Motivating Student's Programming Language Study by Applying Second Language Acquisition

Dr. Lei Gu[1], Dr. Lulu Sun[2]

1. Georgia State University 2. Embry-Riddle Aeronautical University.

Abstract

Computing Fundamentals for Engineers (ENGR1671) is a fundamental engineering course at a two-year college in southeast region of U.S. MATLAB was introduced in this course as a programming language to solve engineering problems. In fall 2017, a NSF sponsored project: Applying Second Language Acquisition to Facilitate a Blended Learning of Programming Languages (SLA-aBLe), which was developed by a private institution in the southeast region of U.S. was implemented in ENGR1671 to motivate student's programming language study for the first time. Due to the diverse background of student population, traditional lecture and lab was still the main teaching method. Students' academic performance, and perceptions toward the SLA-aBLe project materials were analyzed and compared with students who took the course in previous semesters without using SLA-aBLe materials. The goal of this study is to improve the engagement and learning experience of engineering students enrolled in an introduction to computing and programming course.

Keywords

Blended Learning, programming Language, MATLAB, engineering, second language acquisition

Introduction

Regardless the major, all engineering students need to take at least one programming language in their first two years study in college [1]. Different programming languages for engineering courses are offered at the college level including C++, Java, and MATLAB. The goal of engineering students at Perimeter College at Georgia State University (GSU-PC) is to transfer to a four-year engineering institution in Georgia through the Regents' Engineering Pathways (REP) Program after completion of curriculum requirements at GSU-PC. Computing Fundamentals for Engineers course (ENGR 1671) at GSU-PC is a required MATLAB programming course for engineering students who plan to transfer to Georgia Institute of Technology through REP program. Although the course is a 3-credit-hour course, the engineering department offers extra one credit hour lab to help students be successful. The class meets twice a week, and each class period is 1 hour and 45 minutes including 75 minutes traditional lecture and 30 minutes lab. Majority of students have very limited programming experience before taking this course. Undergraduate students find learning a programming language to be difficult, especially without previous exposure. In fall 2017, an NSF sponsored project: Applying Second Language Acquisition to Facilitate a Blended Learning of Programming Languages (SLA-aBLe), which was developed by a private institution in the southeast, US was implemented in two sections as supplemental tools to motivate student's programming language study. This NSF funded project applied second language

acquisition to facilitate an introductory programming language study using MATLAB in a blended learning environment. SLA-aBLe courses utilized a framework that stresses fluency by dividing course content into five main stages: preproduction, early production, speech emergence, intermediate fluency, and advanced fluency. When paired with familiar language rules and syntax, students find it easier to learn programming languages when they are presented from an SLA orientation and demonstrate deeper levels of understanding than would be found following rote memory tasks [2-8].

Methodology

At GSU-PC, the textbook of Computing Fundamentals for Engineers course (ENGR 1671) is 'Matlab: An Introduction with Application' by Amos Gilat published by Wiley. Ten chapters from the textbook are covered in ENGR 1671 including array, data types, data management, plotting, conditional statements, loops, function and numerical analysis. Several chapters including character strings, cell arrays, structures and high-lever I/O functions from textbook 'Engineering Computations with MATLAB' by David M. Smith at Georgia Institute of Technology are also covered to compile with teaching guideline of Computing for Engineers (CS 1371) at Georgia Institute of Technology. There are two sections of ENGR 1671 offered in fall 2017 with a final enrollment of 42 students. The SLA-aBLe project contains tutorial screencasts on EDpuzzle.com with embedded quiz problems and lab assignments to reinforce the comprehension in ENGR 1671 except numerical analysis and function concepts. Students could view screencasts multiple times until they fully understand the material, and then complete the interactive quizzes online. Solutions are explained in the EDpuzzle after quiz submission.

