Leilani Battle,
University of Washington

Alvitta Ottley,
Washington University
in St. Louis

Testing Theories of Task in Visual Analytics

Theory research plays a critical role in many areas of computer science, including artificial intelligence, programming languages, and human-computer interaction. However, striving to encapsulate the human experience within a math equation or a concise model can seem reductive at best, especially when the goal is to design software that *actively cooperates* with people to achieve complex goals, such as in visualization research. That being said, naive reductions can also help researchers test their understanding of the complexities of human behavior. This column will explore the role of theory in visualization research and our vision for the future of visualization theory work.

Visualization has traditionally used qualitative theory approaches to produce guidelines for designing new visualization systems. To do this, researchers conduct user studies and/ or literature surveys to collect relevant data. They then analyze this data for recurring themes and patterns, often formalized as *taxonomies*, which can be thought of as theoretical models of user analysis behavior. For example, Matthew Brehmer and Tamara Munzner identify 11 interaction types commonly used in visualization interfaces [1], such as *filtering* irrelevant data points via range sliders and buttons, *navigating* a dataset via dragging or scrolling, and *selecting* data points of interest by hovering, clicking, or highlighting. Taxonomies can also represent more sophisticated patterns within the visual analysis process, such as sequences of common interactions or even directed flow charts.

With the rise of data-driven modeling techniques such as machine learning, deep learning, and reinforcement learning, researchers are also interested in *computational models* of user analysis behavior [2,3]. For example, one could instrument a visualization system to log users' interactions and extract features from these logs to train a machine learning model. With this approach, models of user analysis behavior could potentially be derived automatically rather than by hand. Since computational models are already in a programmed form, they can also be integrated directly into visualization systems, enhancing nascent features such as recommending new visualizations to explore as a person analyzes their dataset [4]. However, a significant challenge is generating cleanly labeled input data to train the models. Although model-training software is automated, the labeling of input data is often still a manual process, eroding the benefits of using this technique over manually derived theoretical models.

Rather than keeping these two approaches—taxonomies and computational models—separate, we argue for *integrating* them, thereby enabling the strengths of one approach to mitigate the weaknesses of the other. For example, theoretical taxonomies can provide classification labels for computational models. If we can write programs to apply these labels to interaction logs, we can also train machine learning models to represent these theories through code that we can then integrate into new and existing visualization systems.

To this end, we embarked on a project to investigate the landscape of existing visualization taxonomies and determine which theories we can use to label user interaction logs programmatically [5]. We observe an exciting pattern among these taxonomies: They seem to define user analysis behaviors as a form of expression that parallels how people express themselves through natural language, that is, through words, sentences, and paragraphs. In this case, individual interactions represent words; interaction subsequences represent sentences; and interaction subsequences can be chained together to form larger structures akin to paragraphs.

Building on this insight and core concepts from automata theory, we developed a new framework for expressing individual visualization taxonomies as *regular grammars*. A regular grammar consists of a set of terminals, a set of nonterminals, and a set of production rules. Terminals form a dictionary of primitive values akin to words, examples being the drag slider, zoom, and hover interactions mentioned above. Nonterminals are variables representing aliases for any structure of interest, including terminals. Production rules are functions that can be assigned to a nonterminal. Finally, production rules can be as simple as mapping one terminal to a corresponding nonterminal. For example, labeling a logged slider drag with its corresponding interaction label (*filter*). They can also be more sophisticated, such as mapping a common subsequence of logged interactions to a nonterminal or mapping entire regular expressions to a nonterminal.

Our approach can express virtually any existing visualization taxonomy programmatically by defining it first as a regular grammar (see Figure 1 for an example). To demonstrate the versatility of our approach, we used it to define regular grammars for seven visualization taxonomies from the literature. We then wrote code to apply these grammars to three different interaction log
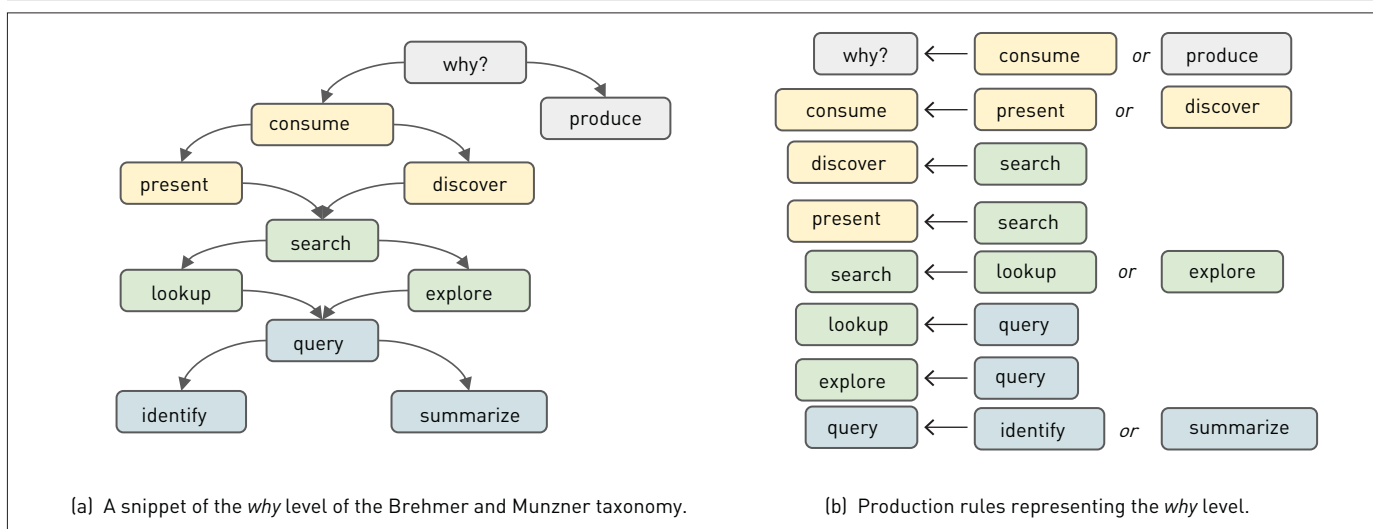
(a) A snippet of the *why* level of the Brehmer and Munzner taxonomy.

