
Designing for Curiosity-Driven Software Learning

Volodymyr Dziubak

University of Manitoba
66 Chancellors Cir,
Winnipeg, MB R3T 2N2, Canada
vdziubak@cs.umanitoba.ca

Andrea Bunt

University of Manitoba
66 Chancellors Cir,
Winnipeg, MB R3T 2N2, Canada
bunt@cs.umanitoba.ca

Paste the appropriate copyright/license statement here. ACM now supports three different publication options:

- **ACM copyright:** ACM holds the copyright on the work. This is the historical approach.
- **License:** The author(s) retain copyright, but ACM receives an exclusive publication license.
- **Open Access:** The author(s) wish to pay for the work to be open access. The additional fee must be paid to ACM.

This text field is large enough to hold the appropriate release statement assuming it is single-spaced in Verdana 7 point font. Please do not change the size of this text box.

Each submission will be assigned a unique DOI string to be included here.

Introduction

The versatility of feature-rich software makes them practically impossible to master –there is always the opportunity to learn something new, whether it would be a new software feature, or a new approach for a familiar task. While most research on software learnability has studied software learning that is driven by a necessity to complete a certain task (e.g., [1,3,4]), our research investigates software learning that is driven by a users' curiosity and intrinsic motivation to expand their software skills. We investigate how to design and build tools to support different activities associated with the process, such as curiosity-driven exploration of software tutorials and reflection on curiosity-driven software learning.

Supporting Curiosity-Driven Exploration of Software Tutorials on Social Media

In our recent work [2], we found that professional digital designers often monitor social media, such as Twitter, curious about learning new software capabilities and the latest software techniques. Twitter is a great platform for such monitoring, as its users often share links to new and popular software tutorials. However, tweets do not provide designers with enough information to decide whether a tutorial has valuable knowledge and whether it is worth spending their time watching or reading it.

We designed Switter – a Twitter client tailored for curiosity-driven software learning. Switter augments each tweet that shares a Photoshop tutorial with the list of Photoshop tools and commands mentioned in the tutorial. Switter also allows its users to navigate tweets that contain tutorials using an interactive replica of Photoshop’s interface. A weeklong field study with nine expert Photoshop users showed Switter’s utility for curiosity-driven software learning. Specifically, Switter helped participants to locate tutorials which might contain new knowledge, validate their existing knowledge, and learn new software skills and techniques.

Supporting Reflection on Curiosity-Driven Software Learning

As our next step in studying curiosity-driven software learning, we are investigating whether reflecting on one’s learning history could potentially direct their learning experience in the future. Imagine a designer whose morning routine involves looking through recently-posted software tutorials on social media, hoping to learn new software techniques. Our findings suggest that designers who are engaged in such curiosity-driven learning generally do not have defined learning goals and do not expect to learn much [2]. However, if the designer had the ability to reflect on their learning history, they might be able to identify exploration paths that led to tutorials that taught them new software techniques. Such insight might then improve the designer’s future curiosity-driven learning and increase their chances of learning something new. We aim to investigate how to design tools for reflection that could potentially direct designers in their curiosity-driven software learning, and improve their learning experience.

About the Authors

Volodymyr Dziubak is a 3rd-year Ph.D. student in the Computer Science Department at the University of Manitoba. Volodymyr’s research involves studying and supporting work practices of professional digital designers and artists, such as curiosity-driven software learning and inspiration seeking. Volodymyr previously he received his B.Sc. (2011) and M.Sc. (2013) degrees in Computer Science in Ukraine (his home country). Before starting his Ph.D., he worked as an interaction designer in an online learning company in Netherlands.

Dr. Andrea Bunt is an associate professor in the Department of Computer Science at the University of Manitoba, where she co-directs the HCI lab. Andrea’s research focuses on software learnability, community-authored help and intelligent interactive systems.

References

1. A. Bunt, P. Dubois, B. Lafreniere, M.A. Terry, and D.T. Cormack. 2014. TaggedComments: Promoting and Integrating User Comments in Online Application Tutorials. *Proc. CHI*, 4037–4046.
2. V. Dziubak, P. Dubois, A. Bunt, and M. Terry. 2016. Switter: supporting exploration of software learning materials on social media. *Proc. DIS*, 1209–1220.
3. C. Kelleher and R. Pausch. 2005. Stencils-Based Tutorials: Design and Evaluation. *Proc. CHI*, 541–550.
4. G. Laput, E. Adar, A. Arbor, M. Dontcheva, and W. Li. 2012. Tutorial-Based Interfaces for Cloud-enabled Applications. *Proc. UIST*, 113–122.