de (Trabelsi *et al*., 2021) montrent que l'utilisation du *trace clustering* aboutit à l'extraction de connaissance et permet d'identifier des dynamiques d'usages typiques en réduisant la complexité des modèles découverts. Le but du *trace clustering* est de regrouper des variants de scénarios d'apprentissage qui possèdent des caractéristiques, au niveau du processus décrit, similaires.

Le *trace clustering* consiste à regrouper les traces avant d'appliquer un algorithme de découverte (Diamantini *et al*., 2016). Quatre approches ont été développées :

– le *trace-based clustering* regroupe les traces en fonction de leur similarité syntaxique. La mesure de similarité est inspirée de la distance de Levenshtein ;

– le *feature-based clustering* calcule un vecteur en fonction des caractéristiques de chaque trace. Parmi les caractérisitques couramment retenues, il y a la fréquence d'une activité dans un variant, la fréquence de succession directes, les sous-séquences maximales, les sous-séquences fréquentes...

– le *model-based clustering* qui réalise le regroupement sur les qualités des modèles minés, et

– l'*hybrid-based clustering* qui combine les approches précédentes (Song *et al*., 2008 ; Zandkarimi *et al*., 2020).

Nous venons de présenter le domaine de la fouille de processus en terminant par une introduction au *trace clustering*. Il s'agit de méthodes visant à regrouper des traces d'exécutions qui décrivent des comportements similaires. Dans la section suivante, nous présentons un état de l'art des approches utilisant la fouille de processus dans un contexte d'apprentissage.

## 3. État de l'art

Il existe de nombreuses techniques d'analyse de données issues de systèmes d'information pédaogiques telles que l'Educational Data Mining (EDM) (Berland *et al.*, 2014) et l'Educational Process Mining (EPM) (Romero, Ventura, 2013). L'Educational Process Mining permet de découvrir des modèles de processus d'apprentissage à partir de journaux d'événements à des fins différentes telles que la prédiction des performances, l'adaptation et la recommandation des étudiants.

La fouille de processus a été également utilisée pour analyser les parcours d'apprentissage des étudiants.

(Beemt *et al.*, 2018) explorent la relation entre le comportement d'apprentissage et les progrès d'apprentissage dans les MOOC afin de mieux comprendre comment les étudiants qui réussissent et ceux qui échouent répartissent leurs activités au cours des semaines de cours. Pour trouver les modèles de comportement d'apprentissage des étudiants dans le MOOC, une analyse des trajectoires d'apprentissage est réalisée en utilisant la fouille de procesus. Le clustering hiérarchique est utilisé afin d'extraire la connaissance. Ils ont trouvé quatre groupes d'étudiants significatifs, chacun représentant un comportement spécifique allant du simple début à la fin du cours. Les techniques d'exploration de processus montrent que les étudiants qui réussissent affichent un comportement d'apprentissage plus régulier.

Dans (Romero *et al.*, 2008), une tâche de prétraitement est effectuée pour regrouper les utilisateurs en fonction de leur type d'interactions avec le cours. Cette étude permet de découvrir les comportements de navigation les plus spécifiques en utilisant uniquement les données groupées plutôt que le jeu de données complet en appliquant l'algorithme *Heuristic Miner*.

(Real *et al.*, 2020) ont présenté les résultats de l'utilisation de techniques de fouille de processus pour valider les parcours d'apprentissage des étudiants dans un cours d'introduction à la programmation. Ils ont utilisé un journal d'événements Moodle contenant 24605 événements soumis par 73 étudiants de premier cycle. Les résultats ont révélé que, dans l'ensemble, les étudiants qui ont réussi et ceux qui ont échoué ont emprunté des chemins différents pour réaliser les activités du cours. Ils ont également obtenu les flux de contrôle et les fréquences des activités et des connexions pour identifier les dépendances et les ressources qui ont démarré ou terminé le processus. L'analyse de ces résultats fournit des informations générales et spécifiques sur les parcours d'apprentissage des étudiants, ainsi que la possibilité pour les enseignants d'observer les comportements et les progrès des étudiants.

