



HAL
open science

Aide à l'analyse des parcours d'apprentissage en IUT par reconnaissance de procédés et recommandations à base de traces

Ronan Champagnat, Joffrey Leblay, Samuel Nowakowski, Mourad Rabah

► To cite this version:

Ronan Champagnat, Joffrey Leblay, Samuel Nowakowski, Mourad Rabah. Aide à l'analyse des parcours d'apprentissage en IUT par reconnaissance de procédés et recommandations à base de traces. Congès National de la Recherche des IUT, Jun 2016, Nantes, France. hal-01346459

HAL Id: hal-01346459

<https://hal.science/hal-01346459>

Submitted on 19 Jul 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Aide à l'analyse des parcours d'apprentissage en IUT par reconnaissance de procédés et recommandations à base de traces

Ronan Champagnat¹, Joffrey Leblay¹, Samuel Nowakowski², Mourad Rabah¹

¹ L3I, ULR, 17000, La Rochelle, France

² LORIA UMR 7503, UL, 54000, Nancy, France

THEMATIQUE – Informatique, section 27

RESUME – Cet article est une présentation des travaux d'expérimentations sur les données des jurys de l'IUT de La Rochelle. Notre objectif est de déterminer les trajectoires de réussite ou d'échec des étudiants durant leur DUT. Pour cela, nous basons nos analyses sur les traces, la fouille de procédés et la recommandation. Nous commençons par introduire notre démarche globale puis présentons les résultats obtenus grâce aux outils du *process mining*, les conjectures qui ont été faites sur ces résultats et, pour finir, les attentes pour la suite.

Mots-clés—*Traces, Recommandation, Fouille de Procédés, learning analytics*

1. INTRODUCTION

Nos travaux s'inscrivent dans le domaine de l'étude et l'aide à l'adaptation de parcours d'apprentissage. Pour cela, nos observations seront centrées sur le parcours de l'apprenant. Les parcours des apprenants sont vus comme des processus d'activité utilisateur. Nous définissons des propriétés sur la qualité de ces processus et il devient possible de repérer des processus qui peuvent aboutir à un échec. Notre objectif est donc d'analyser et de fouiller les parcours d'apprentissages pour pouvoir en extraire une connaissance qui nous permettra de faire des recommandations sur les processus en cours.

Lors de la découverte d'une trajectoire inadaptée, le système notifiera l'apprenant ou le formateur sur cette trajectoire d'éventuel échec. Il pourra, par ailleurs, intervenir automatiquement en déterminant l'action ou l'enchaînement d'actions qui permettront à l'apprenant ou au formateur de reconfigurer les séquences d'activités pédagogiques [1]. Nos travaux s'appuient sur plusieurs domaines d'expertise.

D'abord, la collecte et l'analyse des traces d'apprentissage permettront de récupérer les informations depuis les expériences passées des apprenants, ceci, à travers l'intégration d'un système à base de traces et la définition des observateurs et des modèles de transformation correspondants.

La fouille de procédés permettra ensuite d'identifier les processus représentant les trajectoires des apprenants dans l'application, en exploitant les traces générées précédemment. Ceci permettra au système d'identifier les processus pouvant aboutir à un échec. De plus, en utilisant un graphe constitué des processus prévus et représentant les trajectoires nominales, il devient possible de découvrir des anomalies de tout type. Ces anomalies peuvent permettre la découverte de raccourcis ou de nouvelles méthodes pour certains traitements.

Dans ce qui suit, nous commencerons par présenter les différents domaines qui vont être utilisés dans nos travaux. Les domaines d'expertise seront décrits dans l'ordre de leur utilisation dans le système que nous souhaitons mettre en place. Ensuite, nous décrirons l'intégration de ces trois domaines pour la résolution de notre problématique. Enfin, nous présenterons une mise en application aux données du département Informatique de l'IUT de La Rochelle.

