



Adopting Third-party Bots for Managing Online Communities

SOHYEON HWANG, Northwestern University, USA

CHARLES KIENE, University of Washington, USA

SERENE ONG, Northwestern University, USA

AARON SHAW, Northwestern University, USA

Bots have become critical for managing online communities on platforms, especially to match the increasing technical sophistication of online harms. However, community leaders often adopt *third-party* bots, creating room for misalignment in their assumptions, expectations, and understandings (i.e., their technological frames) about them. On platforms where sharing bots can be extremely valuable, how community leaders can revise their frames about bots to more effectively adopt them is unclear. In this work, we conducted a qualitative interview study with 16 community leaders on Discord examining how they adopt third-party bots. We found that participants addressed challenges stemming from uncertainties about a bot's security, reliability, and fit through emergent social ecosystems. Formal and informal opportunities to discuss bots with others across communities enabled participants to revise their technological frames over time, closing gaps in bot-specific skills and knowledge. This social process of learning shifted participants' perspectives of the labor of bot adoption into something that was satisfying and fun, underscoring the value of collaborative and communal approaches to adopting bots. Finally, by shaping participants' mental models of the nature, value, and use of bots, social ecosystems also raise some practical tensions in how they support user creativity and customization in third-party bot use. Together, the social nature of adopting third-party bots in our interviews offers insight into how we can better support the sharing of valuable user-facing tools across online communities.

CCS Concepts: • **Human-centered computing** → **Empirical studies in collaborative and social computing**; **Collaborative and social computing theory, concepts and paradigms**.

Additional Key Words and Phrases: bots, online communities, community management, moderation, technology adoption

ACM Reference Format:

Sohyeon Hwang, Charles Kiene, Serene Ong, and Aaron Shaw. 2024. Adopting Third-party Bots for Managing Online Communities. *Proc. ACM Hum.-Comput. Interact.* 8, CSCW1, Article 216 (April 2024), 26 pages. <https://doi.org/10.1145/3653707>

1 INTRODUCTION

Bots made by users have become critical to the successful management of online communities, helping communities moderate content, manage membership, sustain engagement, and more broadly extend and customize platform affordances. As communities within a platform often have similar needs, bots made by users are likely to align with existing needs and practices of other users, suggesting that they can be readily shared and adopted across communities to positive effect [18, 31]. However, prior work also finds that users, especially those without technical backgrounds,

Authors' addresses: [Sohyeon Hwang](#), Northwestern University, Evanston, Illinois, USA; [Charles Kiene](#), University of Washington, Seattle, WA, USA; [Serene Ong](#), Northwestern University, Evanston, Illinois, USA; [Aaron Shaw](#), Northwestern University, Evanston, Illinois, USA.



This work is licensed under a Creative Commons Attribution International 4.0 License.

© 2024 Copyright held by the owner/author(s).

ACM 2573-0142/2024/4-ART216

<https://doi.org/10.1145/3653707>

face significant challenges in bot adoption [21, 31, 59]. Because understanding the premise of a technology is important in using it more effectively [48], differences in assumptions, knowledge, and expectations about bots across a broader set of potential adopters impact how they can effectively use those bots or even recognize the value bots can offer them. For example, bot users and developers on the Reddit platform perceive and anticipate the value of bots in different ways [e.g., see 42], such as what exactly a bot can enable them to do and how to best leverage it. These incongruities among users undermine the potential benefits of sharing user-made bots by limiting who can make effective use of those tools and meaningfully assess their impacts on communities.

How can community leaders shift their understandings of bots and leverage them to address community needs, especially when bots are not officially part of platforms but instead third-party tools made by fellow users? That many platforms hosting communities are arranged in a decentralized manner further muddies where community leaders might go to discover and assess bots. What do people do to adopt the bots that meet their needs in the face of these challenges? How does this process enable them to better leverage shared tools to meet the needs of their communities, or otherwise fail to do so?

This study pursues this inquiry through open-ended interviews with a sample of 16 community leaders on the Discord platform. We make three contributions to prior work on bots and (custom) technology adoption. First, we extend prior work by offering evidence that key challenges of revising technological frames about third-party bots by online community leaders on platforms can be addressed through emergent social ecosystems: user-to-user networks of resources, aid, and knowledge across communities. These formal and informal opportunities to discuss bots with individuals in other communities enabled organizational learning and helped close gaps in bot-specific skills and knowledge observed in prior work. Second, we describe how the social nature of this process shifts participants' perceptions of the substantial (at times costly) labor involved in bot adoption. In particular, our interviews suggest that communal and collaborative approaches to bot adoption can be advantageous in multiple ways. Finally, we describe a tension in how social ecosystems align peoples' technological frames about the nature, use, and value of bots, even as community leaders aimed to customize their communities through a deliberate curation of multiple bots. Together, our findings offer insight into how we can better support communities in assessing and using third-party bots and other novel user-facing tools that would meet their needs, particularly by examining intercommunity dynamics.

2 RELATED WORK

The adoption of user-made and customized tools like bots in online communities touches on several key streams of prior CSCW and social computing research. One body of this work demonstrates that bots constitute an important part of online community management, with both positive and negative consequences. Another body of work on the adoption of bots in communities has emphasized bots as valuable user innovations that can be even more valuable when shared. Meanwhile, broader research on technology adoption suggests that shared understandings articulated through technological frames [49] shape how groups make sense of and use novel, third-party tools like bots.

2.1 The impact of bots in online community management

Bots and other user-made software customizations have become critical to the success of online communities and platforms [28, 31, 43–47, 58, 61, 71]. In this study, we define bots as independent software agents running alongside existing platform infrastructure that automate tasks and aspects of community management. Typically these bots operate through a platform's *Application Programming Interface* (API), which gives them limited access to the platform's data and opens up possibilities for decentralized innovation and sharing [26]. Platform support of APIs and bots

exemplifies mechanisms of empowering end-user autonomy, along similar lines to user toolkits [68] and modular “design rules” [1] that support distributed and “incremental” [2] customizations, modifications, and extensions of social computing systems. These customizations enable people to create extensions which align with community needs and enrich their experience of using platforms. For example, Poretski and Arazy [51] found a 15% increase in sales when gaming companies successfully engaged communities in *modding*, or making custom software modifications.