(Martinez *et al.*, 2021) a examiné les trajectoires d'apprentissage des étudiants dans leurs études d'informatique. L'étude s'est concentrée sur la modélisation des caractéristiques qui influencent les taux d'abandon. Par conséquent, les trajectoires des étudiants ayant abandonné leurs études et de ceux qui les ont terminées sont analysées et comparées à l'aide d'outils de fouille de processus. Les auteurs ont constaté que les cours jugés difficiles et entravant la progression académique ont été identifiés, tout

comme les derniers cours suivis par les étudiants avant l'abandon ou l'obtention du diplôme. Ils ont également constaté que les étudiants en décrochage terminent généralement après la première année et suivent des cours de programmation.

Enfin, nous présentons des travaux qui utilisent la fouille de processus avec un objectif de recommandation d'activités pédagogiques.

En utilisant la fonctionnalité de journalisation expérimentale de l'outil de modélisation JMermaid et les techniques de fouille de processus, (Sedrakyan *et al.*, 2014) analyse le modèle de comportement (données d'événements de modélisation conceptuelle de 20 cas et 10.000 événements). Les résultats de ce travail comprennent des modèles qui indiquent une performance d'apprentissage pire/meilleure. Les auteurs soutiennent que les résultats aident à améliorer les conseils d'enseignement pour la modélisation conceptuelle visant à fournir un feedback orienté vers le processus et fournissent des recommandations sur le type de données qui peuvent être utiles pour observer le comportement de modélisation du point de vue des résultats d'apprentissage.

Dans (Leblay *et al.*, 2018), les auteurs ont cherché à proposer une méthode d'assistance basée sur le parcours d'apprentissage pour aider les apprenants à construire leur parcours universitaire. Ils utilisent la découverte de processus pour extraire les parcours d'apprentissage des apprenants précédents, puis utilisent ce modèle pour recommander l'étape la plus appropriée pour guider l'utilisateur ou l'apprenant actuel.

La fouille de processus a été utilisée pour expliquer des comportements, analyser des parcours d'apprentissage et également avec un objectif de recommandation. Ces derniers, travaux sur la recommandation, se basent sur l'analyse du modèle global d'apprentissage. Aucun travail n'aborde la recommandation à travers le *trace clustering* utilisé pour classer les scénarios d'apprentissage.

Dans la section suivante, nous proposons une architecture exploitant le *trace clustering* pour la recommandation de parcours pédagogique.

## 4. Architecture pour la recommandation

Notre architecture se base sur les travaux de (Ghorbel *et al.*, 2015) qui traitent de l'adaptation de contenu pédagogiques pour la personnalisation des enseignements. Cette architecture est organisée en quatre couches : *source* (qui stocke les objets d'apprentissage et les données de profil), *interoperability* (qui gère l'hétérogénéité des données et propose des mécanismes de fusion), *adaptation* (qui propose une adaptation du sénario pédagogique) et *client* (chargée des interactions utilisateur).

Cette architecture permet : i) l'échange de données entre des profils d'apprenants hétérogènes sur la base d'une couche d'interopérabilité et ii) d'adapter aux apprenants la navigation dans les ressources d'apprentissage sur la base d'une couche d'adaptation. Dans (Hachicha *et al.*, 2021), nous étendons cette architecture avec deux autres couches qui utilisent la fouille de processus pour la recommandation. Nous étendons

cette architecture en ajoutant, au niveau de la couche fouille de processus, une étape de *trace clustering*.

L'architecture que nous proposons (Figure 1) est composée de quatre couches : client, recommandation, fouille de processus et source. Nous introduisons brievement les 2 premières et présentons plus en détail les 2 dernières, au coeur de la présente communication.

### *4.1. La couche client*

La couche client permet l'interaction entre l'apprenant et les systèmes d'apprentissage en ligne (LMS). Ainsi, l'apprenant peut envoyer une requête en cliquant sur les liens fournis à travers différents types de dispositifs (PC, mobile...).