2. SYSTÈMES À BASE DE TRACES

Les traces fournissent une image des activités du système. Elles sont d'ailleurs le point central de certains systèmes. Comprendre les traces permet de pouvoir récupérer des informations d'usages et de les analyser afin de visualiser les interactions entre un ou plusieurs humains et un système d'information [2]. Le principe d'utilisation des traces se décompose en trois étapes :

2.1. Collecte

La collecte permet l'observation du fonctionnement d'un système à partir d'une ou de plusieurs sources de données, appelées *sources de traçage*. Elle sert à convertir ou à transformer des informations générées par l'interaction utilisateur/système pendant l'activité en une trace initiale, dite *trace première*. Il y a deux techniques de traçage, l'approche ad hoc et celle par instrumentation de l'environnement de travail [3]. L'approche par instrumentation de l'environnement de travail est la technique de traçage utilisée dans le système proposé de notre cas d'étude. Elle consiste à collecter les traces par rapport à un objectif d'observation ainsi que de tracer un ensemble restreint d'observables, jugés pertinents, pendant la session d'observation. Avec cette méthode, il est possible d'associer des labels aux traces pour mieux pouvoir les traiter par la suite. Ces labels sont appelés *obsels*¹ et sont utilisés dans le cadre des traces modélisées ou *m-traces* [4]. Elles seront détaillées dans la suite de cette communication.

2.2. Transformation

Les traces obtenues à l'issue de la phase de collecte n'étant pas toujours exploitables directement, il est alors nécessaire d'effectuer une ou plusieurs transformations dont le résultat est une nouvelle trace appelée *trace transformée*. Dans cette étape, et

grâce à une collecte par instrumentation de l'environnement, nous pouvons créer des liens entre les traces ; ce qui permettra au *process mining* de les organiser sous la forme d'un graphe.

2.3. Exploitation

L'exploitation est l'étape où les traces obtenues peuvent être utilisées. Elles peuvent être utilisées telles quelles ou bien servir à l'élaboration d'indicateurs plus complexes. Il existe plusieurs méthodes d'exploitation des traces [5]. La visualisation, ou le fait de mettre en forme les données pour les afficher, souvent dans un but de surveillance. Le calcul d'indicateurs, où le but est de créer des données regroupant les informations des traces. Les valeurs de l'indicateur peuvent prendre des formes numériques, alphanumériques ou mêmes graphiques.

Il est aussi possible d'utiliser les techniques de *data mining* (ou fouille de données) qui permettront l'analyse de grandes quantités de traces numériques [6]. Dans ces cas-là, l'analyse se décompose elle-même en trois phases : (i) prétraitement des traces, (ii) découverte des patterns (ou motifs) d'utilisations intéressantes et (iii) analyse de ces patterns.

L'exploitation permet l'identification des processus types, les points de blocage ou autres. Elle est plus efficace sur les traces modélisées (*m-trace*) car, comme précisé plus haut, elles peuvent posséder des labels qui facilitent le traitement. Les *m-traces* sont des données plus complexes, car elles sont associées à la notion de modèle de traces. N'étant pas utilisées dans notre cas d'étude, elles ne seront pas présentées ici, en revanche, il est possible d'en trouver une description dans [7].

Dans notre cas applicatif, les traces récupérées ne sont pas toutes au même format. Ceci est dû aux diverses modifications de nomenclature interne au cours du temps. Aussi, il a été nécessaire de les faire passer dans un ETL (*Extract Transform Load*), un module de transformation qui modifie le format des données afin de les normaliser (le format utilisé sera expliqué dans la partie « Mise en application »).

Grâce aux traces (modélisées ou non), il est alors possible de faire des rétrospectives sur les expériences des apprenants et ainsi constituer une source de connaissances qui pourrait accompagner certains apprentissages pour des contextes définis. Il est envisageable d'utiliser ce principe en lien avec la fouille de procédés (*process mining*) pour construire une base des processus types et ainsi, grâce à la recommandation, de vérifier dynamiquement les traces afin d'identifier quel type de processus est entamé par l'apprenant.

