

Motivating Introductory Computing with Big Data and Blocks

Austin Cory Bart
Virginia Tech
2202 Kraft Drive
Blacksburg, VA, 24060, USA
acbart@vt.edu

ABSTRACT

My research is focused on motivating and scaffolding introductory computing experiences. Non-traditional computing students in courses such as Computational Thinking pose a unique challenge because of their limited motivation and skill, and the scale of such classes. To overcome these limitations, I propose a novel block-based programming environment leveraging mutual language translation to authentically transition students from the block-based code to a text-based programming language. The system will use sophisticated program analysis to guide students to success and report learner progress to instructors. The tool will synergize with a large collection of beginner-friendly big data sources in order to authentically contextualize the learning experience and thereby improve learner motivation. Data will be gathered on students' motivation and learning progress and instructors' perceptions of the success of the tool in order to answer my hypotheses about the efficacy of the this approach and the tools that are developed to support it.

Categories and Subject Descriptors

K.3.2 [Computers and Education]: Computers and Information Science Education—*Computer Science Education*.

Keywords

Computational Thinking, Big Data Science, Block-based Programming

1. PROGRAM CONTEXT

I am in the third year of a PhD program at Virginia Tech in Computer Science. I have completed all but two of the courses needed for my program. Additionally, I have just completed a 4-course certification in the Learning Sciences. My research is focused on motivating and scaffolding introductory computer science classes. I have established a major on-going research project to integrate Big Data Science into

introductory programming experiences as a motivating context (the CORGIS project). This project and its associated software scaffolding are now deeply integrated into a course on Computational Thinking meant for a diverse crowd of learners. Although preliminary results on the materials have begun to be gathered and published, much work remains to evaluate them and revise the associated technology.

2. CONTEXT AND MOTIVATION

Computational Thinking is increasingly considered a 21st century competency, and plays a growing role within universities' general education curriculum. Although the term itself is historically ill-defined, a given educator can operationalize it into well-defined learning objectives – typically, applying computer science concepts and computational techniques to frame problems and devise solutions in broad areas of study. At Virginia Tech, we have created a new course for non-major CS students on Computational Thinking. These students represent diverse majors: arts, sciences, agriculture, and many others. Non-major learners represent a challenge because they are not confident it will be a useful experience nor about their ability to succeed in the course. Complicating these problems is that enrollment must scale aggressively due to its critical role within general education requirements. The problem here is three-fold: How do we meaningfully motivate these learners, guide them to success, all while scaling the course offering to as large a population as possible? This work will provide new insights in how to successfully motivate and guide large numbers of non-major students through the introductory programming experience and also provide new technology to support the process.

3. BACKGROUND & RELATED WORK

The current dominant approach to motivating introductory students focuses primarily on games and animations as the primary context, such as the popular Scratch/Snap and Media Comp curriculums. Although these materials have proven successful for younger learners, they have been criticized for their inauthenticity for more mature audiences. An extensive intervention [2] failed to convince students that Media Computation was an inherently valuable skill.

Motivation is a complex construct, and there are many theories to explain engagement. The MUSIC Model of Academic Motivation [3] holistically explains motivation within academic contexts. The model identifies 5 components to motivation – eMpowerment (sense of agency), Usefulness (utility to their short-term and long-term interests), Success (self-efficacy), Interest (situationally and to their identity),

and Caring (of the stakeholders in the learning environment towards them). Students are motivated when one or more of these constructs is sufficiently activated. Since each learner and experience draws upon these components differently, it is crucial to appeal to as many as possible. This model identifies the limitation of relying on situationally interesting materials, instead of broadly satisfying learners' needs.

I suggest Big Data Science as an alternative context to motivate learners and convince them of the utility of the material while personalizing the experience and providing agency – every student can find data in their discipline and interests. This emerging context is growing in popularity [1]. Of course, motivation is only half the battle. Most of these students must overcome low self-efficacy and inexperience with the material. I propose using a new Block-based environment as our primary scaffold. To support the transfer from an inauthentic block-based editor to conventional text coding, we leverage the emerging area of Mutual Language Translation [4] to provide 1-1 isomorphism of blocks with the textual equivalent. This mechanism is coupled with state-of-the-art program analysis techniques to provide just-in-time feedback to the learner and feedback to instructors.