### *4.2. La couche recommandation*

La recommandation se fait en cours d'exécution et consiste à déterminer, à partir du modèle des processus et de la séquence réalisée jusque là par l'utilisateur, quelle activité doit être réalisée pour finir le processus en optimisant des critères préalablement définis. Les mécanismes de recommendation mis en place sont en dehors de la portée du présent article

### *4.3. La couche source*

La couche source contient des bases de données distribuées de journaux d'événements, de modèles de processus, de profils globaux et de ressources d'apprentissage. Les journaux d'événements (*Event Logs*) sont des fichiers qui contiennent une grande quantité de données brutes sur l'interaction des apprenants avec le LMS. Comme les données dans les journaux d'événements sont souvent bruités, cette base de données nécessite une étape de prétraitement. Cette couche contient toutes les traces, laissées par l'utilisateur, que nous allons exploiter afin d'extraire son scénario d'apprentissage.

La base de données de *Global Profil* enregistre un profil pour chaque apprenant qui contient une vue globale de ses données qui sont distribuées dans les différents systèmes d'apprentissage en ligne. La construction du profil global est une étape préliminaire de la couche d'interopérabilité détaillée dans (Troudi *et al*., 2020). Le profil contient plusieurs caractéristiques : des données personnelles, des données démographiques et des données sociales qui caractérisent les personnes connues (amis) et les intérêts (les intérêts de l'apprenant et ceux de ses amis). Les intérêts représentent les objets d'apprentissage sur lequel l'apprenant prend plaisir à passer du temps à apprendre. Nous étendons ce profil avec des informations sur le type de situation d'apprentissage (à distance, en face à face ou hybride), l'aspect social (travail individuel ou collaboratif) et le style d'apprentissage (aspect théorique ou pratique).

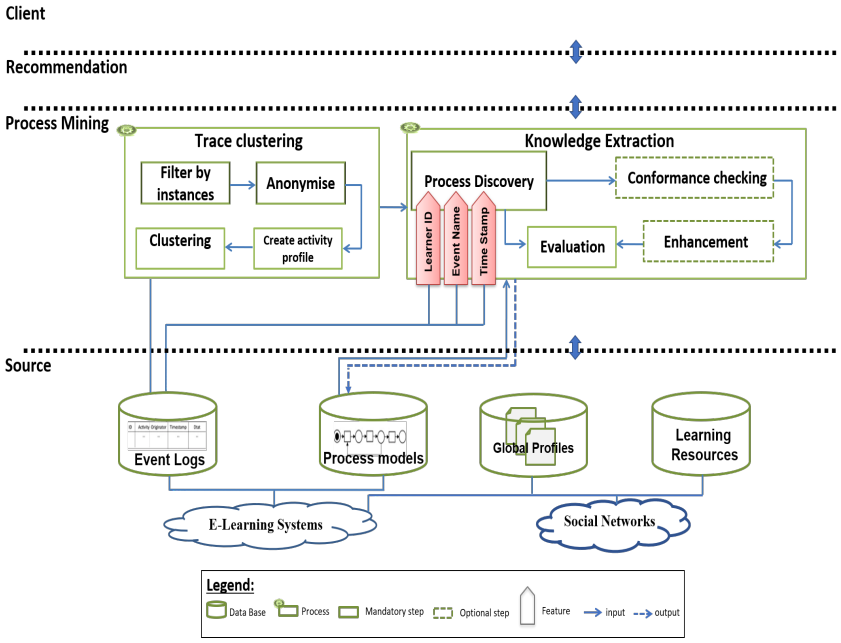


FIGURE 1. *Architecture logicielle pour la recommandation d'activités basée sur le trace clustering*

La base de données des ressources d'apprentissage, *Learning Resources* contient tout élément impliqué dans le processus d'apprentissage tel que les cours, quiz, pages web, images, vidéos...

