

The Application Of Ai Technologies In Stem Education: Case Of Data-Driven Intelligent Tutoring Systems

Nguyen Van Y

Ho Chi Minh City Cadre Academy (HCA), Ho Chi Minh City, Vietnam
n.vy@hcmca.edu.vn

Bui Trong Hieu

Faculty of Information Technology, Ho Chi Minh City University of Transport, Ho Chi Minh City, Vietnam
hieu.bui@ut.edu.vn

Abstract— One of the most effective ways to learn is through problem solving. Nowadays, it is widely known that problem solving is a central subject and fundamental ability in the teaching and learning. Besides, problem solving is integrated in the STEM+C (Science, Technology, Engineering, and Math plus Computing, Coding or Computer Science) fields. One of the technologies in which artificial intelligence (AI) is being implemented in the field of education is through Intelligent Tutoring Systems (ITSs). (ITSs) have been shown to be effective in supporting students' domain-level learning through guided problem solving practice. Intelligent tutoring systems provide personalized feedback (in the form of hints) to students and improve learning at effect sizes approaching that of human tutors. However, creating an ITS to adapt to individual students requires the involvement of experts to provide knowledge about both the academic domain and novice student behavior in that domain's curriculum. Creating an ITS requires time, resources, and multidisciplinary skills. Because of the large possible range of problem solving behavior for any individual topic, the amount of expert involvement required to create an effective, adaptable tutoring system can be high, especially in open-ended problem solving domains. Data-driven ITSs have shown much promise in increasing effectiveness by analyzing past data in order to quickly generate hints to individual students. However, the fundamental long term goal was to develop "better, faster, and cheaper" ITSs. The main goal of this research work is to: 1) present ITSs used in the STEM+C education; and 2) introduce data-driven ITSs for STEM+C education.

Keywords— Intelligent Tutoring Systems, STEM+C, Data-Driven Hint Generation.

I. INTRODUCTION

Problem solving is an important skill across many fields, including science, technology, engineering, and math plus computing, coding or computer science (STEM+C). Working open-ended problems may encourage learning in higher 'levels' of cognitive domains. Intelligent tutors have been shown to be as effective as human tutors in supporting learning in

many domains, in part because of individualized, immediate feedback in the form of hints, enabled by expert systems which diagnose the knowledge state of the student. An additional benefit of computer-based environments is that they record extensive logs of student work, at a detail otherwise not possible [1]. According to Freeman et al. [2], within STEM+C fields, most students are expected to have skills in computer programming after completing their university studies, while many other faculties are also beginning to encourage education in this area. However, many students find great difficulty with the learning of programming and it becomes a barrier to their further studies of computer science and other disciplines. This difficulty is in large part due to students' inability to solve their programming exercises, and this may discourage them to progress further when help can be obtained immediately. In order to address this problem, various approaches have been proposed to help students learn solving programming exercises.

Traditionally, face-to-face and one-to-one human tutoring had been the best option for tutor. However, human tutors are not always available and that's why computer based tutoring is developed to provide as an alternative support. Intelligent Tutoring System (ITS) is an example of computer-based tutoring which is developed emulating the human tutor. ITSs can provide personalized feedback to students automatically, but they can take large amounts of time and expert knowledge to build, especially when determining how to give students hints. Data-driven approaches can be used to provide personalized next-step hints automatically and at scale, by mining previous students' solutions. Instead of taking much time for modeling domain knowledge, the data-driven approach uses a mass of correct student programs. The data-driven approach uses correct student solutions in order to construct a solution space that contains all solution states students have created in the past (e.g., in the former semesters of a programming course).

II. BACKGROUND

II.1 Intelligent Tutoring Systems

It is a well-established fact that face-to-face and one-to-one human tutoring is the best tutoring field. However, it is extremely expensive in terms of both physical and human resources. ITSs are a natural

solution that can be used to address this problem, as they are developed to give personalized feedback and help to students who are working on problems. The fact the ITSs are formed by three fields: Computer Science, Psychology, and Education, as illustrated in Figure 1, in which, (i) Artificial Intelligence (AI) addresses how to reason about intelligence and thus learning, (ii) Psychology (Cognitive Science) addresses how people think and learn, and (iii) Education focuses on how to best support teaching/learning [4].