Bots play various roles to enrich the experience of online communities. Often, bots take on *administrative and governance-related* roles that facilitate community management, such as identifying rule violations, greeting newcomers, assessing content quality, or fixing technical issues like broken links [e.g., on Wikipedia in 75]. In these capacities, bots automate a series of otherwise tedious tasks that relieve the burden of labor on human administrators and contributors, encoding various social mechanisms into algorithmic ones to aid the governance of a large-scale community [46]. Meanwhile, Seering et al. [58] showed how bots also take on *social* roles in communities to enhance engagement, share information, send moderation warnings, offer mini-games, and promote users. In some situations, bots may be preferable to engage users about sensitive topics that may be socially taboo or difficult to discuss with others [53].

The value of bots in taking on various tasks comes from the fact that they automate tasks at scale [19, 23, 61]. The importance of scalability is perhaps most salient when it comes to the moderation of hateful speech, spam, and other unwanted behaviors. As the rate of interactions becomes difficult for humans to manage, bots can algorithmically cut down what people must manually sift through and moderate, as well as effectively combat malicious bots automating the spread hateful content, misinformation, spam, scams, and so on [44]). For example, Chandrasekharan et al. [7] developed *Crossmod*, a moderation tool which could detect comments on Reddit that would have been removed by moderators with an accuracy of 86%, demonstrating how bots could automate the moderation of a great bulk of content. The usefulness of bots to this end is particularly salient in synchronous chat-based online communities, which face pressures to address issues of scale, engagement, and moderation in *real-time* [34, 72].

Geiger [17] notes how the automation of tasks by bots turns bots into important infrastructural elements of platforms. An important implication of this framing of bots-as-infrastructure is that when bots are successfully added as seamless attributes of a platform, they become relatively invisible [62]. This invisibility is consequential because bots are widely used and can have profound effects on communities. For example, on the peer production community Wikipedia, bots and various technological tools scaffold the production and maintenance of encyclopedic content [46, 47]. At the same time, these tools can have unintended effects, such as disproportionately penalizing newcomers and discouraging their retention as valuable contributors [25]. More broadly, researchers note how automated tools retain critical limitations [5, 24], and algorithmic approaches can encode biases [12]. As a result, ensuring that users can understand, anticipate, and evaluate the potential effects of bots they adopt is crucial. For example, moderators often lack data and visual support to assess and account for false positives of their automated moderation and filtering tools [31]. While this particular scenario calls for ability to audit after a bot has been adopted, being able to anticipate the effects of bots is also important when people decide to actually use bots, before they put in the labor to set up and maintain the tool in their communities.

2.2 Technological frames in adopting bots and custom user tools

Research on the adoption of bots in online communities has often focused on how users *develop* bots as custom tools or extensions of platforms [17, 18, 36, 64]. This work highlights how users shape their sociotechnical environments in generative ways, underscoring the potential for user

autonomy, creativity, and control. Made by users themselves, tools like bots are inherently well-aligned with users' needs [68]. Because community leaders on a platform often have similar needs and expectations, researchers and designers have noted the potential for such tools to be shared across them [18, 31], in a hopeful vision for how user innovation can democratize opportunities for tailoring digital spaces. As outlined in §2.1, bots designed by or with users have contributed incredible value to communities. This has invigorated calls for the further development of user tools that enable customized management of online experiences on platforms, center users in the design process, and can be widely adopted across communities [32, 33, 55, 74].

In this vein, Jhaver et al. [31] notes that the expertise of community moderators will make them valuable resources in the future development of shared moderation tools. This work describes how community moderators on Reddit use one such shared moderation tool *Automod*, a bot originally created by Reddit user and moderator Chad Birch. Community moderators on Reddit similar practices and concerns, and *Automod* quickly became popular and was eventually made an official Reddit feature. However, the research by Jhaver et al. [31] underscores several challenges and barriers in the use of *Automod*, especially for users who had less technical expertise. For example, as *Automod* used regular expressions, which requires some familiarity with computing, one concern was whether users could adequately anticipate the performance of the tool (e.g., catch false positives that would unfairly penalize community members) to consider unintended impacts of their configuration of the bot.

The case of *Automod* demonstrates how a user-made tool is at once a well-attuned extension of a platform and the focus of technology adoption when it is more broadly adopted by a diverse array of users who share some core needs that the tool addresses. When a user-made bot is shared because the developer feels it will be useful for others, this distinction between the bot as a custom tool and as a *third-party* tool is important. As users adopt a bot as a third-party tool, they face new questions the original developer did not: whether they feel the tool will actually meet their needs, how to set it up in their community, how to troubleshoot it when things go wrong, and so on.

This challenge is not unique to *Automod*; Long et al. [42] observes a general prevalence of misunderstandings about what bots can do and should do between bot users and developers. The literature on technology adoption offers insight into how and why these misalignments between the users who make bots and the users who adopt them matter. In particular, an influential approach in CSCW and social computing comes from Orlikowski [48]'s study of the adoption of the groupware *Lotus Notes* in an organization, which describes how existing mental models about organizational practices lead to the unexpected failures in the adoption of new tools. Subsequent work also on *Lotus Notes* by Orlikowski and Gash [49] articulates the significance and implications of these mental models in the theoretical concept of *technological frames* used to "identify the subset of members' organizational frames that concern the assumptions, expectations, and knowledge they use to understand technology in organizations." Said simply, technological frames describe the shared understandings about a technology's *nature* (its capabilities and functionality), *value* (perspectives of why the technology is adopted), and *use* (the actual, day-to-day usage of the technology) in a group. Taking a social constructionist perspective to understand the effects of technologies in organizational settings [3], research building on the work of Orlikowski demonstrates how various social processes implicated in technological frames matter for how groups adapt, use, and assess the outcomes of technologies [36, 60].

Technological frames offer a useful lens for understanding the challenges in adopting tools like bots because community leaders encompass a wide range of technical backgrounds and may have different goals, even as they have some organizational similarities that make sharing and adopting user-made tools compelling. When adopting any novel tool, peoples' varied frames often need

to shift to be able to accommodate it; without such a shift that enables people to understand the “premises and purposes” of the tool, people are likely to use it in less effective ways [48].