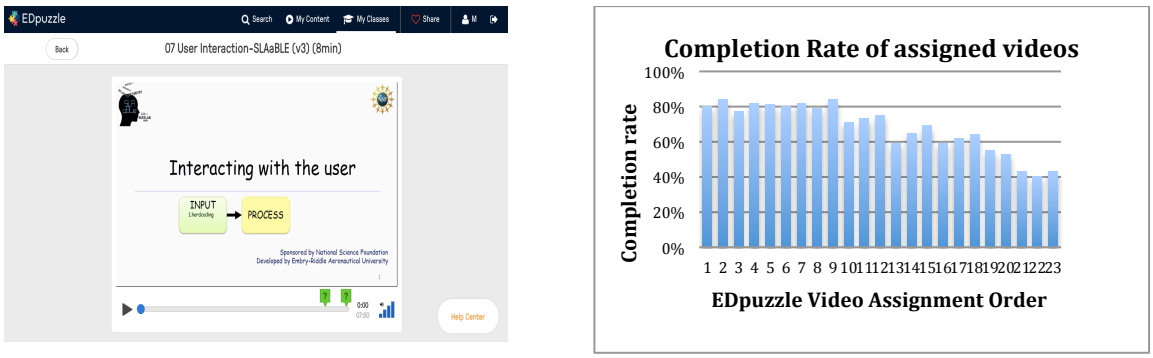
Nowadays, many universities are using flipped classroom or hybrid pedagogy with prerecorded videos in an engineering field. A flipped or inverted course design requires students to watch videos, read materials, or complete basic assignments outside of the traditional lecture setting [9], and lab sessions are offered in class. However, due to the diverse background of students in two-year-college, flipping classroom was not a success based on author's teaching experience in Statics at GSU-PC. Therefore, EDpuzzle videos were offered as supplemental learning tools. In the classroom, traditional lecture embedded with the lab assignments from the textbook and the SLA-aBLe lab exercises were offered. Students experienced think-pair-share in the lab time. They tried to complete the program by themselves first and then grouped to share ideas, the instructor facilitated the discussion in the lab time as well. After class, more complicated homework assignments were given to reinforce the level of comprehension at the intermediate fluency stage.

Results

According to the course plan, the author assigned the following screencasts as shown in Table 1 to students before introducing the materials in class. One snapshot is shown in Figure 1.a. 'Prevent skipping' is the setting for all videos. Completion rates of video assignment are shown in Figure 1.b. It has noticeable decreasing completion rate based on the order of the assignments.

Table 1. Video Assignment list

Video Assignment name	Video Assignment Order
06 Script Files-SLAaBLE (v3)	1
07 User Interaction-SLAaBLE (v3) (8min)	2
08 Numerical Inputs-SLAaBLE (v3) (7min)	3
09 Char Input (v3) (7min)	4
10 Basic Output (v3) (15min)	5
11 Formatted Output (v3) (13min)	6
12 Intro to Logic-SLAaBLE (v2) (6min)	7
14 Boolean Operators-SLAaBLE (v2) (8min)	8
15 Order Of All Operators-SLAaBLE (v2) (6min)	9
16 If Statements Basic (v3) (14min)	10
17 If Statement - TensionCable (v3) (10min)	11
18 If Statement - elseif (v3) (11min)	12
19 If Statements - Nested (v3) (13min)	13
20 Loops Intro (v3) (10min)	14
21 While Loops General (v3) (9min)	15
22 While Loops -Trap Invalid Inputs (v3) (13min)	16
23 While Loops - Running Total/Product (v3) (15min)	17
24 For Loops Intro (v3) (15min)	18
25 Nested Loops (v3) (13min)	19
Vectors03 Plotting - Basics (v1) (13min)	20
Vectors04 Plot Straight Lines (v1) (5min)	21
Vectors05 Plot Curved Lines (v1) (17min)	22
Vectors06 Combining Plots (v1) (10min)	23



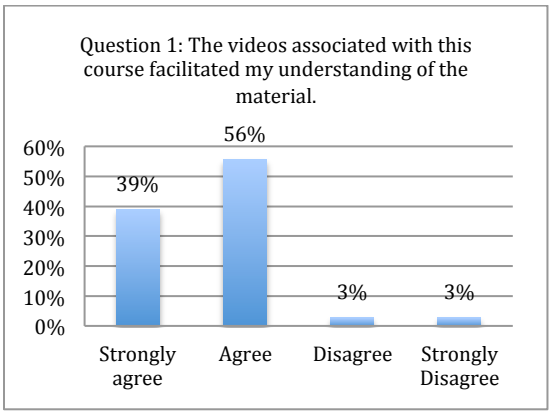
(a) (b)

Figure 1. Sample EDpuzzle screencast assignment and Completion rates vs. screencast assignment order