### *4.4. La couche fouille de processus*

La couche de fouille de processus permet de découvrir des modèles de processus d'apprentissage basés sur les journaux d'événements. Elle comprend une étape de *trace clustering* afin d'identifier des groupes homogènes de modèles d'apprenants (instances de processus possédant des caractéristiques proches). Nous nous sommes basés sur l'approche *feature-based clustering* en considérant la fréquence des activités par variant.

Notre apport porte sur l'ajout de cette étape de *trace clustering*. Nous cherchons à regrouper les étudiants en fonction de leurs scénarios d'apprentissage préférentiels avant d'extraire des informations qui nous aiderons pour la couche de recommandation.

Comme le montre Figure 1, cette étape commence d'abord par le filtrage du journal des événements par instances pour nous assurer que seules les activités des apprenants

soient conservées. Nous éliminons ainsi le bruit (activités d'administration du cours Moodle), les *outliers*, et les traces incomplètes (abandons en cours de formation)

Nous avons ensuite créé le profil d'activité qui est utilisé pour diviser le journal des événements. Le profil d'activité est une matrice $N \times M$, où $N$ représente le nombre d'apprenants et $M$ le nombre d'activités. Chaque ligne de cette matrice correspond à une trace vectorielle qui est composée de fréquences d'activités.

Enfin, nous appliquons un algorithme de *clustering* pour diviser les traces en fonction du profil d'activité. Afin de sélectionner l'algorithme de *clustering* le plus adapté à notre cas nous avons réalisé des essais avec *DBSCAN*, *Agglomerative Clustering*, *Gaussian Mixture Model* et *k-means*. Le détail est donné dans la section suivante (5).

Les résultats du *trace clustering* constituent l'entrée de l'algorithme de découverte des processus. Ils sont constitués du scénario pédagogique suivi par les étudiants d'un même regroupement. Il est alors possible d'appliquer les algorithmes de découvertes de processus afin d'obtenir le modèle de scénario pédagogique pour un groupe.

Chaque modèle de processus montre le comportement d'utilisation le plus courant des apprenants dans un LMS. L'analyse du modèle de processus permet de visualiser et de reproduire le comportement réel de l'apprenant, de trouver des *patterns* dans le comportement d'apprentissage des apprenants, et plus encore de proposer une explication sur le scénario d'apprentissage et ainsi de la recommandation faite.

## 5. Validation

Afin de valider notre architecture nous avons extrait les journaux d'événements des apprenants ayant suivi le cours « Introduction aux interfaces homme-machine (IHM) » créé sur la plateforme Moodle à l'Université de La Rochelle (France). Un scénario pédagogique a été établi permettant aux apprenants d'atteindre l'objectif final. Les journaux d'événements comprennent 42 438 événements de 100 étudiants (c'est-à-dire 100 traces) ayant suivi le cours pendant un semestre. Chaque événement correspond à une activité réalisée par un apprenant.

Nous nous concentrons, dans cet article sur l'étape de regroupement des traces pour déterminer s'il est possible d'obtenir des ensembles de scénarios pédagogiques pertinents pour la recommandation.

Afin d'extraire des clusters de traces homogènes, nous avons créé le profil d'activité, puis nous avons appliqué l'algorithme de clustering. Dans notre cas particulier, le profil d'activité est une matrice $100 \times 28$, où $100$ représente le nombre d'apprenants et $28$ représente le nombre d'activités (par exemple *Course module viewed* ou *Quiz attempt started*). Nous avons calculé le nombre d'occurrences de chaque activité dans chaque trace.

Nous avons effectué plusieurs expériences avec différents algorithmes de clustering tels que *DBSCAN* (Ester *et al.*, 1996), *Agglomerative Clustering* (Müllner, 2011), *Gaussian Mixture Model (GMM)* et *k-means* (Hachicha *et al.*, 2023).

Pour découvrir le modèle de processus pour chaque cluster, nous avons appliqué l'algorithme Heuristique Miner, car il a donné de meilleurs résultats dans nos travaux précédents (Hachicha *et al.*, 2021). Un exemple est donné par Figure 2 pour le cluster C2.