However, how community leaders can shift their frames about third-party bots on platforms remains unclear. In particular, the platforms hosting the communities that would appear to benefit the most from shared user-made bots present some structural challenges for doing so. First, they are often decentralized, wherein communities operate relatively independently as distinct spaces. Second, they do not often integrate user-made bots as official features, which instead operate as third-party tools people choose to use. Additionally, in the case of *Automod* discussed earlier, the bot was a single tool that became officially integrated as part of the Reddit platform. However, a bot is usually one of many user tools that might be adopted. Together, these factors all obfuscate how and where community leaders on platforms can shift their assumptions, expectations, and knowledge about bots to discover shared tools, come to understand their benefits and costs, and anticipate their potential effects as they seek to choose and implement one. Without addressing these aspects of bot adoption, the potential benefits of sharing valuable user-made tools become substantially overshadowed by the constraints of who can use them without incurring heavy costs, both in terms of labor and in terms of unintended consequences. Such costs are ultimately taken on by communities who have serious concerns and needs that bots can address when platform affordances to manage communities are insufficient.

3 RESEARCH DESIGN AND METHODS

3.1 Empirical setting

We conducted interviews with a sample of community leaders with relevant experiences on Discord. Discord is a popular VoIP and synchronous chat-based community platform made up of communities called “servers,” which users join with an invite link. Discord users can have special roles and permissions in servers as moderators, administrators, or the server “owner” (generally the creator of the community). The main interactions in Discord servers are text-based instant messaging (chat) and voice-based communications between users in “channels” dedicated to specific topics. Bots on Discord are automated programs that interact with Discord’s API to perform specific tasks on a Discord server, from providing entertainment to assisting with community management.

We focus on Discord for several reasons. First, Discord has become an important site of study for understanding the success and management of online communities [14, 27, 57], including the use of Discord alongside other community platforms such as Reddit. Although the platform is relatively new, it is large and growing, reporting 140 million monthly (active) users at the end of 2020 [54]. Discord hosts millions of discrete communities that are predominately managed by volunteers, making it a valuable setting for understanding community-level dynamics and patterns across many groups that share a technical infrastructure. Second, the immense scale and diversity of interactions across communities on the synchronous platform necessitates automated tools in community management [35]. Discord thereby offers a setting where the effects of bots are tangibly felt, which helps us surface the stakes of bot adoption. Relatedly, Discord provides an ideal setting to understand bot adoption in communities as it offers an API that is used widely to develop and implement bots and produces multiple resources about bot development for its users. Some bots are present in millions of Discord servers,¹ and Warren [69] noted that “more than 30 percent of Discord servers now use bots.”

¹For example, bots such as Dank Memer, Mudae, and OwO, according to bot usage metrics on the top.gg site.

3.2 Data collection

Between January–July 2022, we conducted 16 semi-structured interviews with current or former Discord server leaders who had adopted bots on their communities. We pursued statistically non-representative sampling [67] and intentionally sought diversity in our interview pool along dimensions of location, account age, and the bots used. We recruited participants in three ways: from two servers where community leaders hang out, from servers dedicated to certain bots, and by directly messaging community leaders who were using certain bots. To thank participants for their time, we offered participants an Amazon e-gift card valued at 20 USD. Interviews were on average 56 minutes and took place over video calls. Twelve interviews were conducted by the first author, and four were conducted by the third author.

The protocol guiding our conversations focused on: (1) an overview of the bot and relevant community to *contextualize* the bot adoption process; (2) the *story* of how participants adopted the bot; (3) how participants' impression and use of the bot changed since initial adoption to elicit *reflection* on the adoption process. Taking advantage of the strengths of qualitative approaches in articulating phenomena [11], we focused on discussing detailed descriptions of the adoption process with participants to articulate less visible forms of work powering the successful management of online communities. Across interviews, this protocol remained relatively consistent in focusing on contextualization, the bot adoption story, and reflection. However, as described in 3.4, earlier interviews helped refine the questions we asked about the story of the bot adoption process, particularly between the two rounds of interviews. To contextualize our data, we also asked about participants' background with bots and Discord.

3.3 Description of interview pool

The description of our interview pool is summarized in Table 1. Age ranged between 18 and 31 years old at the time of interview. The participants included nine men, six women, and one non-binary individual. While most participants were in the United States, some were in South and Southeast Asia, the Middle East, and Western Europe.

Participants reported a wide range of Discord experiences (5-85 months on the platform at the time of interview) and technical expertise. We also asked participants generally about the community the bot was used for as that could have bearing on why a bot might be added. Because participants were (or had been) on multiple servers, we did not discuss the details of every server the participant was on. In some cases, participants only described the community in brief and in the abstract. However, from the server contexts we heard, our pool captured experiences on diverse kinds of communities including those for learning, hobbies, YouTubers, friends, companies, NFTs, university sports teams, and games. A summary of select server community topics can be seen in Table 1. We note that while most of the communities discussed by participants were relatively large, with thousands of members, two participants (P2 and P8) also mentioned smaller communities with about 50 members. The largest communities mentioned, with hundreds of thousands of members, tended to be those that were officially affiliated with games or companies. Overall, participants were leaders of communities across different sizes although the adoption of bots seemed to occur more frequently in larger communities, which is supported by past work from Kiene and Hill [35].

Finally, we summarize key bots substantively discussed with interviewees in Table 2 in the Appendix. While several bots came up repeatedly across interviews, the semi-structured conversations also left room for discussion of multiple bots. Our interviews therefore capture *multiple* experiences of particular bots as well as experiences across *many* bots, including some custom bots. Our focus here was on bots that facilitate various aspects of community management (e.g., moderation, governance, newcomer onboarding, etc.). However, participants frequently also mentioned bots

P#	Age (years)	Gender identity	Geography	Account age (months)	Technical experience	Server topics
P1	28	M	Southern USA	66	Medium	Gaming, Blockchain
P2	20	F	Midwest USA	13	Low	Fandom
P3	18	M	Southern USA	54	High	Bot
P4	19	M	East Coast USA	48	High	Company
P5	19	M	Southern USA	48	N/A	YouTuber, Gaming
P6	24	M	United Arab Emirates	72	Medium	University
P7	22	M	Southern USA	70	Low	Personal
P8	19	F	Malaysia	48	Low	Personal
P9	21	NB	Southwest Germany	62	High	Bot, Coding
P10	20	M	Southwest Germany	69	High	Personal
P11	20	M	Central India	60	N/A	Ads, Content creators
P12	23	M	Thailand	5	Low	Gaming
P13	27	F	Southwest France	77	High	Language learning
P14	26	F	East Coast USA	86	High	Music bot, personal
P15	31	F	West Coast USA	27	Low	Homestead
P16	22	F	Southern USA	69	Low	Gaming

Table 1. Summary information about participants. Some details have been obfuscated to protect participant privacy. “Technical experience” refers to technical knowledge noted by the participant, with “N/A” if the participant did not give any information to this end.

used for other tasks like entertainment for the community, which we include in Table 2. Some bots included “premium” features for a subscription fee.