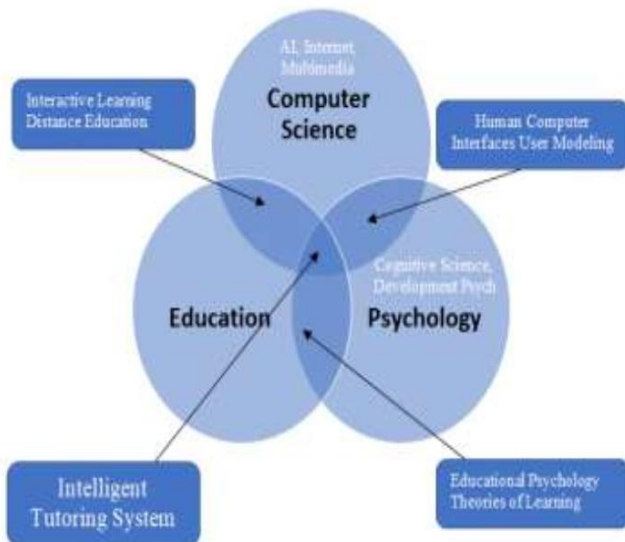


Fig. 1. The development of an Intelligent Tutoring System using methods and instruments from three different domains

According to Lee & Chen [5], an Intelligent Tutoring System (ITS) is a computer system that provides immediate and customized instruction or feedback to learners. The classical architecture of an Intelligent Tutoring System includes the following four components (Figure 2) [6, 7, 8, 9].

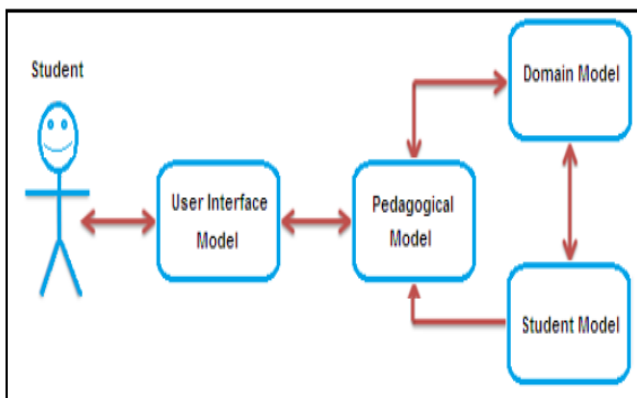


Fig. 2. The typical architecture of an ITS

This traditional view of ITSs is still very accepted by the ITS community. However, recent studies stress functionality over structure [6, 7, 8, 9], describing ITSs as having two main loops [10]: 1) the inner loop and 2) the outer loop (Figure 3). The inner loop is responsible for providing personalized feedback, hints, and direct

problem solving assistance to students. The inner loop also assesses students' competence and registers it on the student model. Using the information that is obtained about the student, the outer loop performs task selection.

Until tutoring is done, do:

1. *Tutor poses a task*
2. *Until the task is achieved, do:*
 - 2.1. *Tutor may give a hint*
 - 2.2. *Student does a step*
 - 2.3. *Tutor gives feedback on the step*
3. *Student submits the solution to the task*

Fig. 3. ITS Loops

This work is inspired from VanLehn's [11] two loop characterization of tutoring systems. The main task of the outer loop is to select an appropriate programming exercise for the student. The inner loop is responsible for giving hints on student steps.

Here, this research work focus on the inner loop. This work does not support an outer loop which can create an overall student model and intelligently choose which programming exercises to show to the student.

Nesbit and colleagues [12], in their paper, "Work in Progress: Intelligent Tutoring Systems in Computer Science and Software Engineering Education", research on ITSs has accelerated over the last decade, and scholarly interest in such systems has never been greater. ITS have been developed for a wide range of subject domains (e.g., mathematics, physics, biology, medicine, reading, languages, philosophy, information technology and computer science) and for students in primary, secondary and postsecondary levels of education. Founded on several decades of research on human cognition and intelligence, Intelligent Tutoring System is now a fast growing area in academia and industry. We now turn our attention to some research on ITS in the STEM+C fields.

II.2 Intelligent Tutoring Systems and STEM+C

According to Graesser et al. [13], ITSs have been developed for nearly four decades on many STEM+C topics. ITSs have been developed for a wide range of STEM+C subject matters. Many have targeted mathematics and other well-formed, quantitatively precise topics. Besides, most applications of ITS appear to concern STEM education and training [14].