Nous appliquons les mesures de qualité classiques (Fitness (F), Précision (P) et Généralisation (G)) sur les modèles découverts.

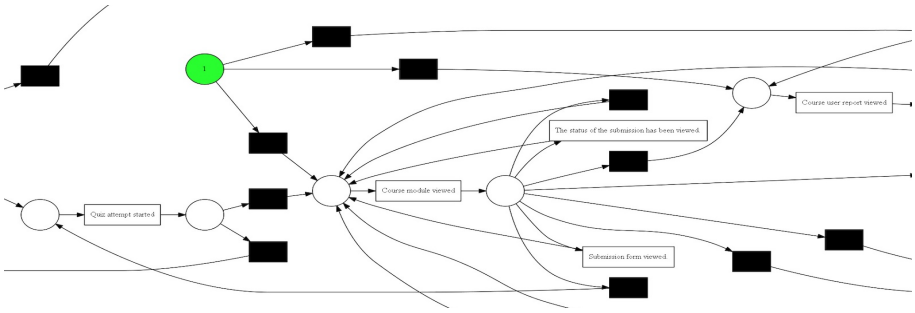


FIGURE 2. *Extrait du modèle de parcours d'apprentissage découvert pour le cluster C2*

Nous avons fixé le nombre de clusters à 5 après une étude expérimentale (*Elbow method*). Table 1 présente les résultats des mesures après l'application des algorithmes de clustering [1] *k-means*, *DBSCAN*, *Agglomerative Clustering* et *Gaussian Mixture Model* respectivement. Les cinq clusters sont identifiés par les labels C0, C1, C2, C3 et C4. Les cluster différent en fonction des activités des apprenants. Dans C3 le comportement décrit correspond uniquement à utiliser moodle pour effectuer les quiz. Dans C2 les étudiants vont visualiser le cours avant de faire le quiz. Le cluster C4 correspond à des étudiants qui en plus ont téléchargé des archives disponibles sur la page moodle...

Puis, nous avons calculé la moyenne de chaque métrique afin d'identifier l'algorithme de clustering le plus performant. Nous avons utilisé pour cela la bibliothèque PM4Py [2] qui implémente le calcule basé sur la notion d'alignement. La moyenne de chaque métrique mesurée est abrégée par Avg.

Les résultats ont révélé que les modèles de processus découverts basés sur les clusters générés par *Gaussian Misxture Model* sont les meilleurs comparés aux autres algorithmes. Nous avons obtenu $0{,}9837$ pour la valeur de Fitness, ce qui implique que le modèle rend mieux compte des comportements présents dans le journal des événements. En outre, nous avons enregistré la valeur la plus élevée en termes de précision, ce qui indique que le taux d'activités dans les journaux d'événements est de $0{,}3489$ par rapport au total des activités détectées dans le modèle de processus. Nous

1. Les implémentations des calculs de clustering ont été faites avec la bibliothèque scikit-learn
2. https://pm4py.fit.fraunhofer.de/

avons enregistré $0{,}7692$ pour la valeur de Généralisation, ce qui confirme la capacité du modèle à généraliser les comportements présents dans le journal d'événements.

TABLEAU 1. *Critères de qualité obtenus après utilisation de diférents algorithmes de clustering*

