Early Identification of Novice Programmers' Challenges in Coding Using Machine Learning Techniques

Alireza Ahadi

University of Technology, Sydney
Sydney, NSW, 2008, Australia
Alireza.Ahadi@uts.edu.au

ABSTRACT

It is well known that many first year undergraduate university students struggle with learning to program. Educational Data Mining (EDM) applies machine learning and statistics to information generated from educational settings. In this PhD project, EDM is used to study first semester novice programmers, using data collected from students as they work on computers to complete their normal weekly laboratory exercises. Analysis of the generated snapshots has shown the potential for early identification of students who later struggle in the course. The aim of this study is to propose a method for early identification of "at risk" students while providing suggestions on how they can improve their coding style. This PhD project is within its final year.

Categories and Subject Descriptors

K.3.2 [**Computers and Education**]: Computer and Information Science Education – *computer science education.*

Keywords

Snapshot Analyzis; Novice Programmers; Machine Learning

1. CONTEXT AND MOTIVATION

Every year, tens of thousands of students fail introductory programming courses world-wide. As a consequence, studies are retaken or postponed, careers are reconsidered, and substantial capital is invested into student counseling and support. World-wide, on average one third of students fail their introductory programming course. Even when looking at statistics describing pass rates after teaching interventions, as many as one quarter of the students still fail the courses. Thus, automated early identification of students' performance within the course is important. The output of this PhD study will not only contribute to shaping assessment/automated tutoring systems, but also help improve our understanding of the ways novices develop their coding patterns and suggest changes to the pedagogy of introductory programming courses.

2. BACKGROUND & RELATED WORK

In "Methods and Tools for Exploring Novice Compiling Behavior", Jadud presented a method to quantify a student's tendency to create and fix errors, which he called the *error quotient* [1]. In his study, the correlation between the error quotient and the average score from programming assignments was mediocre but statistically significant ($r = 0.36$; $p = 0.012$), while the correlation between the error quotient and the grade from a course exam was high ($r = 0.52$; $p = 0.0002$). Rodrigo et al. used an alternative version of Jadud's error quotient, and found that in their context the correlation between the error quotient and the midterm score of an introductory programming course was strong and statistically significant ($r = -0.54$; $p < 0.001$) [2]. In essence, this suggests that the fewer programming errors students make, and the higher the midterm grade.

Watson et al. also conducted a study using Jadud's error quotient, and found a significant correlation between the error quotient and their programming course scores ($r = 0.44$) [3]. They proposed that the amount of time that students spend on programming assignments should be taken into account, and that one should consider the files that a student is editing as a part of the error quotient calculation. They proposed an improvement to the error quotient called *Watwin*, and found that with this improvement the correlation increased from ($r = 0.44$) to ($r = 0.51$). They also noted that a simple measure, the average amount of time that a student spends on a programming error, is strongly correlated with programming course scores ($r = -0.53$; $p < 0.01$).

3. STATEMENT OF THESIS/PROBLEM

The current formulation of this PhD study's research question is as follows:

What, if there is any, would be an environmental independent common attribute of the data collected from novices in different programming languages which could automatically classify students according to their ability of coding?

The research goals are as follows:

Data Collection: data used in this study is collected from source code snapshots generated by novices enrolled in an introductory programming courses at a) University of Helsinki, Finland. b) University of Technology, Sydney, and c) snapshots generated by novices enrolled in an database fundamental course at University of Technology, Sydney. The collected programming snapshots represent line-edit level [4] snapshots of the students' main source code while working on a programming task, while the database SQL SELECT statements represent different attempts of novices in writing SQL code.

Replication: analyzing the collected data in an unsupervised machine learning framework to assess the state of the art techniques described above. The collected source code snapshots

ICER'16, September 9-11, 2016, Melbourne, Australia.
ACM 978-1-XXXX-XXXX-X/XX/XX.
http://dx.doi.org/#######

will be used to identify possible clusters of novices according to different features extracted from their source codes.

Proposition and Comparison: analyzing the collected data in order to compare the performance of new methods proposed as a part of this PhD with the state of the art methods. Based on the output of different machine learning tools, a set of context-independent features extracted from the data as well as a systematic approach for analyzing them will be proposed.

4. PROGRAM CONTEXT

Accomplishing the final output of this research project involves fulfilling the objectives of three main phases. The first phase is dedicated to analysis of snapshot data in different contexts [5]–[9] to understand the data and find possible links to what previously has been reported [10]–[16]. The second phase is devoted to designing machine learning derived techniques for snapshot data analysis [6] . The third and last phase of the project is dedicated to performing the comparative analysis of the proposed techniques of analyzing snapshot data with the state of the are techniques in a large scale on different datasets.

