Chapter 6
Exploring the Dimensions and Effects of Computer Software Similarities in Computer Skills Transfer

Yuan Li
Columbia College, USA

Kuo-Chung Chang
Yuan Ze University, Taiwan

ABSTRACT

Computer software similarities play important roles in users' skills transfer from one application to another. Despite common software attributes recognized in extant literature, a systematic understanding of the components and structure of software similarities has not been fully developed. To address the issue, a Delphi study was conducted to explore the underlying dimensions of software similarities. Inputs gathered from 20 experienced Information Systems instructors show that Computer Software Similarity is a multi-dimensional construct made up of interface similarity, function similarity, and syntax similarity. Each dimension consists of software attributes that users perceive to be transferable in learning new applications. A field study was carried out to test the impact of the construct. Results from a survey on students' learning two software applications confirm the expectation that Computer Software Similarity facilitates the students' skills transfer between the applications. These studies provide a basis to better design training programs for improved training performance.

INTRODUCTION

Computer skills transfer is an important area of research in end user computer training (Powell & Moore, 2002). Also known as the carryover effect (Agarwal, Sambamurthy & Stairs, 2000) and transfer of training (Salas & Cannon-Bowers, 2001), it refers to the reuse of a person's computer skills acquired from previously learned software applications in new applications (Singley & Anderson, 1985). For scholars doing research in this area, an important mission is to understand why

DOI: 10.4018/978-1-4666-2059-9.ch006

it happens and how to facilitate skills transfer. A popular explanation is that computer skills transfer occurs when a to-be-learned software application shares similar attributes with previously learned applications, and the extent of similarity determines the amount of transfer. A number of common software attributes were recognized, such as menu items (Smelcer & Walker, 1993), functional keys (Polson, Bovair, & Kieras, 1987), user commands (Singley & Anderson, 1988), and dialog structures (Foltz, Davies, Polson, & Kieras, 1988); more recent studies helped to uncover additional items, including the database knowledge in Enterprise Resource Planning (ERP) system training (Coulson, Zhu, Stewart, & Rohm, 2004) and the knowledge structure in computer gaming (Schuelke, Day, McEntire, Boatman, Boatman, Kowollik, & Wang, 2009).

Progress has been made in this area, but notably a satisfying solution has yet to be developed to adequately specify what the common software attributes are in computer skills transfer. Firstly, most research has been focused on only one type of software attribute, but a cross-validation of the impact of multiple attributes was seldom made, questioning the validity of previous findings in the existence of multiple attributes. Secondly, most studies were based upon a single area of software, such as databases (Shayo & Olfman, 1998) or text editors (Singley & Anderson, 1988), but limited attention was paid to the software attributes that are shared across application areas. Due to these limitations, the extant literature suffers from the lack of a more comprehensive and generalizable list of attributes. The consequence is that when new generations of computer applications are introduced to a networked, ubiquitous computing environment (Olson & Olson, 2003), new features of the applications may not be properly integrated in existing knowledge frameworks to effectively predict the transferability of skills (Urbaczewski & Wheeler, 2001). Considering the importance of computer skills transfer for both research and practice (Agarwal et al., 2000), the quest for common software attributes should be continued.

This research attempts to achieve two objectives: to search for a more comprehensive and generalizable list of common software attributes to measure computer software similarity, and to find empirical evidence of the impact on computer skills transfer. To this end, two interrelated studies were conducted (Poston & Royne, 2008): the first study applied an inductive approach in search of a set of common software attributes due to the lack of sufficient theoretical basis. Specifically, the Delphi method (Schmidt, 1997) was applied to solicit opinions from a group of experienced computer instructors to aid in the recognition of common software attributes and the discovery of the underlying dimensions of the items. Based on the outcomes, the second study was conducted to test the impact of the common software attributes on computer skills transfer. The results show a three-dimensional model of computer software similarity (including function similarity, interface similarity, and syntax similarity), with each dimension containing a number of software attributes that are predictive of skills transfer. The list of items covers various aspects of software features and is not restricted to a single area of applications, making it more comprehensive and generalizable than those recognized in previous studies. This has significant meanings for further research and practice.

The article is structured as follows. First, past research in this area is briefly summarized, with achievements and limitations recognized. Next, the approach to the research questions is introduced, followed by the description of the Delphi study. Then, the empirical test is presented to verify the impact of the common software attributes on computer skill transfer. Findings are finally summarized and the limitations and implications of the research are discussed.