3. FOUILLE DE PROCÉDÉS

La fouille de procédés, ou *process mining*, est utilisée à la base pour les processus métiers (*business processes*). Les processus sont créés à partir des traces qui sont récupérées depuis un système. Le *process mining* permet de découvrir les processus réels du système et de les surveiller. Le résultat de la phase de fouille est un graphe (par exemple RdP ou BPMN) représentant tous les processus observés. Nous utilisons le *process mining* dans le but d'identifier des processus particuliers comme un enchaînement d'actions non prévu par le système ou un processus bloquant ralentissant le travail des autres ressources. Pour la construction du graphe permettant ces recherches, il est possible d'utiliser les algorithmes de création (*Discovery Algorithm*) comme l'algorithme Alpha [8]. Il serait aussi possible d'utiliser les algorithmes de *data mining* comme l'algorithme GSP (*Generalized Sequential Pattern algorithm*). En revanche, comme les algorithmes de *Discovery* ont été développés spécifiquement pour la création de données propres au *process mining*, et que ceux-ci sont implémentés dans les logiciels de *process mining* utilisés, ils seront préférés aux autres algorithmes potentiels.

L'algorithme Alpha, est un algorithme de création (*discovery*) qui vise à reconstruire les enchaînements des suites d'événements et à les assembler dans un graphe. Il fait cela en se servant d'une clé primaire comme lien entre les événements : si deux événements se suivent directement sur la même valeur de clé primaire alors ils sont liés par un lien de causalité. Si deux événements liés par un lien de causalité se retrouvent en succession directe dans le sens opposé au sens qui a engendré leur lien alors ils sont en parallèle.

Le domaine est encore jeune mais il existe déjà plusieurs outils pour modéliser les réseaux de processus comme ProM [9] et Disco qui permettent de faire de la visualisation à partir de trace. Pour une utilisation plus simple de ces logiciels, la norme XES (Extensible Event Stream) à base de XML permet de faciliter la communication entre les outils de création et les outils de recherche dans le *process mining* [10].

Une fois que le *process mining* s'est chargé de créer l'enchaînement des processus, il devient extrêmement simple de suivre les processus des apprenants depuis les traces observées par le système. Puis, grâce au système de recommandation, il sera possible d'anticiper les futures actions entreprises et de proposer des processus alternatifs.

4. SYSTÈMES DE RECOMMANDATION

Les SR sont des composants qui recueillent des informations sur les préférences des utilisateurs pour construire un ensemble d'items (par exemple, des films, des chansons, des livres, des applications, des sites Web ou même des cours particuliers). L'information peut être acquise explicitement (typiquement en recueillant les évaluations des utilisateurs) ou implicitement (généralement en surveillant le comportement des utilisateurs, tels que les chansons écoutées, applications téléchargées, les sites Web visités et les livres lus) [11]. Les SR peuvent inclure des caractéristiques démographiques des utilisateurs comme l'âge, la nationalité, le sexe, ainsi que des informations sociales comme les *likes*, *followers*, *followed*, *tweets*, et *posts*. On observe une tendance grandissante pour l'utilisation des informations de l'Internet des objets comme les localisations GPS et RFID permettant de faire des rapprochements aidant à la recommandation [12]. Les SR utilisent des méthodes de filtrage comme le filtrage collaboratif [13], à base de contenu [15] et hybride [16].

Dans le cadre de notre étude, nous utiliserons le filtrage collaboratif qui est un algorithme de recommandation calculant ses prédictions et recommandations à partir des notes ou des comportements des autres utilisateurs, sur le comportement dans notre cas, dans le système. L'hypothèse fondamentale est que les opinions des autres utilisateurs peuvent être sélectionnées et regroupées de manière à fournir une prédiction raisonnable des opinions des utilisateurs traités. Le principe est simple. Si plusieurs utilisateurs, qui ont un profil ou une liste d'opinions similaires avec l'utilisateur traité, ont une opinion sur un item (musique, film, livre ou

même un autre utilisateur) que l'utilisateur traité n'a pas encore vu alors il est possible que celui-ci ait la même opinion que les autres. Le FC peut aussi être utilisé avec les items plutôt que les utilisateurs, on appelle cela le *FC à base d'items* [13]. Le FC joue un rôle important dans les SR bien qu'il soit, bien souvent, mixé avec d'autres techniques [14] comme le filtrage à base de contenu.