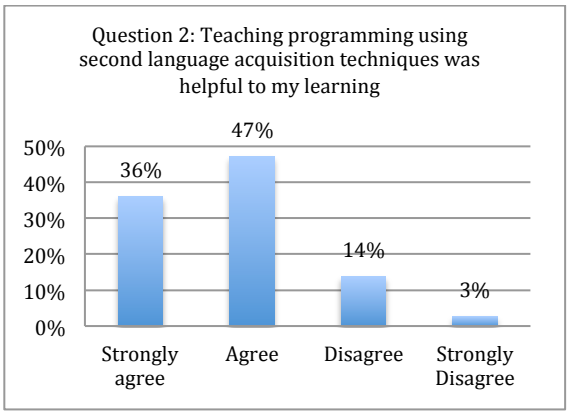
To seek a full understanding of the effectiveness of using the SLA-aBLe project learning materials in ENGR1671, a survey was conducted among students in two sections in 2017 [10]. Eighty-six percent of students (n= 42) completed the survey. From the survey, students from two SLA-aBLe course sections in fall 2017 suggested that the SLA-aBLe

materials were effectively designed. Students indicated the teaching techniques (traditional lectures with SLA-aBLe learning materials) used in this class are engaging. Some question responses are shown in Figure 2 to Figure 5.

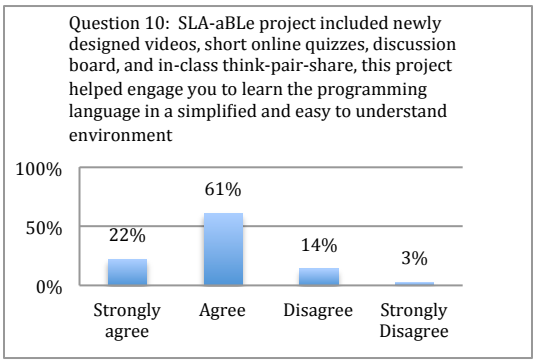
Figure 2.a demonstrates that about 94 % of the students either strongly agree or agree that the videos associated with this course facilitated their understanding of the material. Figure 2.b demonstrates that about 93 % of the students strongly agree or agree that teaching programming using SLA-aBLE was helpful to their learning**.** In Figure 2.c, 83% of the students either strongly agree or agree that SLA-aBLe helped engage them to learn the programming language in a simplified and easy to understand environment**.** Figure 2.d demonstrates that about 44 % of students prefer the instructional videos on the EDpuzzle site, only 11% of students prefer the instructional videos not on the EDpuzzle site and about 44% of students have no preference.



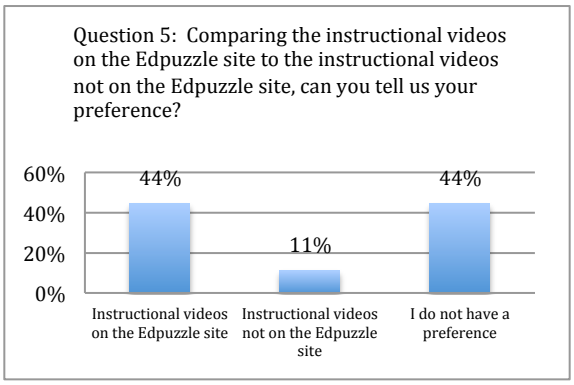
(a)



(b)



(c)



(d)

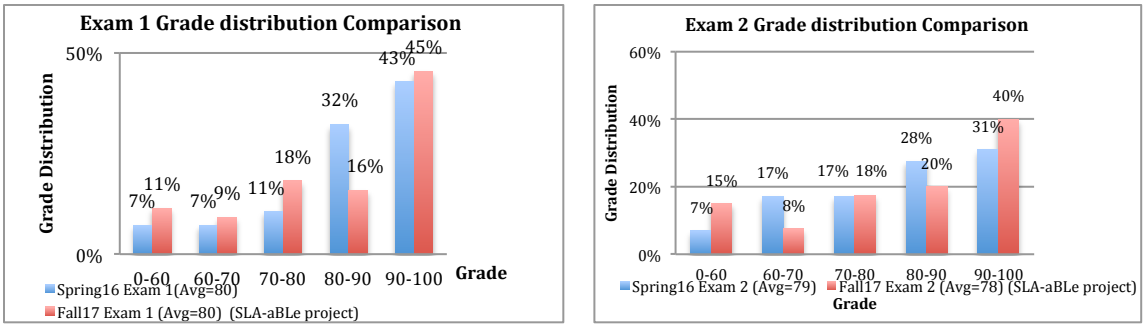
Figure 2. Survey questions responses in fall 2017 semester

Original comments and suggestion to the SLA-aBLe are listed below without grammar correction.