(b) Production rules representing the *why* level.

Figure 1: The Brehmer and Munzner taxonomy has three levels [1]: *how* a user interacts with a visualization (lowest level), *what* is being interacted with, and *why* the user is performing these interactions (highest level). A snippet of the *why* level is depicted in (a). In (b), we show how to express concepts from the *why* level using our regular grammar approach. Higher-level motivations, such as presenting and discovering insights, connect to lower-level analysis tasks, such as searching and querying the underlying data.

datasets collected by visualization researchers, yielding an innovative method for mapping taxonomies to user interaction logs. We analyzed the resulting mappings with the goal of assessing the breadth and depth of existing taxonomies for supporting log-data analysis. We find that existing visualization taxonomies are often designed to be highly generalized, which can clash with the contextually rich logs generated by typical visualization systems. For example, Tableau Desktop records at least eight different types of filtering events [6]. In contrast, there is only one filter terminal present in existing taxonomies, suggesting that we lose important contextual information when going from system log events to abstract taxonomy labels.

Our findings point toward an exciting new direction for visualization theory work: designing taxonomies as context-aware *functions* rather than as static *labels*. With functions, researchers can formally specify which contextual cues to adopt in their taxonomic definition to support richer log analyses. For example, input device type (touchscreen, mouse and keyboard, VR headset, etc.) could become an input parameter to determine which types of *filter* interactions may be applicable from a given taxonomy. Our regular grammar approach can easily accommodate these changes by including parameters within

the terminal set and adding production rules to define relationships between parameters and other variables.

More broadly, our research encourages a more precise approach to visualization theory development, making it easier to validate research findings as well as adopt innovations from outside visualization. For example, by representing visualization theories as programs, we can evaluate them using techniques from programming languages, software engineering, and system design. To this end, we are developing a new language for declaratively specifying a user's analysis task in terms of the user's objective in completing the task and insights discovered while completing the task. Our larger vision is to develop *integrative* technology that broadens the impact of our community's theory contributions within and beyond visualization.

ENDNOTES
1. Brehmer, M. and Munzner, T. A multi-level typology of abstract visualization tasks. *IEEE Transactions on Visualization and Computer Graphics 19*, 12 (2013), 2376–2385.
2. Monadjemi, S., Garnett, R., and Ottley, A. Competing models: Inferring exploration patterns and information relevance via Bayesian model selection. *IEEE Transactions on Visualization and Computer Graphics 27*, 2 (2020), 412–421.
3. Xu, K., Ottley, A., Walchshofer, C., Streit, M., Chang, R., and Wenskovitch, J.

Survey on the analysis of user interactions and visualization provenance. *Computer Graphics Forum 39*, 3 (2020), 757–783.
4. Zeng, Z., Moh, P., Du, F., Hoffswell, J., Lee, T.Y., Malik, S., Koh, E., and Battle, L. An evaluation-focused framework for visualization recommendation algorithms. *IEEE Transactions on Visualization and Computer Graphics 28*, 1 (2022), 346–356.
5. Gathani, S. Monadjemi, S., Ottley, A., and Battle, L. A programmatic approach to applying visualization taxonomies to interaction logs. *arXiv preprint arXiv:2201.03740*
6. Battle, L. and Heer, J. Characterizing exploratory visual analysis: A literature review and evaluation of analytic provenance in tableau. *Computer Graphics Forum 38*, 3 (2019), 145–159.

⬤ **Leilani Battle** is an assistant professor in the Allen School at the University of Washington (UW). Her research focus is on developing interactive data-intensive systems that aid analysts in performing complex data exploration and analysis. She holds an M.S. and a Ph.D. in computer science from MIT and a B.S. in computer engineering from UW.
→ leibatt@cs.washington.edu

⬤ **Alvitta Ottley** is an assistant professor of computer science and engineering and psychological and brain sciences (courtesy) at Washington University in St. Louis. Her research improves decision making, learns from interactions, and personalizes visualizations. She received an NSF CRII to support medical decision making and an NSF CAREER to design context-aware visualizations.
→ alvitta@wustl.edu