## DISSERTATION DEFENSE

# Rebecca Krosnick
## Improving Web Automation Tools through UI Context and Demonstration

Monday, November 6, 2023
9:30am – 11:30am
3725 Beyster
Hybrid – Zoom  Passcode: 488630

**ABSTRACT:** User interface (UI) automation allows people to perform UI tasks programmatically and can be helpful for computer or smartphone tasks that are tedious, repetitive, or inaccessible. UI automation works by programmatically mimicking a user's interactions on a UI, for example clicking a button or typing into a text field. Traditionally people create UI automation macros by writing code, which requires programming expertise and familiarity with UI technologies. Researchers have explored direct manipulation interfaces and programming-by-demonstration (PBD) to make creating UI automation more accessible for people with less programming experience. With PBD, the user provides demonstrations of how they want their program to behave in a small set of scenarios, and the system then infers a generalized program. A key challenge of PBD is correctly inferring the user's intent.

In this thesis, I address important challenges in authoring UI automation macros by leveraging user-provided demonstrations and parameters, and structural patterns in the UI to infer generalized automation; and in understanding UI automation macros by 1) highlighting selected elements on the target UI, 2) visualizing high-level behavior through sequences of actions and UIs visited, 3) visualizing generalizations through color-coding UI elements and grouping corresponding UIs, and 4) providing feedback on validity and uniqueness of element selection logic. First, I conducted two studies observing how programmers write automation code. One of the key challenges participants experienced was identifying appropriate UI element selection logic. Next, I designed two programming-by-demonstration systems, ParamMacros and ScrapeViz, that enable users to create automation macros without writing code. ParamMacros supports users in creating macros that answer parameterized questions about website content. The user provides a natural language question they want to answer, identifies which parts of that question can be parameterized, and then demonstrates how to answer one instance of that parameterized question -- ParamMacros then leverages structural patterns in the website DOM to infer a generalized macro. ScrapeViz supports users in creating distributed hierarchical web scraping macros, similarly leveraging demonstration and DOM structural pattern approaches. ScrapeViz provides users a storyboard-like visualization illustrating macro behavior across UI pages and an interactive table linking scraped data to their page source.

**CHAIR:** Prof. Steve Oney