Association for Information Systems
AIS Electronic Library (AISeL)

SIGHCI 2003 Proceedings

Special Interest Group on Human-Computer
Interaction

2003

Usability and Efficacy Reactions to Object-Orientation: The Impact of Prior Knowledge

Liping Liu
University of Akron, liping@uakron.edu

Follow this and additional works at: http://aisel.aisnet.org/sighci2003

Usability and Efficacy Reactions to Object-Orientation: The Impact of Prior Knowledge

Liping Liu
University of Akron
liping@uakron.edu

ABSTRACT

In this paper, we examine how prior knowledge impacts usability and efficacy reactions to object-oriented techniques. We develop research hypotheses based on the multiconstraint theory of analogical reasoning. We empirically test the hypotheses in an open learning setting. We observed a significant interaction effect: the subjects with prior knowledge on either data or process modeling technique perceived greater difficulty and less confidence in learning object-oriented techniques than novices as well as those who have prior knowledge on both structured techniques. Prior knowledge explained 19% of the variance in both usability and efficacy reactions and, as a common cause, partially explained their correlation.

Keywords

Usability, self-efficacy, object-oriented techniques.

INTRODUCTION

Most current systems analysts were trained in structured techniques. Most information technology (IT) curricula are still teaching structured techniques as the primary topic for systems analysis and design. However, with the recent standardization of the unified modeling language (UML), the trend in software development is moving toward more object-orientation, which is believed to be in many ways different from its structured counterparts and requires a different mindset in modeling business problems. To respond to the radical change, many organizations are faced with the task of retraining their existing analysts as well as new hires.

There exist studies examining how prior knowledge on process models impacts the learning of object-oriented (OO) techniques using objective measures, such as task performance (Agarwal et al., 1996b, Boehm-Davis and Ross, 1992, Morris et al., 1999), cognitive effort (Morris et al., 1999), and cognitive differences (Vessey and Conger, 1994, Lee and Pennington, 1994). In this study we draw attention to a different inquiry — how systems analysts with prior knowledge react to object-orientation. In particular, we consider two trainee reactions — the usability of OO techniques and the self-efficacy of learning OO methodology — as the effectiveness criteria for retraining systems analysts.

Among many trainee reactions, usability (difficulty) is the only one that predicts actual learning such as post-training knowledge and task performance (Warr and Bunce, 1995). Self-efficacy is an antecedent to and consequent of other training outcomes (Gist et al., 1989, Gist et al., 1991), and measures the substantive value of training (Agarwal et al., 1996a). In addition, how systems analysts make behavioral choices is more based on their subjective beliefs rather than objective counterparts. Their after-retraining work attitude and job satisfaction also depends on these subjective beliefs. Therefore, it is important that we examine these trainee reactions to better manage the technology transition for organizations and design effective retraining programs for software designers.

HYPOTHESES DEVELOPMENT

Prior Knowledge: Knowledge is internalized information related to concepts, procedures, and judgments, and a justified personal belief that increases one's capacity to take action. It can be internally represented as IF-THEN rules, mental models, or propositions, exist in one of three progressive forms: declarative knowledge, procedural knowledge, and schemas and scripts, and be classified into six hierarchical levels: recall, comprehension, application, analysis, synthesis, and evaluation.

Usability: Usability refers to the degree to which one believes that using a system is free of effort (Davis, 1989). It captures the cognitive and emotional effort required to master training materials (Warr and Bunce, 1995). There exists extensive research on how to design usable systems for non-technical end users (Adler and Winograd, 1992). However, our knowledge on the usability of development tools is sparse and the results are inconclusive.

Self-Efficacy: Self-efficacy (SE) refers to the degree to which one is confident in performing a specific task (Gist and Mitchell, 1992). It relates to motivational and behavioral concepts such as proactive attitudes, adaptability to new technology, and learning and achievement. In the IT context, computer SE refers to the judgment by an individual of his or her capability to use an information technology (Compeau and Higgins, 1995). Marakas et al. (1998) made a further distinction between computer SE and task-specific SE. Following the same distinction, in this study we define SE as an individual's estimate of his or her capability to perform OO modeling tasks.

Analogical Reasoning

Learning theories all recognize the role of analogical reasoning in learning. Proposition-based theories posit that learning is a process of making proposition-based inferences; incoming information are compared against stored knowledge, represented as propositions, for assessing their similarities, which can then be used to create a new instance in memory or refine existing knowledge. Production-based theories, which assume knowledge is represented as IF-THEN rules, posit that the learner first draws heavily on analogies and examples to understand how the declarative knowledge is applied to problem solving. Then the procedure knowledge is compiled into schemas, scripts, or other abstract knowledge structures so that exercising the knowledge becomes automatic. Schemas and scripts are activated unconsciously based on similarities when interpreting new concepts or events and analogical reasoning is further enhanced (Gick and Holyoak, 1983).