| | k-means | | | DBSCAN | | | Agglomerative | | | GMM | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | F | P | G | F | P | G | F | P | G | F | P | G |
| C0 | 0.9922 | 0.1951 | 0.7718 | 0.9887 | 0.1441 | 0.7852 | 0.9892 | 0.2037 | 0.8019 | 0.9862 | 0.2393 | 0.8429 |
| C1 | 0.9847 | 0.2218 | 0.8235 | 0.9841 | 0.2479 | 0.8554 | 0.9796 | 0.2960 | 0.7447 | 0.9799 | 0.1972 | 0.7597 |
| C2 | 0.9652 | 0.1624 | 0.6668 | 0.9865 | 0.1909 | 0.6172 | 0.9923 | 0.1485 | 0.7963 | 0.9652 | 0.1624 | 0.6668 |
| C3 | 0.9838 | 0.2884 | 0.7663 | 0.9332 | 1 | 0.7836 | 0.9652 | 0.1624 | 0.6668 | 0.9923 | 0.1459 | 0.7607 |
| C4 | 0.9923 | 0.1486 | 0.7938 | 0.9802 | 0.1586 | 0.6003 | 0.9909 | 0.1623 | 0.7553 | 0.9949 | 1 | 0.8162 |
| Avg | 0.9836 | 0.2023 | 0.7644 | 0.9745 | 0.3483 | 0.7283 | 0.9834 | 0.1945 | 0.7530 | **0.9837** | **0.3489** | **0.7692** |

TABLEAU 2. *Comparaison des modèles découverts en utilisant le trace clustering et sur l'ensemble des traces*

| Modèles de processus obtenus | F | P | G |
|---|---|---|---|
| sans utiliser le trace clustering (Hachicha *et al*., 2021) | 0,9903 | 0,1596 | 0,7611 |
| en utilisant le *trace clustering* | 0,9837 | **0,3489** | **0,7692** |

Afin de valider l'apport du *trace clustering*, nous avons effectué une comparaison entre les modèles de processus découverts sans appliquer le clustering et avec. Pour faire la comparaison, nous utilisons les critères de qualité classiques (Fitness, Précision et Généralisation). Table 2 montre que les modèles obtenus en utilisant le *trace clustering* sont de meilleure qualité. En effet, la valeur de Précision passe de $0{,}1596$ à $0{,}3489$ et la valeur de Généralisation augmente légèrement. Même si la valeur de Fitness diminue légèrement, notre expérimentation montre que le *trace clustering* nous permet d'extraire des modèles plus pertinents. Modèles qui nous permettront de produire des recommandations plus précises.

## 6. Conclusion

Dans le cadre de l'élaboration d'un outil d'aide à la construction d'un parcours personnalisé, nous visons à développer un outil qui recommande en se basant sur la fouille de processus. Nous avons présenté le domaine de la fouille de processus. Puis nous avons proposé un état de l'art des travaux utilisant la fouille de processus pour l'apprentissage. Nous n'avons pas trouvé de travaux utilisant le *trace clustering* pour la recommandation dans le domaine de la formation.

Le *trace clustering* vise à regrouper des traces d'exécutions qui possèdent des dynamiques proches. Nous avons proposé une architecture pour la recommandation qui utilise le *feature based clustering*. Nous avons validé cette architecture à partir de données issues d'un cours d'introduction à la programmation d'IHM comportant $42438$ événements. Nous avons montré que nous pouvons déterminer des regroupements pertinents.

L'intérêt de passer par le *trace clustering* est que l'on détermine à quelle catégorie de parcours d'apprentissge un apprenant se rattache. Pour ce parcours nous pouvons

extraire un modèle et ainsi nous pouvons fournir des éléments d'explication de la proposition de recommandation faite.

Les perspectives de ces travaux portent sur la définition d'algorithmes de recommandation qui seront destinés à améliorer l'efficacité du processus de suivi et d'anticipation des actions en lien avec un objectif déclaré ou estimé. Nous visons à développer une plateforme numérique définissant un espace facilitateur et fédérateur d'engagement installant une dynamique nouvelle de l'occupation de l'espace d'apprentissage et d'enseignement et intégrant la gestion personnalisée ou collaborative des ressources d'apprentissage. Nous visons à favoriser l'accessibilité, la continuité et la porosité de l'apprentissage en s'appuyant sur la recommandation des ressources adaptées à un projet personnel ou professionnel d'apprentissage réalisés sur un territoire.

La méthode que nous proposons, qui consiste à considérer le parcours d'un apprennant comme un processus, puis à utiliser les traces laissées par l'apprenant pour regrouper et découvrir les modèles de comportement pour recommander la prochaine action afin d'atteindre un objectif, peut être utilisée dans différents contextes. Par exemple celui des bibliothèques numériques où il est possible de caractériser la navigation de l'utilisateur afin d'adapter le contenu proposé.