3.4 Qualitative analysis

Following the grounded theory approach in Charmaz [9], our analysis involved line-by-line inductive coding, iterating as needed, and then memoing with initial themes, patterns, and directions. Our interviews were conducted in two rounds: the first in January–February, 2022 (10 interviews) and the second in May–July 2022 (6 interviews). Initial coding and memoing of this first round of interviews revealed parts of the protocol described earlier in §3.2 that required follow up. We then pursued the second round of interviewing, using a revised protocol focused on those parts of the bot adoption story that remained unclear from the first round. After this second round, we conducted initial coding on the new interviews. We determined data saturation by a heuristic of redundancy in content, themes, and codes that suggests replicability [11, 16], keeping in mind Braun and Clarke [4]’s observation that saturation “cannot be determined (wholly) in advance of analysis.” Finally, we conducted a new round of coding and memoing on all interviews to synthesize across them resulting in over 3000 initial codes.

The first author organized the initial codes into focused codes to collapse repetitive codes and identify the most relevant ones. Organizing the focused codes into a map of clustered, related codes,

the first author then developed axial codes of higher-level themes. From this thematic map, we generated additional memos. Through iterative discussions around the axial code map and memos, we refined a set of findings until reaching agreement.

4 FINDINGS

We highlight three key themes identified in our interviews. The first theme (§4.1) focuses on challenges around bots as tools whose third-party nature heightened uncertainties about the costs and considerations of adopting them. The second (§4.2) elaborates how an emergent user-driven network of social and informational resources spanning across communities has helped address these challenges, and the third (§4.3) centers on how the social nature of bot adoption shaped users' technological frames of bots across Discord communities.

4.1 Challenges of adopting third-party bots

As participants sought to adopt bots, they described core challenges around the security, reliability, and fit of the bot. These challenges captured the uncertainties of adopting a tool made by someone else, which often created ambiguities about what one could expect or assume about it. Challenges were therefore not just about configuring bots but also about the costs and considerations in adequately assessing the nature, value, and potential use [49] in order to adopt a tool that fit with the community's needs and goals.

4.1.1 Security and reliability concerns in layers of dependencies. Adopting a bot (i.e., “inviting” it to the server) entailed granting it administrator permissions to the server, like managing chat channels and chat content as well as kicking and banning users. This meant that once added, a bot could effectively “take over [a] server” (P2) without much recourse, and participants recurrently described third-party bots as potential security risks that could “break their servers” (P14). However, first-hand experiences of malicious bot developers were not especially common. Instead third-party bots more often posed potential security risks because they were developed and maintained by fellow Discord users who may be vulnerable to bad actors or simply unreliable, which could be harmful to communities. P13 remarked that they had “a lot of trust issues when it [came] to bots” because they had heard about poor security practices of developer teams, even those of big bots that everyone seemed to trust. For example, in some cases, developer teams accidentally leaked their bot *token* (i.e., a key that enables access to and control of a bot) in the wild, which enabled malicious users to use it to take over a bot and destroy servers.

These concerns were heightened because participants nevertheless saw bots as core to their work of effective and safe community management. For example, after a developer lost the data powering the bot (e.g., records of who adopted the bot and what their settings were), P16 found that their work to configure a moderation bot in their community had been lost. While seemingly lower-stakes than a bot takeover, this could be consequential as bots like the moderation bot of P16 functioned as security mechanisms on servers, catching scam and virus links, producing activity logs supporting moderation work, and filtering hateful language. Bots could also go down due to reasons outside of the control of developers, such as when Discord updated its API and disrupted the functionality of bots, or if infrastructure such as hosting servers went down. When the services of a bot were disrupted, communities became vulnerable to risks and doubly took on the costs of failure. In light of this concern, participants described adding multiple bots that had similar functions as back-up tools.

The unpredictable layers of dependencies that third-party bots evoke contrast with the sense of certainty and control afforded by custom bots (as described by participants such as P9, P10, and P15). However, control also came with a sense of responsibility: “When you have ownership over

something, you truly have ownership over the problems” (P15). Whether communities wanted to take on this ownership over such problems was not always clear. For example, P4 recalled two different communities they were involved in that took very different stances: in one case, the community “didn’t want the developer to be in control whether the bot was on or off [and wanted] the security of hosting it themselves”; in another, the community preferred to “trust the developer to keep the bot online because otherwise that [would be] more responsibility” on themselves. Thus, the uncertainties of third-party bots as external tools could be both a vulnerability and a boon: while they imposed distinct concerns, they also enabled individuals to distribute the responsibility of bot-related security practices.

4.1.2 Barriers to selecting the “right” bot. Participants needed to decide whether a bot would be an appropriate choice as a technology to support their needs and processes [48]. Interviews with participants revealed an enormous number of third-party bots one could adopt, many of which overlapped in functions. While participants generally liked having a wide range of choices, this also meant that knowing what bots were available and which were good fits was not always straightforward. Sometimes, participants had not initially realized that bots were capable of doing certain kinds of tasks. In particular, while many bots that directly interacted with community members were recognizable to even newcomers, bots that supported the critical back-end work of community management were less visible, which made anticipating their potential value and use much more difficult.

Participants described needing to select from many options of bots that essentially performed the same tasks, which further complicated this process. Some factors such as cost, when bot developers sometimes paywalled more “premium” features, were straightforward reasons to adopt or not adopt a bot. However, other important factors — such as whether the bot developer was reputable, the bot worked well, or the bot was easy to use — required more in-depth knowledge and context that couldn’t be gleaned from a description of a bot alone. Participants also cared about what they saw as the social values of the bot developer, such as P10 who got rid of a bot after realizing the bot developers supported Non-Fungible Tokens. More often, however, differences between bots that offered similar functions were fine-grained in expectations about the user experience, which required going actually configuring and using the bot to discern. For example, P5 distinguished two moderation bots by how they allowed users to add words they wanted to filter, with one using a simple graphic user interface where users would input and delete words one at a time like “building blocks” and the other using an open text box for users to enter words separated by commas. While a subtle difference, the bot with the open text box offered more flexibility, enabling mass insertions and deletions of words to make configuring filters less tedious, and therefore preferable to P5.