Thus, while learning OO techniques, individuals with prior knowledge tend to draw analogies back to more familiar structured techniques and maps new concepts onto something familiar. When they have surface knowledge on structured techniques, they represent the knowledge as a set of rules (procedural knowledge) and facts (declarative knowledge), and map OO concepts and skills into individual elements in structured techniques. Analogical mappings tend to be made at superficial levels (Gentner, 1988) such as model elements and relationships. However, after possessing deep knowledge, individuals represent it as a more abstract structure, use the structure to evaluate incoming information for relevance, and place OO concepts and skills into the overall structure according to the similarities. Analogical mappings tend to be made at higher levels such as modeling objectives and cognitive modeling tasks.

There are some empirical observations on the use of analogies in learning OO techniques. Nelson et al. (2002) found that procedural developers tend to map "object" to the familiar concepts of "module," "function," or "database record" and map "class" to the concepts of "database table" or "structure" and so on. Detienne (1995) found that procedural programmers structure OO programs by functional similarity and execution order rather than by class memberships. Pennington et al. (1995) found that procedural analysts decompose a problem driven by actions on the data rather than by domain entities. Based on these findings, we developed an extensive list of analogical mappings, including mappings of elements, relationships, overall models, modeling objectives, and cognitive activities. These mappings cover all the essential OO concepts and skills for the rational unified process, the *de facto* industry standard of OO development process.

Holyoak and Thagard (1989, 1995) found that the use of analogy is guided by a number of general constraints that jointly encourage coherence in analogical thinking. They proposed three broad classes of constraints that form the basis of the so-called *multiconstraint theory*. First, the analogy is guided to some extent by direct similarity of the elements involved. Second, the analogy is guided by a pressure to identify consistent structural parallels between roles in the source and target. These first two constraints form a pressure to establish an isomorphism — a set of consistent, one-to-one correspondences — between the elements of the source and target. Third, analogical thinking is guided by what the analogy is intended to achieve. Holyoak and Thagard (1997) further suggested that the multiple constraints — similarity, structure, and purpose — do not operate like rigid rules dictating the interpretation of analogies. Instead they function more like the various pressures that guide an architect engaged in creative design, with some forces converging, others in opposition, and their constant interplay pressing toward some satisfying compromise that is internally coherent.

The multiconstraint theory implies that the ease of analogical reasoning depends on how much the three constraints can be satisfied and how much compromise one has to make. The easier it is to identify the isomorphism of elements and their relationships between the source and target, the easier one feels about performing the analogical reasoning. The easier it is to achieve the reasoning goals, the more favorable one feels about the ease of learning of the target. Therefore, how prior knowledge affects the usability of OO techniques depends on the extent to which OO concepts and skills can be mapped to structured counterparts.

According to empirical analogical mappings, individuals with prior knowledge on both data- and process-oriented techniques can map all essential OO concepts and skills to their familiar ones. In contrast, individuals with knowledge on either data- or process-modeling techniques alone can only map a portion of them while finding many others to be difficult. Thus, when there is knowledge in both data- and process-modeling techniques, there is greater ease of making analogical reasoning, leading to a more favorable perception on the usability of OO techniques:

H1: Individuals with prior knowledge on both data- and process-modeling techniques perceive the usability of OO techniques more favorably than those with knowledge on either data- or process-modeling techniques alone.

Novices have no analogies to make. They approach the learning task by using general problem-solving strategies such as "divide and conquer." They anchor usability to their general beliefs. In contrast, those with prior knowledge will make an adjustment to reflect their experience of analogical reasoning although their judgment still anchors to the general beliefs (Venkatesh, 2000). In particular, for those with prior knowledge on data- or process-modeling techniques, when they find it difficult to coherently map all OO concepts and skills onto those they are familiar with, their perception will be negatively adjusted:

H2: Compared to novices, individuals with prior knowledge on either data- or process-modeling techniques alone perceive the usability of OO techniques less favorably.

The Determinants of Self-Efficacy

Although experience influences efficacy perceptions, it is the cognitive appraisal that ultimately determines SE. Gist and Mitchell (1992) proposed that three types of information cues are involved in forming SE: task requirements analysis, attributional analysis, and resource analysis. Task requirements analysis produces inferences about what it takes to perform at various levels. The attributional analysis involves judgments about why particular performance occurred in the past. The resource analysis examines the availability of specific resources and constraints for performing the task at various levels.