Open-ended tasks are common in STEM+C education. Problem solving, or, more generally, working in open-ended tasks, is commonly thought to be a fundamental activity in learning STEM+C, either as an educational goal in itself or as a way to develop scientific and mathematical skills. However, instructors

face two main difficulties: assessment is time-consuming, and the steps followed by the students to obtain a solution is hardly available even when detailed reports are requested. To obtain relevant content-related assessment data, ITS applications are designed to capture user interactions by tracing or logging user data. This provides a powerful source of information for both psychological and educational research, as well as for generating useful feedback for users and instructors [15].

According to [16], The challenge's goal was to develop ITSs for middle/high schools within the STEM+C education area that improve student retention, reasoning, and problem solving by at least two standard deviations. One salient benefit would be that the resultant technologies could be easily modified to support Navy training. However, the fundamental long-term goal was to develop "better, faster, and cheaper" ITSs.

II.3 Data-Driven Intelligent Tutoring Systems

As mentioned by [3], data-driven ITS is a subfield of ITS where decision-making is based on the previous student's work instead of a knowledge base built by experts or an author-mapped graph of all possible paths. Successful solutions from the past can be used to provide feedback and hints for students in the present, which circumvents the need to create an expert model. A data-driven tutoring system can be bootstrapped by experts providing missing data. The data-driven approach has proven to work well in combination with artificial intelligence and machine-learning techniques for learning an expert model by demonstration.

III. RELATED WORK

III.1 Intelligent Tutoring Systems for STEM+C education

The four primary articles are followed by a commentary by Fletcher [14] which reflects on these developments as well as the state of ITS in general and assessments of these systems on learning gains.

The first article is SKOPE-IT (Shareable Knowledge Objects as Portable Intelligent Tutors): Overlaying Natural Language Tutoring on an Adaptive Learning

System for Mathematics by Nye et al. [17] (Figure 4). SKOPE-IT is a natural language tutoring system on mathematics. This system is a hybrid of two highly effective. The first system, ALEKS (Assessment and Learning in Knowledge Spaces), is a mastery-based tutoring system based off of Knowledge Space Theory [18, 19, 20]. To produce SKOPE-IT, the ALEKS architecture was combined with the AutoTutor Conversation Engine [21]. This version of AutoTutor is a web service that allows for natural language conversations with one or more conversational agents. The article describes an evaluation comparing the SKOPE-IT system versus the ALEKS system with college-level algebra students.

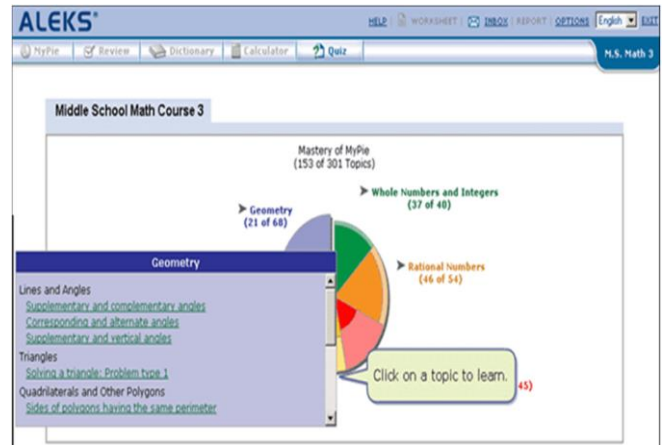


Fig. 4. Student interaction in ALEKS centers around selecting a skill to master, which is done by selecting an available skill from a "pie" which shows groups of related skills

The second article by Inventado et al. [22], Contextual Factors Affecting Hint Utility, describes the use of a new hint utility within a mathematics-based tutoring system, ASSISTments. It describes a randomized trial of the system with the hint utility. The article provides evidence for the impact of the hint utility, a case study of how randomized trials can be implemented with ASSISTments, and lessons learned from the implementation.

The third article by Skinner et al. [23], Development and Application of a Multi-Modal Task Analysis to Support Intelligent Tutoring of Complex, describes the development of an Intelligent Tutoring System for training Robotic Assisted Laparoscopic Surgery (RALS) (Figure 5). This article provides a strong example for the potential that increasing the consideration of the end user, usability, and human factors/human systems can have for improving educational technology [24]. The argument is made for the use of cognitive task analysis (CTA) methods for gathering reliable training corpora. However, the article also argues for the need for a new type of CTA called multi-modal task analysis (MMTA) that elicits knowledge for cognitive, psychomotor, and perceptual skills from experts. The article provides an example of this new technique and how it was used within the RALS training tasks that model the RALS skills.

