Software Skills Acquisition: Confidence vs. Competence

Sheila M. Smith

College students bring varied levels of software skill competence to introductory information technology courses. While students claim a high degree of computer technology expertise, they may be indicating increased self-efficacy (confidence) rather than actual performance (competence). The purpose of this study was to examine the use of corresponding measures of pre- and post-course levels of both computer self-efficacy and performance for six software programs. There was not a significant difference between the pre-course self-efficacy and performance ratings. However, a significant difference existed between the post-course self-efficacy and performance ratings. Recommendations for practice and future research are discussed.

Introduction

Introductory information technology courses typically teach skill acquisition and fundamental software concepts. Basic courses designed to develop students' information technology capabilities are offered at the high school and elementary school level. College students are more likely than the general population to have access to technology at home. Greater availability has elevated self-efficacy (confidence) in ability to perform (competence) tasks with general-purpose business software such as, word processing, spreadsheet, database, presentation graphics, graphical user interface (GUI) management, and telecommunications.

Because of education and access, college students arrive in introductory information technology courses with varied levels of software skill confidence and competence. An individual's confidence in his capability to use a computer is referred to as computer self-efficacy and may contribute to ease of skill acquisition (Compeau & Higgins, 1995). However, students' beliefs about their ability to complete computer-related tasks may heighten or weaken performance. Previous experience may lead students to believe introductory software skill courses are easy. Others may have had difficulty with previous technology-oriented courses. Often students enter introductory information technology courses with self-efficacy perceptions that do not match their performance attainments. Heightened self-efficacy may cause students to expend little effort toward acquiring new software skills. Bandura (1982) stated, "in approaching learning tasks, those who perceive themselves to be supremely self-efficacious in the undertaking feel little need to invest much preparatory effort in it" (p. 123).

The assumption that students have greater software skill than in the past has lead postsecondary educators to question if introductory information technology courses should be taught at the postsecondary level. An underlying issue is whether students actually possess increased competence or merely increased confidence.

Purpose of the Study

This study sought to explore the relationship between computer self-efficacy and performance in an introductory information technology course. Computer self-efficacy may possibly have a strong correspondence with performance; however, research that examines actual performance indicators that are similar to perceptions of computer self-efficacy has not been conducted. Marakas, Yi, and Johnson (1998) highlighted the complexity of the computer self-efficacy construct and solicited additional research.

Sheila M. Smith is Associate Professor, Department of Information Systems and Operations Management, Ball State University, Muncie, Indiana.

The purpose of this study was to examine the difference between pre- and post-course levels of computer self-efficacy and performance for six software programs (word processing, spreadsheet, database, presentation graphics, graphical user interface (GUI) management, and telecommunications). Specifically, the study was designed to address the following research questions:

1. Does a significant difference exist between pre-course self-efficacy and performance?
2. Does a significant difference exist between post-course self-efficacy and performance?
3. Does a significant difference exist between pre- and post-course self-efficacy?
4. Does a significant difference exist between pre- and post-course performance?

Related Literature

Several empirical studies have reported a significant difference between the computer self-efficacy and performance relationship. However, measures that maintain a direct correspondence between computer self-efficacy and actual performance have not been used. Therefore, the literature review details research studies that are tangentially related to the present study and have been used to form the foundation for the current study.

Hasan (2003) examined the influence of eight types of computer experiences on computer self-efficacy. Experience with word processing, spreadsheets, databases, operating systems, graphics, computer games, telecommunications, and programming languages was used to predict computer self-efficacy beliefs among 151 part-time and non-traditional students enrolled in a computer information systems course. Although all eight types of computer experiences had a positive and significant correlation with computer self-efficacy, computer graphics and programming had the strongest effect on perceptions of computer self-efficacy.

