

Rétroactions Personnalisées Basées sur les *Learning Analytics* pour Soutenir l'Apprentissage de la Programmation

Zahi Hodeib^{1,2}

¹ Université de Lille, CNRS, UMR 9189, CRISTAL, F-59000 Lille, France

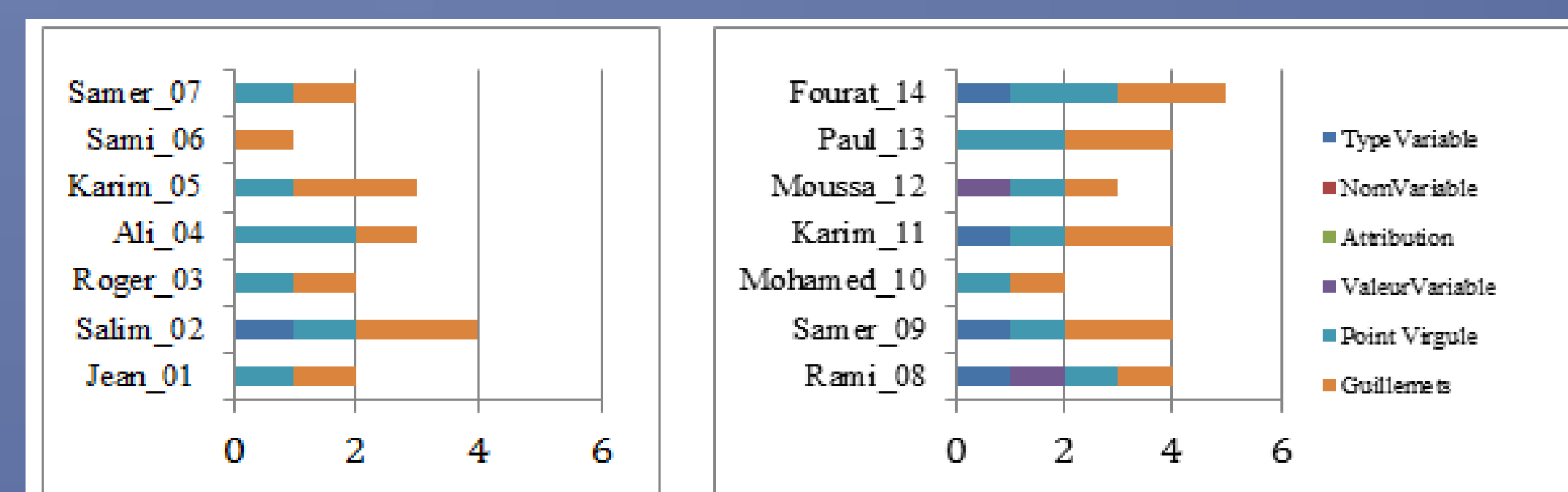
² Université Libanaise, faculté de technologie, RADIOCOM, Saida, Liban
zahi.hodeib@univ-lille.fr

Contexte et Objectif

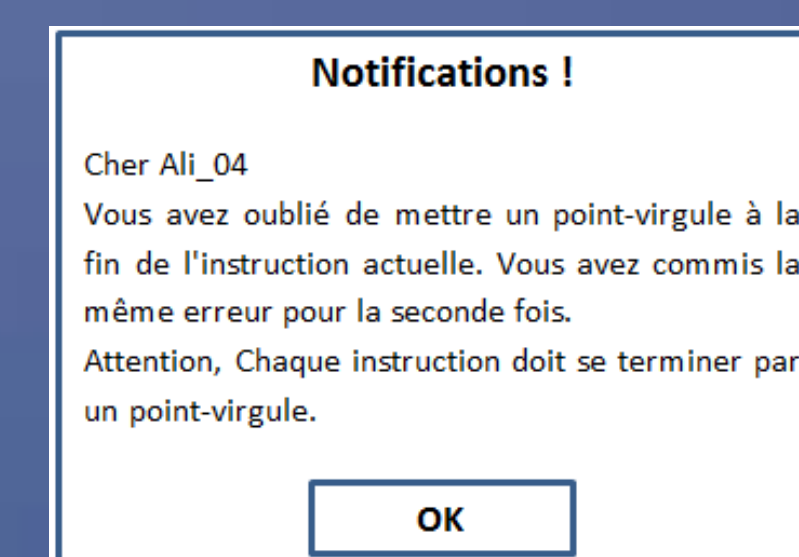
Au cours des vingt dernières années, l'apprentissage de la programmation est devenu essentiel dans le cursus de nombreuses filières académiques. Cependant, son apprentissage pratique représente une difficulté pour les novices [1]. Ce qui explique clairement les taux élevés d'échecs et d'abandon dans les cours d'initiation à la programmation [2]. Généralement, ces activités sont caractérisées par une forte densité d'apprenants, ce qui augmente la charge de travail des enseignants pour assurer un bon suivi [3]. Bien que certains EIAH fournissent des rétroactions aux apprenants, la qualité de ces dernières reste non satisfaisante, vu qu'elles sont basées sur l'analyse d'une faible quantité d'informations collectées sur les apprenants [4]. Les *Learning Analytics* (LA) visent à exploiter efficacement les traces numériques des apprenants pour améliorer le processus d'apprentissage [5]. Nous présentons ainsi un nouvel environnement d'apprentissage de la programmation nommé Codeclick. L'environnement s'appuie sur une approche statistique de LA basée sur les traces numériques des apprenants pour fournir des rétroactions personnalisées et des visualisations destinées à améliorer l'apprentissage des novices.