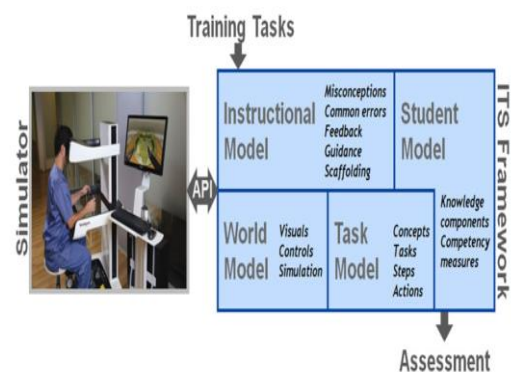


Fig. 5. High-level design of RALS ITS

The fourth article by Graesser et al. [13], ElectronixTutor: An intelligent tutoring system with multiple learning resources for electronics, was a monumental contribution by 25 authors from academia and industry across eight different institutions. The ElectronixTutor (Figure 6) plays off the success and lessons learned from the SKOPE-IT project [17] to create an intelligent learning resource that integrates elements from five highly successful intelligent Tutoring Systems (AutoTutor [21], Dragoon [25], LearnForm[26], ASSISTments [27], BEETLE-II [28]) as well as traditional reading of text materials. The article is both a review paper and an example case for system integration (i.e., ElectronixTutor). The article provides a summary of research for each component system and best practices for ITS development. The article also provides an example case for system integration within the educational technology area. It provides a wealth of history and lessons learned from a wide section of the Intelligent Tutoring System Literature.

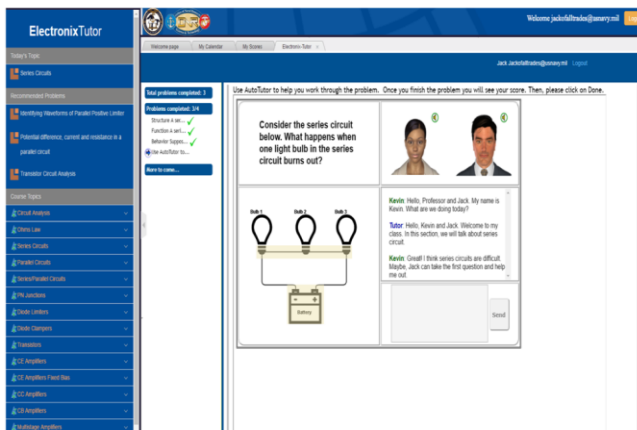


Fig. 6. ElectronixTutor user interface snapshot

In the other hand, as noted by Fletcher [14], Some common themes are suggested by these reflections and more importantly by the four foundational articles that were contributed to this issue by a remarkably wide collection of experienced ITS researchers and developers. These themes might include the following: 1) there is an abiding need for individualization in all learning, including education and training in STEM+C subjects; 2) STEM+C education and training are natural and already widely used applications for ITS, well deserving of continued attention and development; 3) ITS offers an affordable means to provide individualization. But subject matter rudiments may also be provided by drill and practice, which may be a more cost-effective approach for these items. An argument can be made for pairing drill and practice techniques with ITS so that each is used to best advantage in education and training; 4) analysis to determine objectives and standards for learning is as critical for development of ITS in STEM+C topics as it is elsewhere. It deserves full and comprehensive attention—including considerations of context [22] and non-cognitive modalities such as those involving psychomotor and perceptual activity [23]; 5) Also

indicated by Skinner et al. [23] was the possibility and value of assigning objectives and standards for ITS intended to accelerate the acquisition of STEM+C expertise, beyond novice and journeymen levels of knowledge and skill, without increasing time in instruction; 6) Natural language dialogue [17] possibly including a second computer-generated participant [13] is a valuable and worthy capability provided by ITS for STEM+C instruction; 8) It may be time to pursue the approach used by Graesser et al. [13] of combining the best approaches and capabilities of various, existing ITS to design and build ITS for STEM+C. Doing so may well reduce the cost in time and effort of produce ITS—a goal of a number of funding programs.