Marcolin, Compeau, Munro, and Huff (2000) developed an experimental model that compared cognitive and affective (self-efficacy) assessments. Sixty-six subjects completed measures for two different software applications (word processing and spreadsheet). For each of the two software applications, each subject completed the User Competence Depth questionnaire measure of software knowledge (Munro, Huff, Marcolin, & Compeau, 1997), a questionnaire measure of self-efficacy (Compeau & Higgins, 1995, 1999), a commercial multiple-choice test of software knowledge, and a self-reported self-efficacy questionnaire. Results from a repeated measures analysis of variance suggested a statistically significant difference between self-efficacy and knowledge; subjects typically possessed a higher level of confidence in comparison to their knowledge. Results also indicated that self-efficacy for word processing was consistently higher than for the spreadsheet application. Marcolin et al. (2000) also found that confidence about general ability was somewhat higher than confidence regarding specific abilities.

Based on pre- and post-course measures, Karsten and Roth (1998) examined the relationship between computer experience, self-efficacy, and performance in an introductory computer literacy course. Study results found that although a variety of computer experiences enhanced student perceptions of their computer competencies, only those experiences that develop specific computer skills were likely to have an impact on computer-dependent course performance. Results suggested that the applicability of computer assignments rather than quantity contributed to the quality of performance. The study found a moderate relationship between computer self-efficacy and course performance. In addition, a significant difference was found in student’s pre- and post-course computer self-efficacy; higher levels of computer self-efficacy were reported at the end of the semester. The study concluded that even among experienced students, introductory computer literacy courses play an important, confidence-building role in the development and confirmation of skills necessary for proficient computer use.

While these studies explored and found a casual relationship between computer self-efficacy and performance, the use of subjective, self-reported task performance measures may contribute to a narrow conceptualization of the confidence and competence relationship. The current study sought to employ an objective assessment of task performance in an attempt to offer a more accurate and reliable predictor of information technology behavior.

Method

Details of the research instruments used to conduct the study are discussed in this section. This section also addressees the pilot study, selection of participants, research procedures, and data analysis.

Instrumentation

Due to the lack of commercially available research instruments, I constructed two assessments in accordance with Bandura's (1986) criterion regarding consistency of self-efficacy and performance assessment measures. Corresponding confidence and competence items were constructed to ensure measurement equivalency. Each item was constructed so that actual ability to perform a task would be a true measure of competence, not rote memorization or ability to follow directions.

Computer Self-Efficacy Assessment (CSEA). Measures of self-efficacy that are specifically tailored to the task assessed provide greater predictability of performance outcome. In addition, judgments of self-efficacy are task and domain specific. Bandura (1977) stated self-efficacy measures in academic areas are operational of the belief that one can successfully accomplish the behavior to produce the desired outcome. Therefore, I designed the CSEA to measure confidence in ability to perform specific computer task activities. Students rated their level of self-efficacy on a 5-point Likert scale from "completely confident" (5) to "not at all confident" (1). The 23-item assessment was composed of six performance units: word processing, spreadsheet, database, presentation graphics, graphical user interface (GUI) management, and telecommunications. For example, an item in the word processing unit asked students about their level of confidence in their ability to "use an application software template." High internal consistency for CSEA was indicated by the Cronbach's alpha coefficients of .94 for the Pre-Course Confidence and .91 for the Post-Course Confidence.

Computer Competence Assessment (CCA). Consistent with the CSEA, I designed the CCA to measure computer-related task competence. The 16-item assessment asked students to produce documents or complete computer-related tasks in accordance with the 23 items based on the six units listed on the CSEA. For example, in the word processing unit, students were provided a hard copy of a memo prepared by the administrator. Without written instructions, students were given the opportunity to reproduce the memo using a word processing template. High internal consistency for CCA was indicated by the Cronbach's alpha coefficients of .93 for the Pre-Course Competence and .89 for the Post-Course Competence.

Pilot Study

Pilot study participants were 10 undergraduate students (6 males and 4 females) at a midwest university. Seven of the students were sophomores, two were freshmen, and one was a junior. Students were primarily business majors 60% (n = 6), followed by sports management majors 30% (n = 3), and communications majors 10% (n = 1). Students were enrolled in an introductory business information technology course. Administration of the Computer Self-Efficacy Assessment (CSEA) and the Computer Competence Assessment (CCA) took place during the last two 90-minute class periods of a 5-week summer session. Descriptive statistics were calculated; data included mean scores and the frequency of students' computer self-efficacy and computer task competence rating differences. Students provided comments about the difficulty level and comprehensiveness of the CSEA and the CCA. Based on preliminary data and student feedback, only minor modifications were made to the assessments.