Les SR utilisent les méthodes de filtrage pour éliminer les données qui ne ressemblent pas à l'item traité. Mais pour pouvoir identifier les données similaires ou non il faut utiliser un *calcul de similarité*. Une méthode de calcul de similarité est une approche permettant de créer un score pour quantifier la ressemblance entre deux items sur les valeurs observées. Par exemple, en calculant la similarité entre deux apprenants, il est possible de savoir quels peuvent être les processus à échec que l'un d'eux doit éviter, compte tenu de ses faiblesses. Il existe plusieurs méthodes pour trouver ce score :

- Les réseaux Bayésiens (*Bayesian Networks*)
- L'algorithme de Pearson
- La similarité cosinus (ou mesure cosinus)
- Le TF-IDF (*Term Frequency-Inverse Document Frequency*)

Chacune de ces méthodes possède des cas d'utilisation bien précis, mais dans notre cas d'utilisation, la méthode la plus appropriée est l'utilisation des *réseaux Bayésiens (Bayesian Networks)*. Ceux-ci permettent de calculer la probabilité qu'un item soit le suivant en fonction de ce que l'utilisateur a déjà traité. Il s'agit d'un graphe pondéré possédant une notion d'historique d'actions et représentant l'arbre de décision des actions possibles. L'avantage des réseaux bayésiens est qu'ils permettent de recommander des items en prenant en compte les tendances de l'utilisateur. Plus simplement, quand l'utilisateur progresse dans le réseau, il précise ces futures trajectoires (en fonction des trajectoires empruntées par les utilisateurs précédents). De ce fait, si l'utilisateur emprunte une trajectoire menant à un échec, alors le système pourra l'en informer. Il existe une forme de ces réseaux qui ne prend pas en compte la notion d'historique d'action, on parle alors de réseaux markoviens correspondant à un graphe d'états.

5. MISE EN APPLICATION

Le contexte d'application auquel nous nous intéressons est celui des données de passage entre les semestres du DUT du département Informatique de l'IUT de La Rochelle. Notre objectif est d'analyser les parcours des apprenants et de les utiliser pour : d'une part aider leur progression et d'autre part pour découvrir les potentielles matières saillantes. Le parcours désigne le processus suivi par un étudiant et peut avoir une granularité variable : par exemple, à un niveau bas, on peut considérer les UE réalisées et, à un niveau haut, considérer la réussite ou non du semestre entier de la formation.

Cet article traitera les données du jury comme activité pour les processus et ne traitera les notes des UE que comme des données statistiques. Cette granularité permettra de pouvoir observer le parcours des apprenants. Grâce à ce résultat, il sera possible d'identifier les apprenants ayant besoin d'aide pour valider leurs semestres. De plus, avec une analyse statistique, basée sur les apprenants qui ont échoué à un semestre, il est possible de découvrir les raisons de ces lacunes. Pour cela, nous observerons les matières (UE11, UE12, etc.) avec les bacs d'origine.

5.1. Mise en contexte

Dans les IUT, certains étudiants ont des difficultés à réussir leurs semestres, à cause de leur profil (cursus d'origine) ou de leurs faiblesses dans certaines matières. Pour effectuer notre étude, les résultats de l'IUT de La Rochelle ont été mis à disposition. Le but final est de corréliser des informations statistiques aux informations issues du *process mining* afin de pouvoir identifier les causes des échecs, plutôt que seulement se baser sur le parcours macroscopique. L'utilité du *process mining* est d'identifier des problèmes ou des comportements atypiques sur l'enchaînement des données.

5.2. Démarche scientifique

Les données obtenues sont hétérogènes, aussi il est nécessaire de les normaliser afin de les traiter avec des outils de *process mining* (cf. section « Fouille de Procédés »).