Among the three SE information cues, different cues may be used in assessing SE estimates depending on the assessor's experience and task characteristics. When the task is fairly novel or when it has been observed only, one may invoke in-depth and detailed analysis of task requirements as well as resource constraints as the primary information cue for SE judgments. When the task has been performed personally and frequently in the past, the individual is likely to rely more heavily on his or her interpretation of the causes of pervious performance levels and to use interpretations as the primary determinant of SE. In general, judgments about efficacy become more automatic as experience with a task increases.

Learning OO techniques is a novel task to all trainees. Their experience is at best an observer's. Therefore, trainees will most likely use in-depth analysis of task requirements and resource constraints as the primary information cue for their SE judgments. At the same time, analyzing the skill and effort requirements for performing OO analysis bears a striking similarity to perceiving how easy it is to learn OO techniques. Thus, we have the following three anticipations:

H3: Individuals with prior knowledge on both data- and process-modeling techniques have greater self-efficacy in performing OO analysis than those with knowledge on either data- or process-modeling techniques.

H4: Novices have greater self-efficacy in performing OO analysis than the individuals with prior knowledge on either data- or process-modeling techniques.

H5: Self-efficacy is positively correlated with usability; the more favorably one perceives the usability of OO techniques, the more confident he or she feels about performing OO analysis.

RESEARCH DESIGN

We conceptualize prior knowledge using two variables. We use KDM to represent prior knowledge on data models and KPM to represent prior knowledge on process models. We control each variable at two levels: 0 (absence) and 1 (presence) and follow the 2×2 factorial design involving four groups of subjects, where Group A consists of subjects with knowledge on both data and process models; Group B on data models; Group C on process models; and Group D

consists of novices who have no prior exposure to either models.

To implement the design, we recruited potential subjects from senior classes at a large Midwest American university. We requested the rosters of all current and previous classes and screened each candidate with respect to his or her prior knowledge on data and process models. After the screening, we selected 131 trainees to participate in this study. We controlled prior knowledge through relevant courses and provided additional pre-training if necessary. For example, the instructors gave five weeks of extensive lectures and exercises on data modeling techniques to Group B and the same amount of preparation on process modeling techniques to Group C. In addition to regular lectures, these subjects were assigned to solve 20 design problems, one exam, and one large, real business project to fulfill their course requirements. The pre-training treatment was meant to provide equivalent coverage of the same topic in industry training and to prepare the subjects for entry-level systems analyst positions.

We conducted the study using an open learning setting, where trainees worked on their own to learn written materials (Warr and Bunce, 1995). After finishing prior knowledge control, we provided each subject with a training material on OO modeling. The material covers UML, OO concepts such as inheritance, encapsulation, and polymorphism, and how to develop use case and class diagrams to model business problems.

After the two-week open learning period, we conducted training evaluation in an examination setting. As a part of examination, we administered a short quiz consisting of 5 screening questions to ensure that the trainees actually read the training materials. A trainee was dropped from the study if he or she did not score at least 4 points. Eventually, we ended up with 72 subjects and 18 in each controlled group. Among them, 41 were males and 31 females. 52% of them majored in Information Systems and 48% in other business areas. All subjects had about the same level of maturity and computer experience.

Training evaluation consists of two parts. First, we gave the trainees a real systems analysis task and asked them to create an OO analysis model as the blueprint for the system to be developed. Then, each subject was asked to respond to a survey regarding his or her efficacy and usability reactions.

Self-Efficacy: To develop a measure for SE, we followed the five-point framework proposed by Marakas et al. (1999); we focused on the subject's perceived ability to perform a specific task while avoiding the ability assessments on cross-domain or general-domain skills. In object modeling, a subject needs to identify objects, attributes, and methods based on data and functional requirements, and discern object relationships based on data navigation and behavior collaboration. Accordingly, we developed seven questions that assess one's estimated ability to perform each specific task.

Usability: We selected three items from Davis (1989) with no modifications: Easy to Learn, Easy to Become Skillful, and Easy to Use. Then we considered the differences between using a system, and learning OO techniques. The most significant difference is that the latter requires a lot more effort in understanding and comprehending concepts and applying them creatively while the former demands more effort in interacting with the system. Therefore, we dropped the two items related to interaction: Controllable and Flexible and modified and expanded the item "Clear and Understandable" into two items that ask whether OO concepts are straightforward and whether it is easy to comprehend them. To capture the cognitive effort aspect of usability (Goldstein and Gilliam, 1990), we added two items that assess how comfortable a subject feels. Finally, we ended up with eight items in the 7-point Likert scale for usability.