Given the testing and research needed to scope out mental models of how particular bots worked, the selection process of adopting a third-party could be both time- and labor-intensive. Of course, participants sometimes found the process to be relatively smooth, simply adding the first bot they found on a search engine. However, these bots tended to be popular, multipurpose bots that favored a breadth of functions over depth, and were not always a good fit for what a community needed (possibly making community management less efficient).

On the other hand, it could take a considerable amount of work to find better alternatives, not only due to the subtleties of bots described in the prior paragraph but also due to the simple issue of needing to know potential alternatives even existed. Even after selection, it could take even more time to properly set up and configure a bot, especially for more powerful bots which had steeper learning curves and in turn imposed knowledge barriers. P4 recalled that although “the instructions list [of the bot] made it sound like set-up would take an hour, for me, it took the entire week to get [the bot] up and running”; P3 mentioned spending “a couple days [...] over a weekend [...] just like

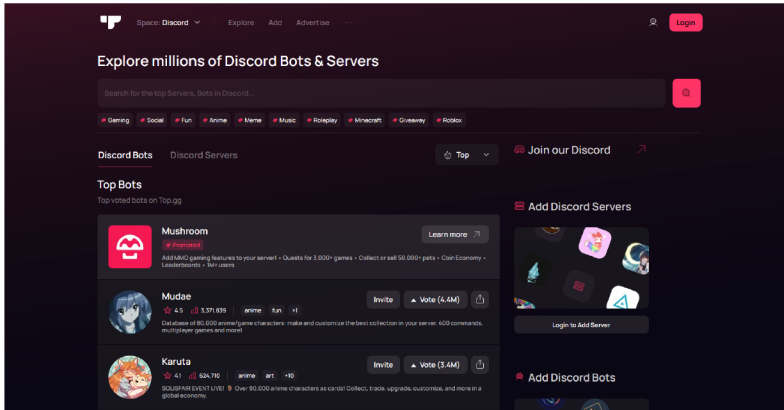


Fig. 1. A screenshot of top.gg, a website that lists and ranks Discord bots mentioned as a useful resource for finding and assessing bots by interview participants.

slowly building [the bot’s configurations] piece by piece and testing it,” while P11 reported taking a whole month to do so. Across multiple bots, this all added up. P12 estimated they had spent up to 600 cumulative hours adopting bots to communities. They explained were no longer very involved with bots (and community leadership roles more generally) because they did not have time for it.

4.2 Leveraging social ecosystems around bots

Despite challenges, participants not only adopted bots but also adopted many of them. In this section, we describe how social ecosystems around bots — a user-to-user network of knowledge, resources, support, and social connections spanning across communities — shaped participants’ understandings of these tools. By creating opportunities for social and organizational learning, these ecosystems helped participants address the challenges of third-party bot adoption and made the work of bot adoption more fun and worthwhile.

4.2.1 Reputation in rankings and word-of-mouth discussions. The most common way participants described learning of and about bots was through other users who had experience with the bot. Because bots deployed in a community were typically visible in the community’s member list as special types of users, participants could identify particular communities to ask about specific bots (in addition to friends or acquaintances in their network). For example, P4 remembered reaching out to another community’s moderation team after being especially impressed with a moderation bot that was efficiently catching spam, including “all sorts of modifications that spammers would make, like replacing an O with a 0, that type of stuff.”

The apparently successful use of a bot elsewhere provided a signal not only of what it could do but also of its value, wherein participants reasoned that since others “[had] already used the bot, it must be working” (P7). How widely adopted a bot was was often a basic signal of reputability that participants leaned on. Participants frequently mentioned websites such as top.gg,² which list and aggregate information about Discord bots as a centralized resource with metrics (see Figure 1) like how many people have added the bot to their community.³ Such websites helped participants find

²<https://top.gg/>

³At the time of this writing but after our data was collected, Discord implemented an App Directory system that integrates bot discovery into Discord’s user interface. The system has core similarities to widely-used ranking sites such as top.gg noted in our interviews, organizing bots into categories and offering an overview of a bot’s features, metrics, and links.

and choose bots by aggregating and sharing user feedback on bots in the form of votes, ratings, and reviews. Because these websites tend to sort all their lists of bots by the number of votes received, potential bot adopters see the bots with the most votes first by default.

However, given the concerns described in §4.1, reputation was more than just numbers. A highly valued technology for one group may be seen differently by another [50], and participants sought more nuanced information to understand a bot's value and use for their community. This search was facilitated by communities which served as hubs for moderators, managers, developers, and other community leaders to gather, network, and chat about the work of maintaining and improving their respective communities. The most prominent such hub was one officially run by Discord called *Discord Moderator Discord* (DMD), but participants also pointed to other hubs that had organically emerged. P1 claimed that these hubs were often the first place people went if they were looking for a bot, and participants mentioned finding bots in these hubs both by lurking (P14) and by directly asking for recommendations (P1, P3).

An important aspect of word-of-mouth discussions on hubs (vs. asking a random community about their bots) was that members of the hub began to get to know one another as they discussed several dimensions of community management in a space dedicated to fostering successful communities. As they did so, participants could solicit trustworthy recommendations from people beyond their existing personal network, search discussions about specific bots archived in the conversational logs, and even receive unsolicited recommendations for bots that others thought they would like. Meanwhile, bot developers who were in the hub also developed a reputation for being reliable, detail-oriented, careful, or engaged — all translating to the reputation of the bot. The dynamics of reputation, reliability, and trust apparent in intercommunity hubs underscore the social nature of bot adoption apparent across our interviews, particularly to address the challenges tied to third-party tools in §4.1. P14 most explicitly surfaced this in recalling how they adopted a bot not necessarily because its functions were clearly superior to alternatives but because they knew that “the developer of the bot was someone very trustworthy in the broader Discord community. I don't want to bring popularity into this, but this person is very well-known in the moderation community and developer community.”

4.2.2 A sense of care in crowdsourced “customer” support. When learning about and setting up a bot, participants reviewed the written guides of bots online, generally in the form of developer documentation. However, documentation across bots appeared to be inconsistent in quality, confusing, or limited in what it covered, leaving the nature of the bot overall unclear to new users. In some cases, documentation was extremely sparse or difficult to find at all. As a result, participants had to construct knowledge about bots on their own and often described turning to other users (e.g., on hubs) or experimenting with the bot. In particular, interviews highlighted the significance of a bot's *support server*, or a Discord community centered around the bot, in answering questions and providing support. Typically owned by the bot developer, these support servers provided access to a developer or their support team for any bot users with questions. On their end, some developers also gave regular updates on the support server or responded to inquiries about bugs, which shaped participants' sense of whether the bot was well-maintained.

