Region-Educational Research and Reviews, 2025,Volume 7, Issue 4
DOI:10.12238/rerr.v7i4.3856
ISSN Online:2661-4634
ISSN Print:2661-4626

A Comparative Study of Learning Behaviors in Online and Traditional Teaching of Software Skills Courses

Zhixuan Zhang

School of Innovation and Creation Design, Shenzhen Polytechnic University, Shenzhen, China

Abstract: Today's students, who grew up in the digital era, show great dexterity in using digital and internet-based tools, thus building a solid base for transferring courses online. Online instruction has now become a prominent dimension of university education and a chief pillar of curricular reform and innovation. An example would be the rise of Massive Open Online Courses (MOOCs), which provided limitless opportunities for millions of learners to access excellent resources online, enabling students to interact with world-class institutions through online courses. Also, with online education on the rise, modern technology can allow for interaction, discussion, collaboration, and assessment to take place in online classrooms, essentially addressing some of the limitations that online teaching used to have when compared to traditional forms of teaching. In this sense, as tides have turned, conventional teaching models are ever-further challenged by their virtual variants, facilitating the experimentation of so-called hybrid online-offline courses aimed at redefining classical university education.

Keywords: comparative study, learning, teaching, behaviors

Introduction

For specific fields, such as art design and engineering design, courses of software skills act as major cornerstones, which have a great impact on students' ability to engage in the workplace. This is especially true in application-oriented universities and more vocational undergraduate schools, where student's ability with software directly correlates with the employability of graduates and the program's broader reputation in society. Employers tend to have high expectations for graduates' proficiency in professional software and often use it as a litmus test in hiring. For small and medium-sized enterprises in particular, it's a top priority to ensure that a candidate can wield professional software to meet basic job requirements[1].So, while they are labeled as foundational, software skills courses are inextricably linked to students' career prospects. For students at applied and vocational institutions, how good their software skills are determines what kind of jobs they will get.

1. The Importance of Course Design

Software skills courses are of utmost importance but there are challenges in delivering them. The rapid iteration of software means that course content must keep pace with technology, and so courses may need to be updated more frequently than other standard foundation courses. Second, as they are closely integrated into industry practices, software skills raise the bar for teachers. Course content ideally needs to be relevant to a range of industry practice and hence technology, but teaching this requires educators to be fluent not only in the use of specialised software (most professional practitioners are not and/or use very old versions that become less relevant over time), but to be able to troubleshoot, and understand the nature of the problems and the logic of the design behind the software.

Take art design programs and look at the inclusion of software skills courses. Professionals in these sorts of fields need to be proficient in advanced software, since much of their work consists of using specific tools to bring designs to life. Without excellent software skills, it becomes extremely difficult to express design ideas well. It is consequently essential to improve the quality of teaching of these courses. Currently, the majority of universities offer software skills courses in the traditional offline format, based on a teaching model in which lecturers explain the fundamental principles of software tools and lead students through the process of applying them in case studies. But the huge societal demand for software skills has given rise to commercial providers of online software skills courses, which can be in the form of live streaming or pre-recorded videos[2]. In particular, researching online and traditional learning behaviors through surveys, interviews, and observations to compare and contrast online learning behaviors with traditional learning behaviors will further empower instructors to improve the quality of their teaching. Using each of these strategies in concert gives teachers the tools they need to create better learning experiences.

Applying to design programs, you could categorize software skills as operational and programming. Luxurious as criticality sounds, it is an alien practice in most tools like Rhinoceros and Keyshot (working at the intersection of product design), or Photoshop and Illustrator (these being tent poles of visual communication) whose milieu is about an aggressive practical mastery; or JavaScript and Python (the preoccupations of the digital media and interaction design) insisting on logical proof and conceptual knowing. An experiment was conducted by taking a currently highly used analytical and descriptive tool in the industrial and product design, Rhinoceros, for comparison study between the teaching online and traditionally learning behaviors. Two classes of students with equivalent proficiency and no prior exposure to the material were grouped into cohorts: one cohort received traditional in-person instruction while the other cohort received online instruction. Both groups had the same amount of instructional hours, and received identical content that included tool principles and practical applications. Shortened responses were measured by identifying learned behavior and conducted interviews and surveys to capture how effective the modes proved to be. The quality of the teaching was assessed through two quantitative measures: time taken to complete a task and the accuracy of software operation[3]. The instructional delivery was organized in three stages including theoretical knowledge sharing; practical case implementation; learning achievement evaluation.