4. STATEMENT OF THESIS/PROBLEM

My primary thesis question is: How can we most effectively motivate a large quantity of introductory Computational Thinking students and guide them to success? My hypothesis is that a Big Data Science approach can be scaffolded with a block-based programming environment that supports learners and teachers in a blended learning experience. I will collect data to test this hypothesis by intervening directly in a new Computational Thinking class here at Virginia Tech using technology that I have created. We will gather self-reported motivational data, in-class observations of students, and fine-grained programming exercise progress. As the course materials are refined, I expect meaningful gains across motivation, ability, and the instructors' reported experiences. A number of secondary questions arises from Big Data Science as an introductory context. A major element of the course is the availability and range of datasets. How does the form and affordances of the data shape the learning experience, and how do we rapidly provide beginner-friendly datasets? These questions will be answered through the development and analysis of datasets.

5. RESEARCH GOALS & METHODS

To test my hypotheses, a number of new pieces of technology will need to be created. I am creating a new block-based programming environment with tools for Big Data Science and transitioning seamlessly to a serious programming experience. This environment will provide just-in-time feedback and report meaningful information to instructors. I am creating a massive collection of big data sources meant for introductory students, overcoming a number of difficult technical obstacles. Instructional materials will be created that take advantage of these tools to optimally instruct students. Motivation in the course will be primarily assessed through self-reported surveys based heavily on the MUSIC Model of Academic Motivation Inventory (MMAMI). This information will be supplemented with and compared against relevant measurements of student engagement (e.g., completion of activities, attendance in class, and course staff observations). Ability in the course will be primarily assessed

through the final semester project, designed as an authentic assessment experience: Students choose a dataset related to their major to answer questions computationally. Ability tracking will be supplemented by fine-grained student solutions to coding problems to evaluate student progress. As a major stakeholder in the course, the instructors' experience is also crucial to understanding the success of our approach. Course staff will be regularly interviewed about the automated tools used in managing a large-scale course. Students will also be surveyed to assess their perception of the courses size (e.g., whether feedback came timely enough, they had the in-class support they needed to succeed, they felt satisfied with the human interactions in the course).

6. DISSERTATION STATUS

Currently, I am writing my preliminary proposal, expecting to defend this summer before ICER. I have already published a SIGCSE paper and hosted two SIGCSE workshops on the novel technology used to integrate Big Data into introductory programming courses. My goal is to complete my dissertation work by Spring 2017, giving me approximately two years from now to implement interventions and gather data. I have helped to integrate the Big Data Science approach into a new Computational Thinking course, which has been offered twice now (first with 20 students, then with 40), enabling me to collect preliminary data (partially published at ITICSE 2015). This fall, the course will be offered to 80 students, and I expect to start gathering serious data to answer my research questions. Much of the technology I propose has been prototyped, including the block-based environment and a collection of 36 (and growing) data sources.

7. EXPECTED CONTRIBUTIONS

I anticipate that this approach to Computational Thinking will provide a useful new technique for introductory instructors. Adoption of the materials will vary widely, with some instructors taking the curriculum wholesale and some using pieces of it. The tools will be readily available and experience will indicate how they can be used most effectively.

8. REFERENCES

- [1] R. E. Anderson et al. Introductory programming meets the real world: Using real problems and data in cs1. In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education, SIGCSE '14*, pages 465–466, New York, NY, USA, 2014. ACM.
- [2] M. Guzdial and A. E. Tew. Imagineering inauthentic legitimate peripheral participation: An instructional design approach for motivating computing education. In *Proceedings of the Second International Workshop on Computing Education Research, ICER '06*, pages 51–58, New York, NY, USA, 2006. ACM.
- [3] B. D. Jones. Motivating students to engage in learning: The MUSIC model of academic motivation. *International Journal of Teaching and Learning in Higher Education*, 21(2):272–285, 2009.
- [4] Y. Matsuzawa, T. Ohata, M. Sugiura, and S. Sakai. Language migration in non-cs introductory programming through mutual language translation environment. In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education, SIGCSE '15*, pages 185–190, New York, NY, USA, 2015. ACM.