III.2 Data-Driven Hint Generation in ITSs for STEM+C education

Using large sets of historical student data to generate hints is a recent development that has already produced some promising results. Due to lengthy development time for an ITS, several researchers have tried to generate hints from past student data. Data-driven tutors reduce the necessary effort even further by mining educational data to generate feedback in the form of hints. They construct an implicit domain from solutions submitted by students. In most cases, feedback is still generated from the differences between the student's program and a previously submitted solution. The use of data-driven methods to develop intelligent tutoring systems is just starting to be explored in the field [29]. Authors of data-driven systems argue that these approaches avoid the need for experts to spend time constructing complex domain models and can lead to additional insights that experts alone would not achieve [30]. In the other words, creation of adaptive educational programs is costly. This is, in part, because developing content for intelligent tutors requires multiple areas of expertise. Content experts and pedagogical experts must work with tutor developers to identify the skills students are applying and the associated feedback to deliver [31]. The Figure 7 below describes in brief what the model in data-driven ITSs for STEM+C education [32].

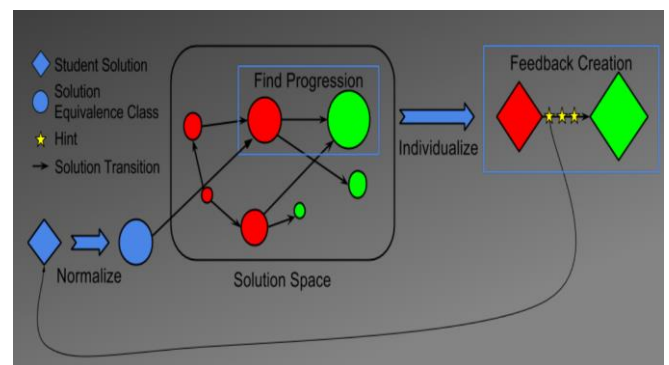


Fig. 7. The Data-driven hint generation in ITSs for STEM+C

IV. CONCLUSION AND RECOMMENDATION

The main goal of this paper is to: 1) presents recent ITSs used in the STEM+C education; and 2) introduce data-driven ITSs for STEM+C education. Besides, this study has surveyed various research paper broadly based on the role of ITSs in STEM+C education field. In the context of data-driven ITSs for STEM+C education, despite the research efforts in recent years, however, generating data-driven hints is still having some problems. In summary, in this work, one gap the author identified that provide the motivation for future researches are listed below: data-driven ITSs for STEM+C education has been expanding as a subfield of ITSs over the past few years, with many different researchers creating new techniques to automatically generate hints. However, most of the systems have only been evaluated on collected student problem-solving traces, and the ones that are being tested on real students are implemented in online learning environments such as MOOCs (massive open online courses), not in individual classrooms. In the context of curriculum and real classroom in an ITS, this indicates that there is significant room for improvement in the field of data-driven ITS for STEM+C education.

