An Exploratory Study of Web Foraging to Understand and Support Programming Decisions

Jane Hsieh[1], Michael Xieyang Liu[2], Brad A. Myers[2], and Aniket Kittur[2]

[1] Department of Computer Science
Oberlin College
Oberlin, OH, USA
jhsieh@oberlin.edu

[2] Human Computer Interaction Institute
Carnegie Mellon University
Pittsburgh, PA, USA
{xieyangl, bam, nkittur}@cs.cmu.edu

***Abstract*— Programmers consistently engage in cognitively demanding tasks such as sensemaking and decision-making. During the information-foraging process, programmers are growing more reliant on resources available online since they contain masses of crowdsourced information and are easier to navigate. Content available in questions and answers on Stack Overflow presents a unique platform for studying the types of problems encountered in programming and possible solutions. In addition to classifying these questions, we introduce possible visual representations for organizing the gathered information and propose that such models may help reduce the cost of navigating, understanding and choosing solution alternatives.**

Keywords*— *information-foraging, exploratory programming, decision-making

I. Introduction

Programming is not solely comprised of coding. Developers spend a significant amount of time foraging for and making sense of the information they need before making changes to a software system [1]. Previous empirical studies have revealed that as much as 35-50% of programming time is spent exploring and seeking information [1-2]. During this time, developers must engage in a variety of cognitive activities such as understanding unfamiliar pieces of code and deciding how to modify existing pieces of software, as well as higher-level decisions such as choosing which APIs to use.

These exploratory activities are *foraging tasks* [4] where developers seek to collect information about the different options or ways of implementing desired programs. Often, the programmer attempts to achieve more than just gathering such content, they also engage in *sensemaking* [4] so that the newly acquired knowledge can be utilized to make decisions about how to implement or amend their own code. In this study, we use available content from Stack Overflow posts to gain insight about the *categories of problems* that programmers experience and the *types of information* that guides their sensemaking and decision-making processes.

We began by manually analyzing a preliminary sample of 92 questions and classifying them into four broad categories of inquiries: methodological (31% of the questions), debugging (29%), conceptual (20%), and concept-specific (20%). These categories closely resemble the types previously identified by a clustering method from machine learning [5]. Next we present the *comparison table* for representing questions involving decision-making tasks in some of these categories. When analyzing the first sample, we noticed that many of the questions contained alternative solutions for solving similar (if not completely equivalent) problems. Furthermore, these additional answers (supplementary to the accepted answer) receive upvotes from the community for the different criteria that they each fulfill. Building off of this observation, we identified that a significant portion of Stack Overflow questions relate to decision making tasks and can therefore be represented in the form of a comparison table.

To verify this hypothesis, we sampled a larger set of questions and attempted to represent the question and answer posts with a table view. The sample query was fine-tuned to capture not only the individually popular questions, but also the "long tail" questions that collectively make up a significant portion of the search traffic [6].

Our results showed that the comparison table is a suitable way of representing information from about half of the Stack Overflow question posts. The usefulness of the comparison table encourages the development and construction of assistive tools utilizing these theoretical models.

II. Sampling Method and Results

A. Preliminary Sampling

In order to find an appropriate sample of questions with diverse topics, we used a variety of search queries. Readily available are the built-in Stack Overflow filters: *relevance*, *newest, votes,* and *active*. However, to obtain any results, these filters must be accompanied by a nonempty search term. There also exists filters that do not require specific search terms: *interesting, featured, hot, week* and *month*. Table I shows the preliminary search queries and the number of questions sampled from each query result.

TABLE I. Preliminary Sampling Queries

Queries	*Questions*
"how to answers:10" [a] with *active* filter	21
"which should views:500000" [b]	20
Hot filter ("hottest" questions today)	20
Month filter	19
"how to" with *votes* filter	13

[a] The tag "answers:10" results in only questions with 10 or more answers

[b] Similarly, "views:500000" filters out questions with less than 500000 views

978-1-5386-4235-1/18/$31.00 ©2018 IEEE

Since these questions were manually categorized, the classification of the sample questions may be subject to bias. However, the question categories were created by the researcher before encountering the categories found in the clustering method utilized by Allamanis and Sutton [5], and to our surprise there was a correspondence between four of the broad categories extracted from their method and ours:

Methodological: questions where the programmer forages for methods or code snippets to achieve a set of specifications.