RESEARCH RESULTS

To assess the efficacy of scale items, we conducted reliability analyses. The correlations between SE items range from 0.47 to 0.85 and between usability items from 0.45 to 0.78. The Cronbach alphas are respectively 0.93 for SE and 0.92 for usability. The indices are very high compared to the acceptable level 0.7, demonstrating the convergent validity of the items. To ensure that the items for the same construct measure a single trait whereas items for different constructs measure distinct traits, we conducted a principal factor analysis with Varimax rotation. Using the Kaiser eigenvalues criterion, we extracted two factors that collectively explained 69.6% of the variance in all items. The rotated factor matrix shows that all the items cleanly loaded onto the correct latent constructs.

Tables 1 and 2 show the results of testing H1 and H2. As they show, the mean usability of Group D (3.889) is higher than that of Group B (3.017) and the difference is significant at the level $\alpha = 0.01$. Similarly, the mean usability of Group D (3.889) is higher than that of Group C (2.989) and the difference is significant at the level $\alpha = 0.01$. Therefore, Hypothesis H2 is strongly supported by the data. By comparing Group D with Groups B and C combined, we found H2 is even more significantly supported at the level $\alpha = 0.005$. The support for Hypothesis H1 can be similarly analyzed. The mean usability of Group A (4.044) is significantly higher than that of Group B (3.017) at $\alpha = .005$, than that of Group C (2.989) at $\alpha = .005$, and than that of Groups B and C combined (3.003) at $\alpha = .001$. Thus, H1 is strongly supported by the data.

Group	Size	Mean	Std. Dev.	Error
A	18	4.044	.957	.226
B	18	3.017	1.050	.247
C	18	2.989	1.087	.256
B & C	36	3.003	1.053	.176
D	18	3.889	1.049	.247

Table 1: Group Mean Usability

Comparisons	T-Value	DF	Sig.
A vs. B	3.070	34	.002***
A vs. C	3.093	34	.002***
A vs. B & C	3.529	52	.001***
D vs. B	2.493	34	.009***
D vs. C	2.528	34	.008***
D vs. B & C	2.918	52	.003***

Table 2: T-Tests of Usability

Tables 3 and 4 summarized the results of testing Hypotheses H3 and H4. They show that the mean SE of Group A is significantly higher than that of both Groups B and C at the level $\alpha = 0.001$. Thus, H3 is strongly supported. The support for H4 is relatively weaker. The mean SE of Group D is higher than that of both Groups B and C. The difference between Group D and Group C is significant at $\alpha = 0.1$ and between Group D and Groups B and C combined is significant at $\alpha = 0.05$. However, the difference between Groups D and B is not significant at the level $\alpha = 0.1$.

Group	Size	Mean	Std. Dev.	Error
A	18	465.56	85.21	20.09
B	18	335.56	105.34	24.83
C	18	307.78	137.69	32.45
B & C	36	321.67	121.64	20.27
D	18	393.89	172.50	40.66

Table 3: Group Mean Efficacy Indices

Comparisons	T-Value	DF	Sig.
A vs. B	4.071	34	.000***
A vs. C	4.134	34	.000***
A vs. B & C	4.488	52	.000***
D vs. B	1.224	34	.115
D vs. C	1.655	34	.054*
D vs. B & C	1.783	52	.040**

Table 4: T-Test of Self-Efficacy

Although not explicated, H1-H4 jointly predicts an interaction effect of prior knowledge. To validate it, we conducted two ANOVA tests using KDM and KPM as two fixed factors and usability (SE) as the dependent variable. The test results show a strongly significant interaction effect, which is significant at $\alpha = 0.001$ and is able to predict 18.8% of the variance in both usability and SE.

To test H5, we conducted a regression analysis using SE to predict usability. The result shows a Pearson correlation 0.52 with t-value = 5.074, which is significant at $\alpha = 0.001$ in a 2-tailed t-test. Thus, H5 is strongly supported. The regression model is significant at $\alpha = 0.001$ with a F-value = 25.75. If it is correct that SE determines usability (Compeau and Higgins, 1995, Venkatesh, 2000), the result here suggests that SE can predict 26.9% of the variance in usability.

CONCLUSION

Before discussing contributions, we shall note that the use of student trainees may affect external validity. The same concern also affects other similar studies. However, since colleges are still teaching structured techniques, the subjects in this study are representative of the population of at least new graduates, whom organizations often have to re-train to do object-oriented analysis and design. After all, the goal of the research design was to maximize the internal validity and provide a precise control of prior knowledge, which would be difficult in field studies.