Participants

Students for the study consisted of 66 undergraduates (28 women and 38 men) enrolled in an introductory business information technology course. The course was 60 percent lecture and 40 percent computer lab based. Ages ranged from 18 to 27 years (M = 19.5 years). The sample's racial composition was: White, 87.9% (n = 58); African American, 4.6% (n = 3); Hispanic, 3% (n = 2); Asian, 3% (n = 2); and Other, 1.5% (n = 1). Students were primarily sophomores, 48.5% (n = 32); followed by juniors, 22.7% (n = 15); freshmen, 21.2% (n = 14); and seniors, 7.6% (n = 5). The majority of students (n = 42) were pursuing a degree in business. The remaining students were from applied sciences and technology (n = 7), architecture (n = 2), fine arts (n = 3), sciences and humanities (n = 3), interdepartmental (n = 2), and undecided (n = 7). Forty-seven (71%) of the students reported that they owned a computer and 44 (66.7%) indicated that they had been using a computer for more than 5 years.

Research procedures

All assessments were administrated during the first week of a 16-week semester. Students completed the Computer Self-Efficacy Assessment (CSEA) within one 50-minute class period. The next class meeting following the CSEA, students were administered the Computer Competence Assessment (CCA). Completion of the CCA required two 50-minute class periods. Bandura (1986) stated that self-efficacy and performance should be assessed within a close period and that self-efficacy assessment precedes performance assessment.

In addition to the submission of a hard copy of certain items on the CCA, all students were provided with a floppy disk and instructed to save all completed assignments on the disk. Several CCA items required the students to list the steps in a computer-related task, such as formatting a disk directly onto the CCA assessment. Students did not receive written instructions in reference to completion of items on the CCA, were instructed not to use the help menu, and were not provided assistance from the administrator. The administrator and graduate assistant were present during all the assessment sessions.

Data Analysis

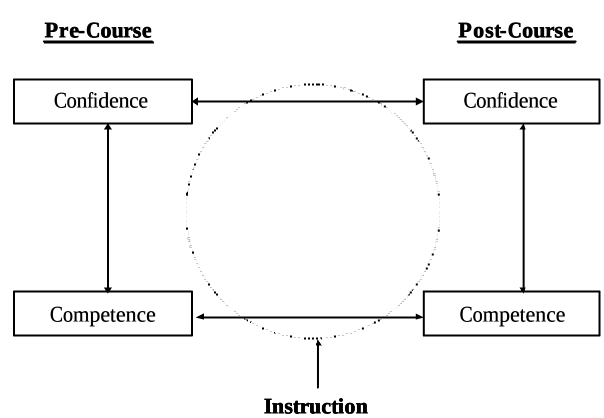
A scoring matrix was developed to analyze the results of the Computer Competence Assessment (CCA). The CCA rating was designed to complement the confidence rating on the Computer Self-Efficacy Assessment (CSEA). CCA ratings ranged from "successful task completion" (5) to "no evidence of task completion" (1). The graduate assistant and I rated all CCA items, such as hard copies of documents, steps listed directly onto the CCA, and each student's floppy disk. Students' personal ratings on the CSEA were compared to the investigator's CCA ratings.

Pearson product-moment correlation (*r*) was used to examine the relationship among key variables of interest. A multivariate analysis using the Wilks Lambda criterion and independent t-tests were used to determine differences between confidence and competence before and after course instruction.

Results

The structure of the data collected and the analysis that was performed is depicted in Figure 1. Correlations and t-tests that examined differences in self-efficacy (confidence) and performance (competence) were conducted for both the pre- and post-course research sessions. The Pearson

Figure 1: A Pictorial Representation of the Structure of the Analysis



product-moment correlation coefficient revealed a significant correlation between the pre- and post-course confidence variables ($r = .55$), and the pre- and post-course competence variables ($r = -.29$) at the .01 level.