Support servers were especially useful because they enabled access to other bot users, not just developers, for “customer support” (a phrase used by P11) in using a bot — especially power users who were more knowledgeable, technically-savvy, and had more experience with it. Having a community of responsive and helpful fellow users was often crucial in helping participants set up, configure, and troubleshoot bots properly, as there were far fewer developers than users of a bot to field inquiries. Having used the bot before, other users knew how to “tweak the bot” (P3) and could share how to “apply [tips] in your own server” (P12). These tips and tricks that came from

experience were especially helpful in implementing bots that involved more elaborate settings which could resemble code. Additionally, they responded to users' particular requests. For example, P11 felt confident that going to the Dyno bot support server would yield a rapid and personalized response to help them troubleshoot something from other users:

If I'm changing a setting and it doesn't work, I'm like, *What the hell happened?* I go to the support server and say *I was doing this, this, and this. And then this happened.* And I will attach my screenshots, like, what is the error that's coming up? What is the output? [...] I can tell them the whole situation. And I'll get a reply within like 10 minutes; that time is like, the max point. Sometimes after just 10 to 15 seconds, someone posts a response in the support server.

In short, support servers enabled learning without needing to rely on the quality or capacity for communication from the bot developer. A sentiment of shared care pervaded many of the statements around support servers. Participants like P3 described how they felt other users cared when helping P3 out during a stressful moment in troubleshooting, with some members of the support server even offering to join a participant's server and fix an issue for them. Such a sense of care cultivated the feeling that there was a community around the bot that was ready to help and jump in to help make their community successful. In contrast, support servers where users felt developers no longer cared and where user-to-user support was not allowed were disappointing, even becoming a reason for participants to switch to a different bot.

However, a sense of community did not supersede the support function of the server; participants were still primarily there to better understand how and why they might use a bot to address community management issues. P13 suggested that centering the community aspect of a support server could be counterproductive to what made a support server feel caring, by burying questions and advice. This could also dilute another direct benefit of support servers, which was by serving as a searchable archive of user discussions that people could reference. Some support servers even deliberately collated user-made guides that were, as P7 recalled with surprise, much more detailed than the documentation or official information about the bot. While talking about learning how to configure a YAML markdown file of a more complex bot, P3 described how the support server had "a whole category of channels that just example configurations, that you can copy and paste, and it does the work for you." In this way, the sense of care of support servers was directly tied to its value as an informational resource for participants to better assess the nature, value, and use of potential bots to adopt.

4.2.3 Members of the community as recommenders and troubleshooters. Participants also recalled turning inwards to their own community members, who had varying degrees of familiarity with bots themselves. By doing so, participants could make decisions about bots that would be appropriate within the particular context of their community. Community members not only served as *recommenders*, but also as *troubleshooters*. For recommending, participants noted they adopted bots proactively recommended by or solicited from community members, sometimes as direct attempts to improve current community management processes. One participant even had a semi-formal suggestions channel for bots, where people could vote for bots to be added.

For troubleshooting, participants described reaching out to others to ensure that a bot was functioning as intended, after initially doing some testing themselves. While P3 mentioned asking a friend to help test a bot, more often participants asked community moderators or staff as well as regular community members. P2 felt that doing so gave them more insight into how the bot impacted the different kinds of users with varying levels of permissions and access in their community:

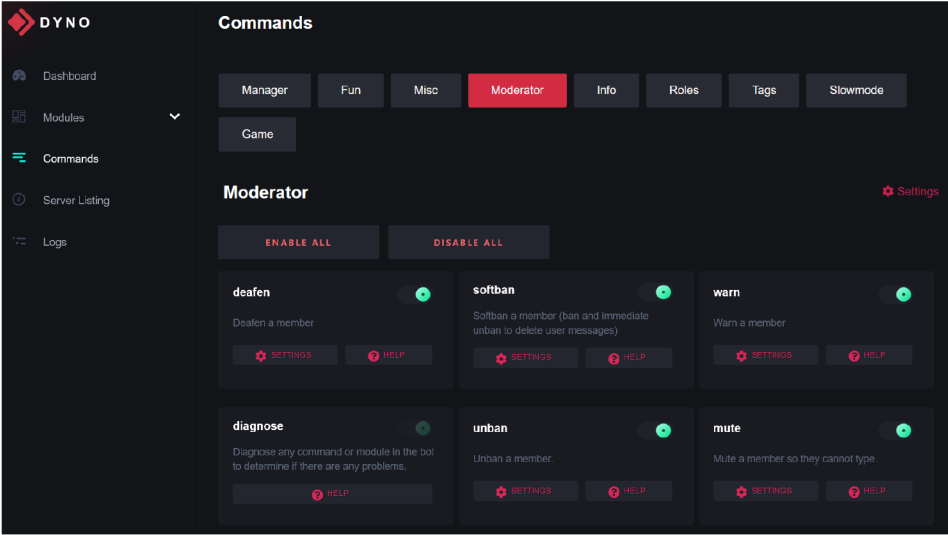
As the owner, I have all the perms. So it's nice to be able to talk to people with less perms and see if their experience is affected [...] I try to ask the moderators because they also have a lot of perms. I try to ask people who have kind of just the base, normal perms - so usually a super active member of the server who's down to do anything, and they're like: *Oh yeah, I'll help you with the bot, sure.*

Getting input from regular community members could be very valuable for catching things that community moderators or staff didn't notice. For example, P5 recalled how community members not only suggested bots but also suggested turning certain features of bots on and off based on what they were seeing on their end.