Ce besoin de caratériser la navigation de l'utilisateur se retrouve également au niveau des sites web. Nous nous rapprochons alors du *Web Usage Mining*, cependant nous proposons un modèle de navigation qui permet de construire un raisonnement explicable pour la recommandation. Notre approche trouverait tout son intérêt pour des environnements où l'utilisateur déroule son propre processus (banque, assurance, administration...).

*Remerciements*

*Ce travail a été soutenu financièrement par le programme PHC Utique du Ministère français des Affaires étrangères et du Ministère de l'Enseignement supérieur et de la recherche et par le Ministère tunisien de l'Enseignement supérieur et de la recherche scientifique dans le cadre du projet CMCU numéro 22G1403.*

## Bibliographie

Aalst W. Van der, Weijters A., Măruşter L. (2004). Workflow mining: Discovering process models from event logs. *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, p. 1128–1142.

Adriansyah A. (2014). *Aligning observed and modeled behavior*. Thèse de doctorat non publiée, Technische Universiteit Eindhoven.

Agrawal R., Gunopulos D., Leymann F. (1998). Mining process models from workflow logs. In H.-J. Schek, G. Alonso, F. Saltor, I. Ramos (Eds.), *Advances in database technology — edbt'98*, p. 467–483. Berlin, Heidelberg, Springer Berlin Heidelberg.

Beemt A. van den, Buijs J., van der Aalst W. (2018). Analysing structured learning behaviour in massive open online courses (moocs): An approach based on process mining and clustering. *International Review of Research in Open and Distributed Learning*, vol. 19, n° 5, p. 37–60.

Berland M., Baker R., Blikstein P. (2014). Educational data mining and learning analytics: Applications to constructionist research. *Technology, Knowledge and Learning*, vol. 19, p. 205–220.

Buijs J. C., Van Dongen B. F., van Der Aalst W. M. (2012). On the role of fitness, precision, generalization and simplicity in process discovery. In *Otm confederated international conferences*, p. 305–322.

Cook J. E., Wolf A. L. (1998). Discovering models of software processes from event-based data. *ACM Trans. Softw. Eng. Methodol.*, vol. 7, p. 215-249.

Cordier A., Lefevre M., Champin P.-A., Georgeon O., Mille A. (2013). Trace-Based Reasoning - Modeling interaction traces for reasoning on experiences. In P. McCarthy (Ed.), *The 26th International FLAIRS Conference*, p. 1-15. St. Pete Beach, Florida, United States. Consulté sur https://hal.archives-ouvertes.fr/hal-00830444

Diamantini C., Genga L., Potena D. (2016). Behavioral process mining for unstructured processes. *Journal of Intelligent Information Systems*, vol. 47, n° 1, p. 5–32.

Ester M., Kriegel H.-P., Sander J., Xu X. *et al*. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Second international conference on knowledge discovery and data mining*, vol. 96, p. 226–231.

Ghorbel L., Zayani C., Amous I. (2015). Improve the adaptation navigation in educational cross-systems. *Procedia Computer Science*, vol. 60, p. 662-670.

Hachicha W., Ghorbel L., Champagnat R., Zayani C. A. (2023). Trace clustering based on activity profile for process discovery in education. In *Intelligent systems design and applications: 22th international conference on intelligent systems design and applications (isda 2022) held december 12-14, 2022*.