Descriptive statistics of the four measurement scales are presented in Table 1. Confidence ratings were higher than competence ratings for both the pre-course ($M = 3.48$) and post-course ($M = 4.31$) research sessions. The coefficient alphas demonstrated a high level of internal consistency.

Table 1: Descriptive Statistics for Pre- and Post-Course CSEA and CCA

Research Session	Assessment	n	M	SD	Alpha
Pre-course	Confidence	66	3.48	.718	.94
	Competence	66	3.39	.889	.93
Post-course	Confidence	66	4.31	.495	.91
	Competence	66	4.01	.793	.89

CSEA = Computer Self-Efficacy Assessment
CCA = Computer Competence Assessment

An analysis of variance was performed on six dependent variables—word processing, spreadsheet, database, presentation graphics, GUI management, and telecommunications—to determine if there were any differences in pre- and post-course self-efficacy and performance.

A multivariate test with a Wilks' criterion revealed a significant difference between both pre- and post-course measures, Wilks $\lambda = .88$, $F(2, 63) = 4.34$, $p < .05$. The first research question sought to determine if there was a difference between pre-course self-efficacy and performance. Significant tests did not reveal a significant difference in the pre-course self-efficacy and performance research sessions, $t(65) = .76$, ($F = .582$, $p > .05$). The second research question that compared the difference between post-course self-efficacy and performance revealed a statistically significant difference, $t(65) = 2.57$, ($F = 6.20$, $p < .05$). See Table 2.

With the use of Wilks' criterion, the dependent variables significant difference was confirmed: Wilks $\lambda = .31$, $F(1, 64) = 70.25$, $p < .05$. The third research question that focused on the difference between pre- and post-course self-efficacy revealed significant differences at the .05 level for all six dependent variables. The combined dependent variables were significantly different, $t(65) = 10.98$, ($F = 120.46$, $p < .05$). Analysis of the fourth research question that asked if there was a pre- and post-course difference in performance yielded a significant difference for five of the six dependent variables. Although, there was not a significant difference in pre- and post-course performance for the telecommunications unit, the combined dependent variables were significant, $t(65) = 3.72$, ($F = 13.52$, $p < .05$). See Table 3.

Discussion

A major contribution of the present study was the empirical test of the model shown in Figure 1. The results indicated that although students were confident in their ability to perform basic software application tasks, their actual performance level was consistently below their confidence level.

The findings of this investigation indicated that before classroom instruction there was not a

Table 2: Tests of Significance for Confidence and Competence, Total Scales (N = 66)

Measure	M	SD	df	t
Pre-Course				
Confidence	3.48	.71		
Competence	3.39	.90	65	-.76
Post-Course				
Confidence	4.31	.49		
Competence	4.01	.79	65	2.57*

Wilks $\lambda = .88$ df1 = 2 df2 = 63 .017 *$p < .05$.

Table 3: Tests of Significance for Pre and Post-Course Scales (N = 66)

Measure	M	SD	df	t
Confidence				
Pre-Course	3.48	.71		
Post-Course	4.31	.49	65	10.98*
Competence				
Pre-Course	3.39	.89		
Post-Course	4.01	.79	65	3.72*

Wilks $\lambda = .31$ df1 = 2 df2 = 63 .000 *$p < .05$.

significant difference in perceptions of self-efficacy and ability to perform. However, at the end of the course, there was a significant difference in self-efficacy and performance ratings for the word processing, spreadsheet, and database units. Students displayed higher levels of confidence than actual competence for word processing, spreadsheet, and database software at the end of the semester.