5. DISSERTATION STATUS

This PhD project is within a year of completion. The data collection has been completed successfully and some preliminary results have been generated based on an original hypothesis. Replication and comparison stages are almost completed. Based on findings so far, there is a significant and strong negative correlation between the number of steps a novice takes to complete a coding task successfully and her overall performance in the programing subject.

6. REFERENCES

[1] M. C. Jadud, "Methods and tools for exploring novice compilation behaviour," *Proc. 2006 Int. Work. Comput. Educ. Res. ICER 06*, vol. 09, no. Figure 1, pp. 73–84, 2006.

[2] M. M. T. Rodrigo, E. Tabanao, M. B. E. Lahoz, and M. C. Jadud, "Analyzing online protocols to characterize novice java programmers," *Philipp. J. Sci.*, vol. 138, no. 2, pp. 177–190, 2009.

[3] C. Watson, F. W. B. Li, and J. L. Godwin, "Predicting performance in an introductory programming course by logging and analyzing student programming behavior," in *Proceedings - 2013 IEEE 13th International Conference on Advanced Learning Technologies, ICALT 2013*, 2013, pp. 319–323.

[4] A. Ahadi, "Applying Educational Data Mining to the Study of the Novice Programmer , within a Neo-Piagetian Theoretical Perspective" in *Psychology of Programming Interest Group*, 2014, pp. 197–202.

[5] A. Ahadi, R. Lister, H. Haapala, and A. Vihavainen, "Exploring Machine Learning Methods to Automatically Identify Students in Need of Assistance," *Icer'15*, pp. 121–130, 2015.

[6] A. Ahadi, A. Vihavainen, and R. Lister, "On the Number of Attempts Students Made on Some Online Programming Exercises During Semester and their Subsequent Performance on Final Exam Questions," in *ACM conference on Innovation and Technology in Computer Science Education*, 2016.

[7] A. Ahadi, J. Prior, V. Behbood, and R. Lister, "Students ' Semantic Mistakes in Writing Seven Different Types of SQL Queries," in *ACM Conference on Innovation and Technology in Computer Science Education*, 2016.

[8] A. Ahadi, V. Behbood, A. Vihavainen, J. Prior, and R. Lister, "Students ' Syntactic Mistakes in Writing Seven Different Types of SQL Queries and its Application to Predicting Students ' Success," in *SIGCSE '16: Proceedings of the 47th ACM Technical Symposium on Computer Science Education*, 2016.

[9] A. Ahadi, J. Prior, V. Behbood, and R. Lister, "A Quantitative Study of the Relative Difficulty for Novices of Writing Seven Different Types of SQL Queries," in *Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education*, 2015, pp. 201–206.

[10] A. Ahadi and R. Lister, "Geek genes, prior knowledge, stumbling points and learning edge momentum: parts of the one elephant?," *Proc. Ninth Annu. Int. ACM Conf. Int. Comput. Educ. Res. (ICER 2013)*, pp. 123–128, 2013.

[11] D. Teague, M. Corney, A. Ahadi, and R. Lister, "Swapping as the ' Hello World ' of Relational Reasoning : Replications , Reflections and Extensions," pp. 87–94, 2012.

[12] A. Ahadi, R. Lister, and D. Teague, "Falling Behind Early and Staying Behind When Learning to Program," in *25th Anniversary Psychology of Programming Annual Conference*, 2014.

[13] D. Teague, M. Corney, C. Fidge, M. Roggenkamp, A. Ahadi, and R. Lister, "Using Neo-Piagetian Theory , Formative In-Class Tests and Think Alouds to Better Understand Student Thinking : A Preliminary Report on Computer Programming Design : Neo-Piagetian Theory of Cognitive Development," 2012.

[14] D. Teague, R. Lister, and A. Ahadi, "Mired in the Web : Vignettes from Charlotte and Other Novice Programmers," 2014.

[15] M. Corney, D. Teague, A. Ahadi, and R. Lister, "Some Empirical Results for Neo-Piagetian Reasoning in Novice Programmers and the Relationship to Code Explanation Questions," *14th Australas. Comput. Educ. Conf.*, pp. 77–86, 2012.

[16] D. Teague, M. Corney, A. Ahadi, and R. Lister, "A qualitative think aloud study of the early neo-piagetian stages of reasoning in novice programmers," in *Proceedings of the Fifteenth Australasian Computing Education Conference*, 2013, vol. 136, pp. 87–95.