Le modèle du Tableau 1 a été utilisé pour obtenir le graphe d'états représentant le modèle des parcours des apprenants (réseau markovien). Le résultat est illustré par la Fig.1. Le bac d'origine n'intervient que pour une aide supplémentaire dans le cas d'un bac non adapté. Si la majorité des apprenants venant d'un bac particulier n'arrive pas à valider un semestre une aide supplémentaire pourra leur être prescrite.

Tableau 1. Format de données pour la création du graphe d'états

Identifiant	Décision du jury	Semestre	Bac d'origine	Note UE11	...
000042	V	S1	S	12,5	...

Pour la modélisation des données en arbre de décision (réseau bayésien), il suffit de reprendre le format précédent en fusionnant les données du semestre, de la décision du jury et en prenant en compte la notion de trajectoire (soit la décision du jury au semestre précédent). Le format se décrira avec les acronymes suivants : V la validation du semestre de droit par les notes obtenues, J la validation par le jury, N l'échec du semestre, A une absence de l'apprenant à un ou plusieurs partiels et E une exclusion de l'apprenant par le jury pour résultats insuffisants. Il est possible de trouver aussi des notions de REO et de DEM signifiant respectivement la réorientation de l'étudiant vers une autre filière ou un autre établissement, et la démission de l'étudiant. Pour finir, nous proposons ce format sous la forme d'une regex composée avec les acronymes ci-dessus : $(([VJNAE]|C[+-])[1-4])\{4,6\}$

Exemple : V1N2N3C-2C+3V4 ou J1V2C-3N4V4

La regex présentée est une simplification inclusive du format mis en place car la regex complète serait plus complexe et moins lisible. Grâce à ce format, nous obtenons le résultat de la Fig.2. Le problème de l'arbre de décision est qu'il demande beaucoup de données pour être mis en place. En revanche, il permet de voir avec plus de précision le parcours de l'apprenant et ses origines.

Ces résultats sont obtenus en travaillant sur des données issues de deux transformations différentes des mêmes traces. En effet, ces transformations sont effectuées sur les mêmes traces premières. La différence consiste dans la notion de trajectoire de provenance des états représentant les activités considérées, qui sont ici la validation des semestres.

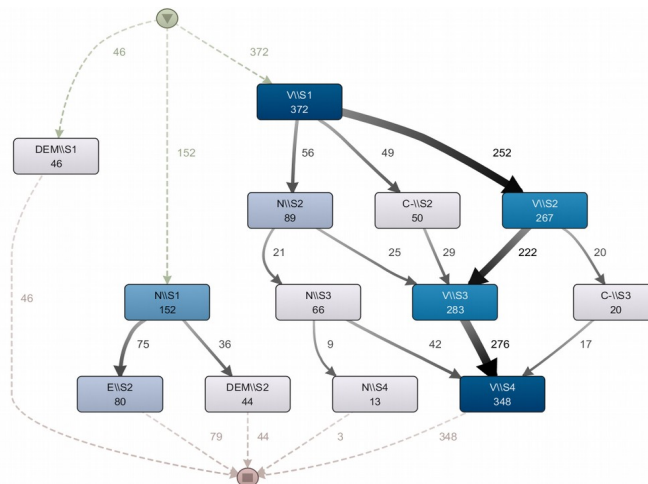


Fig.1 Graphe d'états pour les cas les plus typiques

Malgré les différences de ces données, la corrélation entre elles deux peut être productive. Le graphe d'états permet de voir tous les chemins possibles, tandis que la modélisation en arbre de décision permet de voir les trajectoires des différents parcours. Il est donc possible par une comparaison entre les deux de montrer les chemins non empruntés et potentiellement trouver des raccourcis ou de nouvelles méthodes de traitement.

5.3. Résultats des expérimentations

Ces résultats ont été obtenus sur une base de 620 étudiants des promotions de 2007 à 2013.