The findings demonstrated a significant difference in pre- and post-course self-efficacy levels for all six software units. Students' confidence in their ability to perform computer-related tasks based on the six software units increased over the course of the semester. This finding is in accordance with prior studies that found an increase in computer self-efficacy at the end of instruction (Karsten & Roth, 1998). With the exception of the telecommunications unit, there was a difference in the pre- and post-course performance ratings. Comparison of the pre- and post-course performance ratings indicated that students acquired more competence after the classroom experience. The study findings indicated that classroom instruction on basic system and application software programs contributed to an increase in student performance on concrete activities relevant to the students' academic success. This constitutes evidence that introductory information technology courses with an emphasis on basic system and application software programs should continue to be taught.

Although self-efficacy judgments are generally associated with performance, a number of conditions can affect the strength of the relationship. Discrepancies may arise because of misjudgments of self-knowledge or task requirements (Bandura, 1982, 1997). The most common form of judgmental disparity is one in which beliefs of personal efficacy exceed performance. Optimistic judgments of self-efficacy do not necessarily mean that individuals have inflated views of their capabilities. "Such disparities may stem from exaggeration of one's capabilities or from inadequate knowledge of task demands" (Bandura, 1997, p. 65). This is particularly true with introductory information technology courses that emphasize the use of basic system and application software programs. Many students arrive in introductory technology courses with great confidence in their ability to perform a computer-related task but are often unable to accomplish the task without extensive facilitated classroom instruction. Students often use basic system and application software to perform simple tasks and are unaware that the software has advanced capabilities. Peripheral uses of the computer, such as Internet activities and music media, may also contribute to higher levels of confidence. The ability to use information technology to perform leisure activities does not equate to one's ability to perform workplace relevant activities with the use of word processing, spreadsheet, database, presentation graphics, GUI management, and telecommunications software.

Self-efficacy and performance discrepancies may also arise when teaching strategies do not complement desired learning outcomes. Incongruities in the confidence and performance paradigm form the basis for instructional interventions (Akhras & Self, 2000). Beginning skill development courses may benefit from a systematic, hands-on approach to learning basic software skills. Using the facilitated (step-by-step) teaching strategy enables students to become comfortable, but not completely proficient with, basic system and application software. This strategy lacks an emphasis on problem-solving and other cognitive processes (Davis, 1993). Advanced software skill development courses, and especially courses taught at the postsecondary level, need to employ the constructivist perspective that engages the learner in critical thinking. This strategy empowers the learner by allowing the individual to become the producer, rather than the consumer, of information (Chen, 2003, Forcier, 1996; Gagné, Briggs, & Wager, 1992). The constructivist approach could help eliminate the confidence and competence disparity found at the end of course instruction in the present study.

Recommendations

Based on the results of this study, the following recommendations are provided in an effort to help educators apply research findings to their classrooms. In addition, this section also outlines recommendations for future research efforts.

Recommendations for the Classroom

1. Evaluation of the computer self-efficacy and performance relationship may provide students with realistic performance indices about their general abilities. Awareness of the influential role of self-efficacy beliefs on academic attainments may provide students with an overall authentic view of their current beliefs and abilities. The individual evaluation may be used to guide behavior and help students overcome possible challenges to acquiring software skills.
2. Identification of computer self-efficacy and performance differences may help students focus on specific computer performance units with the greatest inconsistencies. Large discrepancies may be remedied with specialized computer tasks (e.g., word processing, spreadsheet) designed to eliminate targeted self-efficacy and performance differences. Awareness of computer self-efficacy related to computer task performance will enhance classroom learning and instruction. Computer self-efficacy in conjunction with performance assessment may help identify academic strengths and weaknesses while supplying insight into the students' self-perceived barriers to learning or acquiring new skills related to general-purpose business software programs.
3. Classroom assessment of computer self-efficacy and performance may guide the education process by providing educators with directions on how to facilitate learning. Educators may find the assessment useful in predicting outcomes. The prediction of possible outcomes can lead to the development of interventions specifically tailored for student deficiencies and the implementation of alternative resources that contribute to different learning styles.
4. Educators should continue to teach introductory information technology courses at the postsecondary level. However, incorporation of the constructivist teaching-learning strategy is imperative for successful transfer of learning. Difficulty platforms can be established to accommodate varied experiences and education. The implementation of a test-out procedure or placement exam would ensure that students are placed in courses that coincide with their level of competence.