Hachicha W., Ghorbel L., Champagnat R., Zayani C. A., Amous I. (2021). Using process mining for learning resource recommendation: A moodle case study. *Procedia Computer Science*, vol. 192, p. 853-862. Consulté sur https://www.sciencedirect.com/science/article/pii/S1877050921015763 (Knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 25th International Conference KES2021)

Ho H. N., Rabah M., Nowakowski S., Estraillier P. (2016). Toward a Trace-Based PROMETHEE II Method to answer " What can teachers do? " in Online Distance Learning Applications. In *13th International Conference on Intelligent Tutoring Systems*, p. 480-484. Zagreb, Croatia. Consulté sur https://hal.science/hal-01334151

Leblay J., Rabah M., Champagnat R., Nowakowski S. (2018). Process-based Assistance Method for Learner Academic Achievement. In *E-learning conference (el'2018)*, p. 89-96. Madrid, Spain. Consulté sur https://hal.science/hal-01834096

Li G., De Carvalho R. M. (2019). Process Mining in Social Media: Applying Object-Centric Behavioral Constraint Models. *IEEE Access*, vol. 7, p. 84360–84373.

Martinez P., Montañes O., Serralta J. M., Tansini L. (2021). Modelling computer engineering student trajectories with process mining. In *Latin american conference on learning analytics*, p. 48–57.

Müllner D. (2011). Modern hierarchical, agglomerative clustering algorithms. *arXiv preprint arXiv:1109.2378*.

Pika A., Wynn M. T., Budiono S., Hofstede A. H. M. ter, Aalst W. M. P. van der, Reijers H. A. (2019). Towards privacy-preserving process mining in healthcare. In C. Di Francescomarino, R. Dijkman, U. Zdun (Eds.), *Business process management workshops*, p. 483–495. Cham, Springer International Publishing.

Real E. M., Pimentel E. P., Oliveira L. V. de, Braga J. C., Stiubiener I. (2020). Educational process mining for verifying student learning paths in an introductory programming course. In *2020 ieee frontiers in education conference (fie)*, p. 1–9.

Romero C., Ventura S. (2013). Data mining in education. *Wiley Int. Rev. Data Min. and Knowl. Disc.*, vol. 3, n° 1, p. 12-–27. Consulté sur https://doi.org/10.1002/widm.1075

Romero C., Ventura S., García E. (2008). Data mining in course management systems: Moodle case study and tutorial. *Computers & Education*, vol. 51, n° 1, p. 368-384. Consulté sur https://www.sciencedirect.com/science/article/pii/S0360131507000590

Sedrakyan G., Snoeck M., De Weerdt J. (2014). Process mining analysis of conceptual modeling behavior of novices – empirical study using jmermaid modeling and experimental logging environment. *Computers in Human Behavior*, vol. 41, p. 486-503.

Song M., Günther C. W., Aalst W. M. Van der. (2008). Trace clustering in process mining. In *International conference on business process management*, p. 109–120.

Trabelsi M., Suire C., Morcos J., Champagnat R. (2019a). Fouille de processus auto-définis : cas d'étude d ' un moteur de recherche d ' une bibliothèque numérique. In *Inforsid'2019*, p. 131-146.

Trabelsi M., Suire C., Morcos J., Champagnat R. (2019b). User's behavior in digital libraries: Process mining exploration. In A. Doucet, A. Isaac, K. Golub, T. Aalberg, A. Jatowt (Eds.), *Digital libraries for open knowledge*, p. 388–392. Cham, Springer International Publishing.

Trabelsi M., Suire C., Morcos J., Champagnat R. (2021). A new methodology to bring out typical users interactions in digital libraries. In *2021 acm/ieee joint conference on digital libraries (jcdl)*, p. 11–20.

Troudi A., Ghorbel L., Amel Zayani C., Jamoussi S., Amous I. (2020). MDER: Multi-Dimensional Event Recommendation in Social Media Context. *The Computer Journal*, vol. 64, n° 3, p. 369-382. Consulté sur https://doi.org/10.1093/comjnl/bxaa126

Van der Aalst W. (2016). *Process mining: Data science in action*. Springer.

Weijters A., Ribeiro J. (2011). Flexible heuristics miner (fhm). In *Computational intelligence and data mining*, p. 310–317.

Zandkarimi F., Rehse J.-R., Soudmand P., Hoehle H. (2020). A generic framework for trace clustering in process mining. In *2020 2nd international conference on process mining*, p. 177–184.