REFERENCES

- [1]Eagle, Michael, et al. "Exploring networks of problem-solving interactions." Proceedings of the Fifth International Conference on Learning Analytics And Knowledge. ACM, 2015.
- [2]Freeman, Paul, Ian Watson, and Paul Denny. "Inferring student coding goals using abstract syntax trees." International Conference on Case-Based Reasoning. Springer, Cham, 2016.
- [3]Bui, Hieu Trong, and Syed Malek FD Syed Mustapha. "Automated Data-Driven Hint Generation in Intelligent Tutoring Systems for Code-Writing: On the Road of Future Research." International Journal of Emerging Technologies in Learning (iJET) 13.09 (2018): 174-189.
- [4]W. B. Park, Building intelligent interactive tutors: Student-centered strategies for revolutionizing e-learning, Morgan Kaufmann, 2010.
- [5]L. R. Maria and T. T. Chen, "Digital creativity: Research themes and framework," Journal of Computers in Human Behavior 42, 2015, pp. 12-19.
- [6]Aleven, Vincent, et al. "The cognitive tutor authoring tools (CTAT): Preliminary evaluation of efficiency gains." International Conference on Intelligent Tutoring Systems. Springer, Berlin, Heidelberg, 2006.
- [7]Barnes, Tiffany, and John Stamper. "Toward automatic hint generation for logic proof tutoring using historical student data." International Conference on Intelligent Tutoring Systems. Springer, Berlin, Heidelberg, 2008.
- [8]W. B. Park, Building intelligent interactive tutors: Student-centered strategies for revolutionizing e-learning, Morgan Kaufmann, 2010.
- [9]L. R. Maria and T. T. Chen, "Digital creativity: Research themes and framework," Journal of Computers in Human Behavior 42, 2015, pp. 12-19.
- [10]S. G. Soares and J. Jorge, "Interoperable intelligent tutoring systems as open educational resources," Journal of IEEE Transactions on Learning Technologies 6(3), 2013, pp. 271-282.
- [11]Vanlehn, Kurt. "The behavior of tutoring systems." International journal of artificial intelligence in education 16.3 (2006): 227-265.
- [12]J. C. Nesbit, Q. L. A. Liu and O. O. Adesope, "Work in Progress: Intelligent Tutoring Systems in Computer Science and Software Engineering Education," Proceeding 122nd Am. Soc. Eng. Education Ann, 2015.
- [13]Graesser, Arthur C., et al. "ElectronixTutor: an intelligent tutoring system with multiple learning resources for electronics." International journal of STEM education 5.1 (2018): 15.
- [14]Fletcher, J. D. "Comments and reflections on ITS and STEM education and training." International journal of STEM education 5.1 (2018): 16.
- [15]Calvo, Miquel, et al. "Computer-Assisted Assessment in Open-Ended Activities through the Analysis of Traces: A Proof of Concept in Statistics with R Commander." Eurasia Journal of Mathematics, Science and Technology Education 15.9 (2019).
- [16]Craig, Scotty D., Arthur C. Graesser, and Ray S. Perez. "Advances from the Office of Naval Research STEM Grand Challenge: expanding the boundaries of intelligent tutoring systems." International journal of STEM education 5.1 (2018): 11.
- [17]Nye, Benjamin D., et al. "SKOPE-IT (Shareable Knowledge Objects as Portable Intelligent Tutors): overlaying natural language tutoring on an adaptive learning system for mathematics." International journal of STEM education 5.1 (2018): 12.
- [18]Craig, Scotty D., et al. "The impact of a technology-based mathematics after-school program using ALEKS on student's knowledge and behaviors." Computers & Education 68 (2013): 495-504.
- [19]Falmagne, Jean-Claude, et al., eds. Knowledge spaces: Applications in education. Springer Science & Business Media, 2013.
- [20]Hu, Xiangen, et al. "The Effects of a Traditional and Technology-based After-school Setting on 6th Grade Student's Mathematics Skills." Journal of Computers in Mathematics and Science Teaching 31.1 (2012): 17-38.
- [21]Graesser, Arthur C. "Conversations with AutoTutor help students learn." International Journal of Artificial Intelligence in Education 26.1 (2016): 124-132.

[22]Inventado, Paul Salvador, et al. "Contextual factors affecting hint utility." *International journal of STEM education* 5.1 (2018): 13.

[23]Skinner, Anna, et al. "Development and application of a multi-modal task analysis to support intelligent tutoring of complex skills." *International journal of STEM education* 5.1 (2018): 14.

[24]Roscoe, Rod D., Scotty D. Craig, and Ian Douglas, eds. *End-user considerations in educational technology design*. IGI Global, 2017.

[25]VanLehn, Kurt, et al. "Learning science by constructing models: can dragoon increase learning without increasing the time required?." *International Journal of Artificial Intelligence in Education* 26.4 (2016): 1033-1068.

[26]Kumar, Rohit, et al. "First evaluation of the physics instantiation of a problem-solving-based online learning platform." *International conference on artificial intelligence in education*. Springer, Cham, 2015.

[27]Beal, Carole R., et al. "On-line tutoring for math achievement testing: A controlled evaluation." *Journal of Interactive Online Learning* 6.1 (2007): 43-55.

[28]Dzikovska, Myroslava, et al. "BEETLE II: Deep natural language understanding and automatic feedback generation for intelligent tutoring in basic electricity and electronics." *International Journal of Artificial Intelligence in Education* 24.3 (2014): 284-332.

[29]Mostafavi, Behrooz, and Tiffany Barnes. "Evolution of an intelligent deductive logic tutor using data-driven elements." *International Journal of Artificial Intelligence in Education* 27.1 (2017): 5-36.

[30]Price, Thomas W., et al. "The Impact of Data Quantity a Source on the Quality of Data-Driven Hints for Programming." *International Conference on Artificial Intelligence in Education*. Springer, Cham, 2018.

[31]Eagle, Michael John. *Data-Driven Methods for Deriving Insight from Educational Problem Solving Environments*. North Carolina State University, 2015.

[32]K. Rivers "Automated Data-Driven Hint Generation for Learning Programming". PhD thesis, Human-Computer Interaction Institute, School of Computer Science, Carnegie Mellon University, USA, 2017.