Recommendations for Future Research

1. A study is needed that explores the impact of goal setting after the initial self-efficacy and performance assessment. Goals may influence the end-of-semester computer self-efficacy rating and performance level. The establishment of goals throughout the software skill acquisition period may increase student's self-efficacy, motivate the use of self-directed learning techniques, and increase their focus on the learning process (Schunk & Ertmer, 1999).
2. A study is needed that extends the Marcolin et al. (2000) experiment. The study should include matching measurements of computer self-efficacy, software knowledge, and computer-related tasks performance. Administration of the measurements can be conducted before and after course instruction for an in-depth examination of the knowledge (cognition), self-efficacy (confidence), and performance (competence) relationship.
3. A study is needed that examines the practical uses for the constructivist teaching strategy within the information technology curriculum. The constructivist view of education is based on providing the student with a sensory stimulated environment that inspires learning. An information technology course enhanced by multimedia provides the ideal environment for the stimulation of sensory experiences. (Chen, 2003; Forcier, 1996; Gagné, Briggs, & Wager, 1992).

References

Akhras, F. N., & Self, J. A. (2000). Modeling the process, not the product, of learning. In S. P. Lajoie (Ed.), *Computers as cognitive tools: No more walls* (pp. 3-28). Mahwah, NJ: Lawrence Erlbaum Associates.

Bandura, A. (1977). Self-efficacy: Toward a unifying theory of behavioral change. *Psychological Review, 84*(2), 191-215.

Bandura, A. (1982). Self-efficacy mechanism in human agency. *American Psychologist*, *37*(2) 122-147.

Bandura, A. (1986). *Social foundation of thought and action: A social cognitive theory*. Englewood Cliffs, NJ: Prentice-Hall.

Bandura, A. (1997). *Self-efficacy: The exercise of control*. New York: W. H. Freeman and Company.

Chen, C. (2003). A constructivist approach to teaching implications in teaching computer networking. *Information Technology, Learning, and Performance Journal*, *21*(2), 17-27.

Compeau, D. R., & Higgins, C. A. (1995). Computer self-efficacy: Development of a measure and initial test. *MIS Quarterly*, *19*(2), 189-211.

Compeau, D. R., & Higgins, C. A. (1999). Social cognitive theory and individual reactions to computing technology: A longitudinal study. *MIS Quarterly*, *23*(2), 145-159.

Davis, J. R. (1993*). Better teaching, more learning: Strategies for success in postsecondary settings*. Phoenix, AZ: The Oryx Press.

Forcier, R. C. (1996). *The computer as a productivity tool in education.* Englewood Cliffs, NJ: Prentice-Hall, Inc.

Gagne, R., Briggs, L., & Wager, W. (1992). *Principles of instructional design.* Fort Worth, TX: Harcourt Brace Jovanovich.

Hasan, B. (2003). The influence of specific computer experiences on computer self-efficacy beliefs. *Computer in Human Behavior, 19*(4), 443-450.

Karsten, R., & Roth, R. M. (1998). The relationship of computer experience and computer self-efficacy to performance in introductory computer literacy courses. *Journal of Research on Computing in Education, 31*(3), 14-24.

Marakas, G. M., Yi, M. Y., & Johnson, R. D. (1998). The multilevel and multifaceted character of computer self-efficacy: Toward clarification of the construct and an integrative framework for research. *Information System Research, 9*(2), 126-163.

Marcolin, B. L., Compeau, D. R., Munro, M. C., & Huff, S. L. (2000). Assessing user competence: Conceptualization and measurement. *Information Systems Research, 11*(1), 37-60.

Munro, M. C., Huff, S. L., Marcolin, B. L., & Compeau, D. R. (1997). Understanding and measuring user competence. *Information & Management, 33*(1), 46-57.

Schunk, D. H., & Ertmer, P. A. (1999). Self-regulatory processes during computer skill acquisition: Goal and self-evaluative influences. *Journal of Educational Psychology, 91*(2), 251-260.