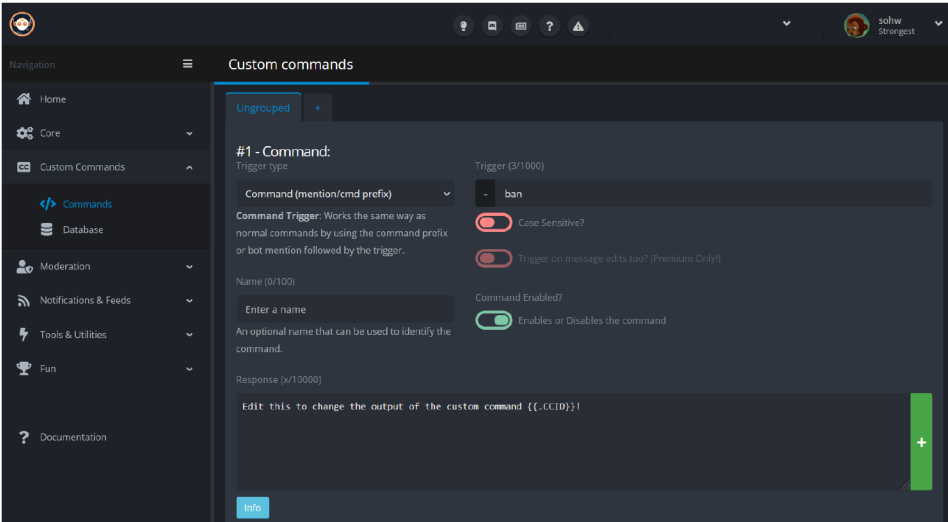
To effectively troubleshoot the setup of bots with community members before officially rolling them out, participants employed distinct strategies, most notably including homebrewed "sandboxes." In particular, participants had private "dummy servers", or servers that they had made to add and test bots over time (sometimes, as a blank copy of the actual community they planned to add a bot to), or private channels on their existing community specifically used for testing bots. In these "sandboxes", participants would invite a few community members they trusted and test functions of the bot with them to "try to make sure that everything's working [...] so we wouldn't break anything" (P7). For bots that helped manage community engagement by interacting directly with the community, this included having community members test whether the bot would respond as desired, such as sharing memes, starting mini-games, or responding to member queries. For bots that helped manage administrative or moderation aspects of the community, this included having community members write messages that should be moderated to check if the bot would appropriately filter and remove such messages and record the apparent rule violation in back-end logs. If an issue was apparent, the problem would be constrained to the dummy server or the specific channel until fixed, and the actual community would continue to smoothly run in an otherwise potentially-disruptive moment.

The involvement of community members in recommending and testing bots appeared to encourage social bonds and trust between members and leaders. Participants felt it was important to make sure community members' wishes for certain bots to be added were respected (especially for bots that interfaced with community members), at times overriding admins who felt some community-proposed bots were not useful. Recalling discussing which bots to add with other members, P8 explained how "brainstorming and coming up with the idea of what we wanted this community to be was very enjoyable." At the same time, bot adoption did not consistently engage nor prioritize community members. Community leaders also stated that it was also not clear that it was always better to. Being too obvious that a bot was being adopted could be undesirable, like if the bot was meant to address issues with scammers and spammers, in case it gave bad actors insight into the community's moderation strategies. Moreover, community leaders ultimately retained executive decision-making power. For example, P9 felt community members should make a convincing case for a bot if they wanted it added: "If a user in one of my community suggests a bot, I expect them to fight a bit about why I should add the bot and what features it has. And if they cannot supply that, then I won't look into it. If they can supply both of those to a reasonable extent, then I will go check the homepage of the bot."

4.2.4 Moving towards more complex tools. Participants often described a trajectory towards using more complex, technical tools as their expectations and assumptions about bots evolved. For example, P16 described starting off with a popular beginner bot to moderate their community before switching to a bot with more advanced and efficient configurations for language filtering. Although P16 had no coding background, their prior experience eased their transition to the more advanced bot: "It was basically using intuition [...] like okay, so this how this [bot] reacted, maybe



(a) Screenshot of configuration panel of Dyno



(b) Screenshot of configuration panel of YAGPDB

Fig. 2. Examples of two configuration panels from the bot Dyno (a), considered a popular, beginner-friendly bot, and YAGPDB (b), a more advanced bot. While configuring Dyno involves enabling commands and adjusting their settings from a GUI dashboard, YAGPDB also enables the creation of encoding custom commands that respond to specific user-defined triggers.

this one will react this way [as well].” Said otherwise, participants described how they could naturally become adept bot users by honing relevant skills and intuitions because there was “a specific way you use bots. And I think once you figure that out, every single bot is pretty easy to use, because they all work the same way” (P10).

Most participants were active as admins or moderators on multiple communities and often drew on their hands-on experience of using a bot as a moderator in other communities, a kind of soft

launch into using the bot while working with more experienced moderators. Having the opportunity to actually try using a bot in the everyday work of community management was especially valuable for bots that involved more back-end configuration. Being a moderator somewhere directly enabled this because one had to learn the tools a community was using to contribute as a moderator. Although getting to know how to use a bot was in some ways a side effect of the moderation work, positive experiences could result in such favorable impressions of bots that participants felt certain they would use the bots elsewhere. While they had no technical background themselves, P12 was able to try using an advanced bot while they served as a volunteer moderator on a big community. Their positive experience using the bot within the moderation team led them to adopt it in their own community later on, using the same configuration settings and strategies.

Drawing from hubs, support servers, and the communities they were involved with, participants were able to build intuitions about how bots worked, learn about better alternatives, and advance to more powerful tools. For participants like P14, because it involved brainstorming with community members or asking others for advice about bots, this process of learning in bot adoption could even feel less like work and more like something enjoyable to do. For others, such enjoyment was in part derived because the process of bot adoption was tied to working together with a team. Ultimately, the result was that participants with no technical background such as P8 and P12 were still able to use bots considered to be for power users effectively. Meanwhile, a sense of expertise was made concrete in the configuration files of more complex bots that required some basic knowledge of programming, necessitated understanding how technologies like bots work, and resembled pseudo-code. Participants noted that feelings of satisfaction and growth from developing bot-specific skills often helped overcome moments of frustration:

It's kind of like solving a Rubik's Cube, where sometimes you don't remember the instructions in the manual, and then you keep trying, and then it works. Is it frustrating to learn? Yes. Of course — when it's still not able to do [what you want]. Once you get it, though, it motivates you to keep doing it. [And] then the next time the frustration comes, it pulls you through. (P12)

4.3 Strategies and limitations of customizing with third-party bots

§4.2 showed how individuals leveraged social ecosystems to address uncertainties about bots and revise their understandings of the nature, value, and use of bots as community tools. In this section, we describe how the social nature of bot adoption both supported and constrained creative use of third-party bots. The open sharing of bots on Discord allowed for the adoption of a diverse selection of bots, each with features that could be further tailored and “stacked” (P1, as discussed below) to meet the specific preferences of communities. However, interviews also suggest that this sociality may also stifle innovation as norms around bot functionality and usage calcify.