- I felt that the pacing of the quizzes could be spaced out just a bit. I felt like I had to remember what we were talking about before working the problem, before I could process the content about to covered.

- Good job!
- The videos were very helpful in that it made you practice coding and problem solving skills.
- I use online resources to study before the lecture or study before the exam. The major issue I have with EDpuzzle is that the video is FIXED to a play rate. There are topics that I wish to fast-forward and some topics that I wish to go over again and I cannot do that with EDpuzzle. I end up using the same video on YouTube to learn at my own pace. The speaker talks too slow so I use YouTube to play at 1.5x rate. Also if incorrect code in submitted, it will not tell you syntax errors.
- Please create hints to the questions provided or multiple attempts.
- The only criticism I have is that the videos sometimes wouldn't specify that you needed to hold on to the code, then certain code was revisited in later videos and the way the videos were named and allotted sometimes made it time-consuming to figure out which video I needed to watch first.
- More videos covering all topics
- Yeah, complete the course according to Amos Gilat. There are no videos for the last chapters.
- Be more clear on the lectures.
- Video pauses if open second tab(s).

In fall 2017, students have taken two exams by October. All materials in exam 1 and exam 2 were covered in the SLA-aBLe project. The exams have same contents and difficulty as in spring 2016. The author at Perimeter College graded all exams with the same rubric. The grades distributions and average grades from fall 2017 and spring 2016 were compared as shown in figure 3. There is no significant difference between the average grades from these two semesters. However, 40% of students got A in fall 2017, but only 31% of students received A in spring 2016. Possible reason is that students with strong motivation have completed more video assignments.



(a) (b)

Figure 3. Exam grade distribution comparison

Conclusion

This is a pilot study of applying NSF funded Second Language Acquisition to Facilitate a Blended Learning of Programming Languages (SLA-aBLe) for MATLAB course in a

two year college. The majority of students agreed that SLA-aBLe is an effective learning tool for learning programming language. Because the video assignment completion rate decreased over time, increasing motivation and reaching weak students are critical for successful implementation of SLA-aBLe project. Meanwhile, students prefer having videos with shorter length. Students with strong motivation have completed more video assignments and therefore boosted up their grade from B to A. However the project did not help students who did not watch the videos. Larger sample population can provide more data in future study.

Reference

1. Azemi, A., Pauley, L. *Teaching the Introductory Computer-Programming Course for Engineering Using MATLAB and Some Exposure to C*. in *ASEE Annual Conference*. 2006.
2. Sun, L., Frederick, C., Espejo, P. S., & Cunningham, R. M., *Can We Teach a Programming Language as a Second Language?* Computer in Education Journal, 2016. **7.7**(3).
3. Solomon, J., *Programming as a Second Language.* Learning & Leading with Technology, 2004. **32**(4): p. 34-39.
4. Wynn, M. *Ky. Ponders Teaching Computer Code as Foreign Language*. 2015 January 29, 2015; Available from: http://www.usatoday.com/story/tech/2015/01/29/ky-computer-code-as-foreign-language/22529629/
5. Sun, L.F., Tina. *Applying Second Language Acquisition to Facilitate a Blended Learning of Programming Languages*. in *ASEE Annual Conference*. 2015. Seattle, WA.
6. Sun, L., Frederick, C., Ding, L., & Rohmeyer, R. *The Application of Second Language Acquisition to Programming Language Study*. in *ASEE Annual Conference*. 2017. ASEE Annual Conference.
7. Frederick, C., Sun, L., *Work in Progress: Using Second Language Acquisition Techniques to Teach Programming - Results from a Two-Year Project*, in *ASEE Annual Conference*. 2017: Columbus, OH.
8. Frederick, C., Pierce, M., Griggs, A., Sun, L., Ding, L. . *Get Rid of Your Student's Fear and Intimidation of learning a Programming Language*. in *9th First Year Engineering Experience (FYEE) Conference*. 2017. Daytona Beach, FL.
9. Jensen, M.J., Howard, A., Jensen, S. . *Flipped Classes: Do Instructors Need To Reinvent the Wheel When It Comes To Course Content?* in *ASEE Annual Conference & Exposition*. 2015. Seattle, WA.
10. Cunningham, R., Espejo, P. S., Frederick, C., Sun, L., Ding, L. *A Second Language Acquisition Approach to Learning Programming Languages*. in *2016 ASEE Southeast Section Conference*. 2016.