Les résultats de la Fig.1 sont les résultats de la mise en application du modèle du Tableau 1. Le résultat est un graphe permettant une visualisation simple des processus existant dans les traces utilisées. La Fig.1 permet de découvrir le(s) état(s) des processus visés les moins fréquents, comme les états V2 et V3 pour le processus visé V1V2V3V4. Ce qui permet de soupçonner la présence dans la Fig.2 d'une action alternative. Cette action n'est pas forcément négative mais dans le cas où elle l'est, il sera nécessaire d'étudier les autres données observées afin de découvrir les raisons de cette action et de tenter d'entreprendre un recadrage par recommandation.

Le graphe de la Fig.2 est le résultat étiqueté par la regex expliquée ci-dessus. Il permet une visualisation des processus avec une notion d'historique permettant d'observer les différents processus avec plus de précision qu'avec le graphe précédent. Le résultat complet est très imposant, il n'en sera traité qu'une partie. La partie traitée dans ce document concerne les étudiants ayant raté leur premier semestre.

Actuellement, les expérimentations, sur les données de la Fig.2, montrent que les apprenants ayant raté leur premier semestre n'ont que très peu de chance d'obtenir leur second semestre. La raison de cet échec pourrait s'expliquer par des manques de compétences. Cela permet de prouver que le second semestre pose ses enseignements sur des compétences qui, si elles ne sont pas acquises, empêchent la compréhension et donc l'apprentissage. Le but final est que le système arrive automatiquement à ces observations afin de pouvoir aider un apprenant dans son processus d'apprentissage.

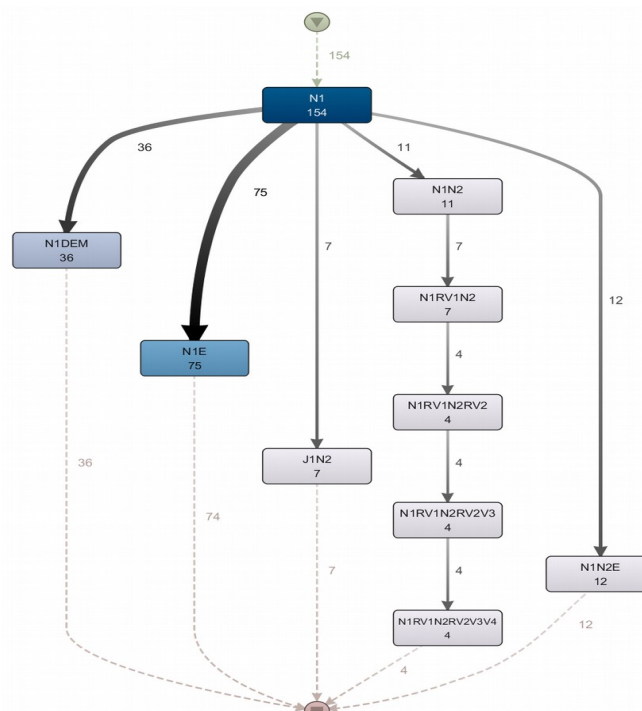


Fig.2 Arbrescence pour les cas d'échecs au premier semestre

6. CONCLUSION

Cet article a présenté notre travail exploratoire dans l'aide à l'analyse des processus utilisateurs appliqué aux parcours des étudiants en IUT. Nous avons exposé une méthodologie organisée autour de trois domaines d'expertise : les traces d'exécutions, les recommandations et le *process mining*. Le but de notre démarche est d'analyser les processus utilisateurs dans des applications interactives afin d'en améliorer le pilotage.

Cet article décrit des travaux préliminaires qui montrent comment, à partir des traces d'exécution d'un système, nous pouvons identifier les processus atypiques, anormaux ou erronés. Dans la suite de nos travaux, nous prévoyons la définition de mécanismes de pilotage à base de systèmes de recommandation dont les données d'entrée seront issues de l'analyse des traces et du *process mining* présentés ici. Ces données seront, au préalable, corrélées aux profils utilisateurs.