This study improves our understanding on the transition from structured to OO techniques, and sheds light on the debate about revolutionary vs. evolutionary theories (Sircar et al., 2001). The existing studies have mixed findings based on objective measures. In contrast, this study suggests that not only the presence of prior knowledge but also the types of the knowledge have different effects. For example, we found that individuals knowing process models perceived greater difficulty and less confidence in learning OO techniques than novices. However, with addition of knowledge on data models, the effect reverses; individuals having knowledge on both data and process models perceive less difficulty and more confidence. Therefore, in terms of usability and efficacy measures, OO techniques represent an evolutionary change from structured ones.

Our findings have a few implications for IT managers. It is commonly believed that usability and self-efficacy predict task performance, job satisfaction, and other work-related behavioral and attitudinal variables. Managers who desire to implement OO techniques should target those individuals with prior knowledge on both data and process models and those with no prior exposure to structured techniques at all; these people are more likely to bring desirable consequence after training or retraining. Also, our findings contradict the common concern that prior knowledge may interfere with the learning of OO techniques. On the contrary, this study found that prior knowledge helps improve trainee reactions to OO techniques, which in turn improve actual learning.

Information Systems is a field full of constant changes. More often than in any other field, IT workers see not just incremental adjustments but fundamental shifts in the way they use technologies. In just the last two decades, we have seen databases evolve from flat files, to hierarchical, to relational, and to OO models, and operating systems from DOS, to Windows, and to Web-based user interfaces. At each turn, IT workers are forced to transfer their existing skills and learn new ones. Whether a transition is successful or not is often measured by their after-transition job performance and satisfaction, which in turn are determined by their efficacy and usability reactions. The current study makes a contribution by introducing the multiconstraint theory to study these phenomena. Future research could apply the theory to other contexts and examine, for example, how prior knowledge on legacy systems predicts trainee reactions to new systems.

REFERENCES

1. Adler, P. S. and Winograd, T. A. (1992) Usability: turning technologies into tools, Oxford University Press, New York, NY.
2. Agarwal, R., Prasad, J. and Zanino, M. C. (1996a) In SIGCPR/SIGMIS '96Denver, CO, pp. 259-267.
3. Agarwal, R., Sinha, A. and Tanniru, M. (1996b) International Journal of Human and Computer Studies, 45, 639-667.
4. Boehm-Davis, D. and Ross, L. (1992) International Journal of Man-Machine Studies, 36, 1-19.
5. Compeau, D. R. and Higgins, C. A. (1995) MIS Quarterly, 19, 189-211.
6. Davis, F. D. (1989) MIS Quarterly, 13, 319-340.
7. Detienne, F. (1995) Human Computer Interaction, 10, 129-169.
8. Gentner, D. (1988) Child Development, 59, 47-59.
9. Gick, M. and Holyoak, K. (1983) Cognitive Psychology, 15, 1-38.
10. Gist, M., Schwoerer, C. and Rosen, B. (1989) Journal of Applied Psychology, 74, 884-491.
11. Gist, M. E. and Mitchell, T. R. (1992) Academy of Management Review, 17, 183-211.
12. Gist, M. E., Stevens, C. K. and Bavetta, A. G. (1991) Personnel Psychology, 44, 837-861.
13. Goldstein, I. L. and Gilliam, P. (1990) American Psychologist, 45, 134-143.
14. Holyoak, K. J. and Thagard, P. (1989) Cognitive Science, 13, 295-355.
15. Holyoak, K. J. and Thagard, P. (1995) Mental leaps: Analogy in creative thought, MIT Press, Cambridge, MA.
16. Holyoak, K. J. and Thagard, P. (1997) American Psychologist, 52, 35-44.
17. Lee, A. and Pennington, N. (1994) International Journal of Human Computer Studies, 40, 577-601.
18. Marakas, G. M., Johnson, R. D. and Yi, M. Y. (1999) Indiana University, Bloomington, IN.
19. Marakas, G. M., Yi, M. Y. and Johnson, R. D. (1998) Information Systems Research, 9, 126-163.
20. Morris, M., Speier, C. and Hoffer, J. (1999) Decision Sciences, 30, 107-135.
21. Nelson, H. J., Armstrong, D. and Ghods, M. (2002) Communications of the ACM, (Forthcoming).
22. Pennington, N., A. L. and Rehder, B. (1995) Human Computer Interaction, 10, 171-226.
23. Sircar, S., Nerur, S. P. and Mahapatra, R. (2001) MIS Quarterly, 25, 457-471.
24. Venkatesh, V. (2000) Information System Research, 11, 342-365.
25. Vessey, I. and Conger, S. (1994) Communications of the ACM, 37, 102-111.
26. Warr, P. and Bunce, D. (1995) Personnel Psychology, 48, 347-375.