Expérimentation et Résultats

Nous avons mené une expérimentation sur deux groupes d'apprenants, formés chacun de sept membres, sélectionnés arbitrairement et pour la même activité. La durée de l'activité était fixée à 10 minutes et les apprenants étaient uniquement des garçons avec un âge moyen de 19 ans. Le groupe G1 a réalisé l'activité en mode avec rétroactions personnalisée et le groupe G2 utilise le mode rétroactions simples. Notre but est d'évaluer le niveau de performance des apprenants afin de fournir des rétroactions personnalisées. Pour ce faire nous nous sommes appuyés sur des statistiques basées sur des indicateurs (*NE*, *NC*, *T*). Ces statistiques sont utilisées pour générer des rétroactions et visualisations. Les résultats préliminaires montrent que les apprenants du groupe G1 ont atteint des niveaux de performance plus satisfaisants en termes d'erreurs commises et de temps moyen passé par activité de 8 minutes.



Visualisations représentant le nombre et types d'erreurs des groupes G1 et G2 respectivement



Exemple d'une rétroaction fournie à la détection d'une erreur

Méthodes

- Collecte des indicateurs et des traces numériques
- Classification et Analyse statistique
- Rétroactions et Visualisations

Codeclick collecte et analyse les traces numériques des apprenants:

- Nombre d'Erreurs (*NE*)
- Nombre de Clics (*NC*)
- Temps passé par activité (*T*) en minutes
-

Il les classifie et les stocke dans une base de données. Ces traces subissent ensuite une phase d'analyse statistique basée sur des indicateurs pour évaluer la performance de chaque apprenant selon la formule ci-dessous et par suite générer des rétroactions et des visualisations illustrant ce niveau de performance. Nous avons attribué une pondération à chaque indicateur en fonction de son importance dans la réalisation de l'activité pratique.

$$P = 100 - 2.5 * NC - 3 * NE - T$$

Conclusion

L'apprentissage de la programmation peut être amélioré par des mécanismes de rétroactions personnalisées, basées sur les traces numériques des apprenants. Nous avons présenté un nouvel environnement d'apprentissage de la programmation nommé Codeclick, qui analyse les traces numériques des apprenants en se basant sur des indicateurs et fournit des rétroactions personnalisées pour soutenir ces derniers. Les résultats préliminaires montrent que les rétroactions fournies par Codeclick améliorent la performance des apprenants et cela s'est traduit par un nombre d'erreurs réduit et un temps acceptable passé par activité.

Références

- Lahtinen E, Ala-Mutka K, Järvinen H-M. A Study of the Difficulties of Novice Programmers. *Proceedings of the 10th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education*. ITICSE '05. New York, NY, USA : ACM, 2005 : 14–18.
- Bennedsen J, Caspersen ME. Failure Rates in Introductory Programming. *SIGCSE Bull.* 2007 ; 39 : 32–36.
- Gulwani S, Radiček I, Zuleger F. Feedback Generation for Performance Problems in Introductory Programming Assignments. *Proceedings of the 22nd ACM SIGSOFT International Symposium on Foundations of Software Engineering*. FSE 2014. New York, NY, USA : ACM, 2014 : 41–51.
- Ferguson P. Student perceptions of quality feedback in teacher education. *Assessment & Evaluation in Higher Education* 2011 ; 36 : 51–62.
- Pardo A, Dawson S, Dawson S. Learning Analytics: How Can Data Be Used to Improve Learning Practice? *Measuring and Visualizing Learning in the Information-Rich Classroom* 2015 .