Comme cadre applicatif, nous avons choisi la réussite des étudiants dans leur parcours de semestres en DUT Informatique. Cependant, notre méthodologie pourra être généralisée à d'autres types de processus tels que l'enchaînement d'activités dans un EIAH (Environnement Informatique pour l'Apprentissage Humain) [1] ou les processus métiers d'un ERP (*Entreprise Resource Planning*).

7. RÉFÉRENCES

- [1] Leblay J. (2016), Aide à la navigation dans les parcours d'apprentissage par reconnaissance de procédés et recommandations à base de traces. Accepté aux 6èmes Rencontres Jeunes Chercheurs en EIAH, Montpellier.
- [2] Lund, K., & Mille, A. (2009). Traces, traces d'interactions, traces d'apprentissages : définitions, modèles informatiques, structurations, traitements et usages. Dans Analyse de Traces et Personnalisation Des EIAH. Lavoisier-Hermès, Paris. 21–66.
- [3] Ho, H. N. (2015). Trace-Based Multi-Criteria Decision Making in Interactive Application for Adaptive Execution. Thèse de Doctorat, Université de La Rochelle.
- [4] Champin, P.-A., Mille, A., & Prié, Y. (2013). Vers des traces numériques comme objets informatiques de premier niveau : une approche par les traces modélisées. *Intellectica*, (59) : 171–204.
- [5] Djouad, T., Settouti, L. S., Prié, Y., Reffay, C., & Mille, A. (2010). Un Système à Base de Traces pour la modélisation et l'élaboration d'indicateurs d'activités éducatives individuelles et collectives. Mise à l'épreuve sur Moodle. *Technique et Science Informatiques*. 29(6) : 721–741.
- [6] Loghin, G.-C. (2008). Observer un Environnement Numérique de Travail pour réguler les activités qui s'y déroulent. Thèse de Doctorat, Université de Savoie
- [7] Settouti, L. (2011). Systèmes à Base de Traces Modélisées : Modèles et Langages pour l'exploitation des traces d'Interactions. Thèse de Doctorat, Université Claude Bernard Lyon 1.
- [8] Van der Aalst, W. M. P., Reijers, H. A., Weijters, A. J. M. M., Van Dongen, B. F., Alves de Medeiros, A. K., Song, M., & Verbeek, H. M. W. (2007). Business process mining : An industrial application. *Information Systems*, 32(5) : 713–732.
- [9] Van Dongen, B. F., Varbeek, H. M. W., Weijters, A. J. M. M., & Van Der Aalst, W. M. P. ; (2005). The ProM Framework : A New Era in Process Mining Tool Support. in Proceedings of 26th International Conference on Applications and Theory of Petri Nets. LNCS 3536. Miami, USA. 444–454.
- [10] Verbeek, H. M. W., Buijs, J. C. A. M., Van Dongen, B. F., & van der Aalst, W. M. P. (2010). XES, XESame, and ProM 6. In Information Systems Evolution (pp. 60–75). Springer.
- [11] Lee, S. K., Cho, Y. H., & Kim, S. H. (2010). Collaborative filtering with ordinal scale-based implicit ratings for mobile music recommendations. *Information Sciences*, 180(11), 2142–2155.
- [12] Levandoski, J. J., Sarwat, M., Eldawy, A., & Mokbel, M. F. (2012). LARS : A Location-Aware Recommender System. in Proceedings of IEEE 28th International Conference on Data Engineering (ICDE'2012). Arlington, VA, USA. 450–461.
- [13] Linden, G., Smith, B., & York, J. (2003). Amazon.com recommendations : Item-to-item collaborative filtering. *IEEE Internet Computing*. 7(1), 76–80.
- [14] Alchikh Haydar, C. (2014). Les systèmes de recommandation à base de confiance. Thèse de Doctorat, Université de Lorraine.
- [15] Pazzani, M. J., & Billsus, D. (2007). Content-Based Recommendation Systems. The adaptive web. LNCS 4321. 325–341.
- [16] Burke, R. (2007). Hybrid web recommender systems. The adaptive web. LNCS 4321. 377–408.