In the theoretical knowledge delivery phase, some students in the traditional group found it difficult to concentrate, while the online group noted pausing the videos repeatedly. In the traditional format, learning time was identical for all subjects, while the study group's learning times varied widely: In the online setting, students doubled up their amount of learning time by pausing, rewinding or changing their playback speed. Follow-up lessons indicated a higher satisfaction rate among online learners, with traditional learners citing a wish for more control over their rate of progress. Within the practical case application phase, on the whole, traditional learners indicated slightly greater satisfaction than online learners, although some online students noted that receiving delayed feedback is a drawback. Online learners spent more time on task than their traditional counterparts consistently. Some traditional learners expressed a desire for self-paced exercises, noting that a one-size-fits-all approach to classroom instruction was limiting their self-ownership, with professors dictating the pace. Others in the traditional cohort felt they needed a recap of the theoretical content, having forgotten previous explanations during practice. Across e-learners, performance split wide: self-driven students achieved their goals while their less driven peers struggled with distraction, switching tabs obsessively, changing interfaces, or pausing videos for long stretches at a time, and producing poor-quality work.

In the learning outcome assessment phase, the traditional learners displayed more classroom vitality, exhibiting significantly higher instructor-student interaction rates and peer-peer interactions than the online group. While using an online platform, the same stage was again most distracting, but it had been engaged with by fewer participants in the online setting compared to before. This meant that the offline part was shining brighter than the online one, filling the rooms with student inquiries and satisfaction, surpassing the experience of online learning. Overall, the results suggest that in courses focused on software skills, online teaching shines in providing autonomy over tools theory, especially in cases where students' understanding varies in speed. Mechanisms for pausing and reviewing increase efficiency and depth of understanding. But experience showed there were differences: Some students produced the expected results, while others lagged behind due to distraction or procrastination, which indicates that online learning requires a lot more self-discipline. Traditional teaching, in contrast, did a better job maintaining both attention and progress. During assessment, real-time feedback and interactive elements, as evidenced in traditional settings, not only boosted engagement but also improved understanding, fostering critical competencies, such as problem-solving and adaptability—cornerstones of software mastery.

2. Conclusion

Traditional teaching traditionally provides a richer experience in the classroom, while online learning is more efficient learning behavior in a software skills course. With artificial intelligence and educational technology evolving, universities now focus on course quality and online teaching is now going to be priority in the upcoming future. While this cannot completely replace traditional methods, its usage in higher education is growing, especially in courses covering software skills. A hybrid style can be more suited towards the needs of individual students by combining the benefits of online and offline (indeed, with each of these formats making it possible for the other, blended learning). Online teaching will begin to leverage technologies such as virtual reality and AI to deliver a more direct and immersive, intuitive learning experience, while traditional teaching helps preserve the people aspects of learning face-to-face, sparking students' curiosity and sense of initiative. Rethinking software skills outcomes should focus on balancing these modes to design a quality hybrid delivery — enhancing professionalism and robustly addressing students' software skills and capabilities with industry expectations.

References

[1] Mubarak, Ahmed Ali, Ahmed, Salah A.M., Cao, Han. MOOC-ASV: analytical statistical visual model of learners' interaction in videos of MOOC courses. INTERACTIVE LEARNING ENVIRONMENTS. 2023; vol. 31 (5):3055-3070.

[2] Chao, Fan , Zong. Liyong. Human Dynamics Analysis on MOOC Online Learning Behaviors. Open Education Research. 2016; 22(2): 53-58.

[3] Zheng, Yuxia , M. A. Yanlei. Research on MOOC Curriculum Recommendation Model of Higher Vocational English Based on Improved Intensive Learning Network. Scalable Computing: Practice & Experience. 2024; 25(1):205-216.