Debugging: questions with specific context such as error messages.

Conceptual inquiries: abstract questions about concepts not explained comprehensively in the API documentation.

Concept-specific: questions where the forager seeks to understand how to use particular methods or commands.

When forming these categories and classifying questions into their respective types, we noticed that most of the *methodological* and *concept-specific* questions (51% of all posts) contain answers with multiple options. Each solution is valuable to the community due to a unique set of criteria that they may fulfill. Frequent criteria include factors such as performance/speed, compatibility (with libraries, browser and language versions, etc.), and readability.

Such questions and their multitude of crowdsourced answers suggests that half or more of the questions posted can be visually represented with a *comparison table* - where rows consist of options and columns display the various criteria. For each criteria that an option fills, their intersecting cell can be marked to symbolize relevance. This visualization may help users to not only understand the different options, but also guide them in choosing the one that is most appropriate to their specific situation. To evaluate the practicality of this representation, we needed a larger sample of questions to test the proportion of questions posted that can be represented with the comparison table.

B. Test of Model using Refined Sample Queries

We utilized two new queries to test the usefulness of the comparison table. This stage takes advantage of the advanced search filters of Stack Overflow and how to use them without a search term. Hence, the first 50 questions were collected using the query "*is:question views:2360000*", which asks for all questions with 2.36 million or more views (there were exactly 50 as of 7/12/18).

However, choosing questions with the most views can be considered cherry-picking since the most popular questions may only be representative of a narrow set of topics, and indeed we do observe a high correlation between popular questions and their compatibility with the comparison table. It is important we consider not only this specific set of popular topics, since previous research has indicated that only a small portion of the search interests from individual information seekers lie within the most popular questions. The remaining interests of the population makes up the majority of topics in the "long tail" – topics which are less frequently viewed in total, but collectively they cause a significant portion of the total search traffic [5][6].

To sample questions that belong more to the "long tail", we composed a decidedly restricted query: *"is:question created:2018-06-15 answers:3"* - to find questions with three or more answers that were asked on a particular day. A total of 90 questions were assessed with this query, and we found that 88% of the most viewed questions naturally fit well with a comparison table. And in the final sample, we discover that approximately half (49%) of the questions were representable with the proposed table. This result is consistent with the hypothesis that questions involving decisions (51% of both samples) can be depicted in a tabulated format.

III. Relevance and Implications

This study is intended to motivate the design of mental models such as the comparison table to reduce the cost of collecting and organizing information for foragers. Many tools can be built based upon proposed designs, and our research group is in the process of developing a web-browsing tool that utilizes the comparison table. Future work is needed to test that such tools are useful to developers as they forage for information in real-life programming contexts. It would also be interesting to study to what extent decision questions like these are common in other domains besides programming, and if our proposed tools could help in those situations as well.

Acknowledgment

This research is supported in part by the CMU REUSE program, funded by NSF grant CCF-1560137 and in part by NSF grant CCF-1814826. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the NSF.

References

[1] A. J. Ko, B. A. Myers, M. J. Coblenz, and H. H. Aung. 2006. "An Exploratory Study of How Developers Seek, Relate, and Collect Relevant Information during Software Maintenance Tasks," IEEE Trans. Softw. Eng. 32, 12 (December 2006), 971-987.

[2] D. J. Piorkowski, S. D. Fleming, I. Kwan, M. M. Burnett, C. Scaffidi, R. K.E. Bellamy, and J. Jordahl. 2013. "The whats and hows of programmers' foraging diets," Proceedings CHI '2013. 3063-3072.

[3] M. Allamanis and C. Sutton. 2013. "Why, when, and what: analyzing stack overflow questions by topic, type, and code," Proceedings of the IEEE Conf. on Mining Software Repositories (MSR '13). 53-56.

[4] K. Fisher, S. Counts, and A. Kittur. 2012. "Distributed sensemaking: improving sensemaking by leveraging the efforts of previous users," In Proceedings CHI '12. 247-256.

[5] M. S. Bernstein, J. Teevan, S. Dumais, D. Liebling, and E. Horvitz. 2012. "Direct answers for search queries in the long tail." In Proceedings CHI '12. 237-246.

[6] B. Evans and S. Card. 2008. "Augmented information assimilation: social and algorithmic web aids for the information long tail," In Proceedings CHI '08. 989-998.