4.3.1 Creative license in stacking bots. Throughout our interviews, participants repeatedly described adopting *multiple* bots, devising what P1 called a “bot stack:” a set of bots that, together, offer users what they want and need in a combination of their functions. In place of creating custom all-in-one bots, a sense that one was customizing the community with third-party bots was facilitated by two creative licenses participants retained by stacking bots.

First, participants were able to cherry-pick functions across bots, curating a set of functions that they wanted to have in their community through in their bot stack — a non-trivial task, as suggested by the challenges of adopting even one bot described in §4.1. Second, participants often found that many bots offered varying degrees of customizability in their settings, with some being highly specific in the particulars of how a function worked: “You could decide, like alright: let them know why they were warned, why they were muted, why they were banned, or whatever” (P16).

Both the effort of curation and the customizability of many bots ultimately offered participants a sense of control over how third-party bots behaved, echoing sentiments about how custom bots could be made to “exactly look the way I want and work the way I want” (P10). In short, while a single bot may only partly align with an individual’s uses, participants could fulfill their needs and expectations for what bots could do for their communities through the curation of many.

The curation of multiple third-party bots became a lens through which communities built the unique community space they envisioned. This was most obvious when participants talked about wanting to “stand out” with interesting bots that would engage users and foster a lively community (P2), or even when participants liked bots simply because they offered more customization. However, it also became apparent as participants negotiated the capabilities bots could offer them to appropriately manage the kinds of community members, content, and behaviors that passed through their community spaces. One participant (P7) explained how they wanted a bot to verify users’ age so that they could comfortably have conversations without worrying about minors being in their community. In another case, P4 focused on adding a bot to their stack to easily filter out all swear words and slurs, because the community they were adopting the bot for (as a paid intern) was owned by a company focused on high school education resources.

Participants were conscientious and proactive about thinking about how they could “make the server into what [they] want[ed] it be” (P8), using other other communities as reference for thinking about their own and actively discussing what would be best in hubs, support servers, and their communities. In part, this was because bots could not simply be added endlessly into an ever-growing stack: some bots could “break” one another by accident if they had similar features, and having too many bots could complicate the load of managing them overall. The mental load of dealing with configurations across bots meant that participants had to be relatively thoughtful about what went in the stack. To this end, P5 and P12 both recommended deliberately looking through other servers to “see maybe what assortments of bots that different servers with different needs will use” (P5), a reflective strategy that they had frequently used in the past as a jumping off point to identify potential useful bots as mentioned earlier in §4.2.3. This facilitated discovery of interesting tools and functions, beyond those a participant was already aware of. Once a bot was identified as a potential tool of interest, one could then look to other communities as case studies of how the bot was being used before deciding whether and how to implement it.

4.3.2 Constraints of social ecosystems around bots. As participants sought to customize community spaces with bots, their choices remained shaped by the prevailing technological frames about bots within the social ecosystem. This became clear in three key ways. First, the very practice of bot adoption in of itself occurred because participants now saw bots as integral to Discord, given that bots were noticeable everywhere. One participant commented that they could not imagine a Discord community running without bots, and another (P4) claimed that some bots were so popular that “it was like one of those things your community manager knew about, when she’s like 28 years old and hasn’t used Discord before.” In other words, the prevalence of bots across communities attributed a widespread sense of value and importance to bots: *that bots existed and should be added*, even from the perspective of newcomers.

Second, participants often described genres of bots without any prompting from the interviewer — such as ‘fun’ bots, ‘moderation’ bots, ‘utility’ bots, ‘logging’ bots, ‘multipurpose’ bots, or ‘music’ bots — when talking about the bots they encountered in other communities and in discussions in hubs. These informal genres defined categories of functions that bots might offer, thus suggesting to participants the nature of bots: *what bots can do and how they work in communities*. Participants who were also developers suggested that these genres were not just random labels, but reflected distinct paradigms of the functionalities that bots offer to communities over time. For example, P13

recalled how after one moderation bot offered a new strategy for configuring moderation actions, several subsequent moderation bots that did the same thing appeared and soon came to dominate the design of moderation bots. These paradigms became apparent to participants as they explored more about bots in hubs and other relevant discussion spaces.

Finally, the social ecosystem informed *what bots people should use for what problems* by signaling which bots were popular, reliable, and safe as described in §4.2. It also told users what bots might make a community more interesting, successful, or engaged. In some cases, this was because it seemed any community worth its salt should have a certain bot. For example, P15 remarked that having certain kinds of moderation-related bots was simply a “common knowledge thing.” Similarly, P7 added a bot that counted up the stats of their community, such as number of active members, because “it became common between a lot of communities to have a bot that displayed that kind of information. So yeah, we were like, yeah, we’ll just go ahead and put it in.” In other words, P7 adopted the bot not because it seemed especially useful or like a good fit for their community, but because it had come to reflect a common characteristic of Discord communities. In other cases, it was because a bot seemed to truly make a community stand out, which a community leader might want to replicate in their own community: “If you join other servers, you can see that there’s something special about that server. So, I saw [the bot adding something special] and then I was like, I should have that in my server too” (P11).

5 DISCUSSION

Our interviews showed how participants addressed heightened uncertainties about third-party bots through social ecosystems that offered participants many formal and informal opportunities to revise their frames [49] about bots, both supporting and constraining how bots became understood and used. Below, we discuss the implications of these social ecosystems in enabling individuals to develop bot-specific knowledge and skills; shifting perceptions of the labor of bot adoption into something satisfying and fun; and structuring individuals’ sense of the value and roles of bots across communities with diverse needs and goals.

5.1 Closing the gaps of bot-specific knowledge and skills

Across interviews, people with diverse technical backgrounds were able to adopt bots that ranged in complexity, including individuals without coding experience successfully adopting bots that involved configurations in markdown languages that might normally pose knowledge barriers. This was surprising as prior work has emphasized that misalignments between bot users and developers produce misunderstandings about bot capabilities and roles [42] as well as constrain who can use them [21, 31, 59].

Bots in our study did pose potential barriers that could produce such misalignments, with uncertainties about the nature, value, and use [49] of third-party bots. The stakes of addressing such uncertainties were heightened by the fact that bots served important security and safety functions. However, our interviews showed how formal and informal networks of social support, resource sharing, and discussion among users helped close knowledge gaps so that bots could be effective and useful for a broad range of potential bot users. Orlikowski [48] notes how *Lotus Notes* ultimately failed because users did not understand the collaborative nature of the software. In our interviews, we observed that social and organizational learning about the nature, value, and use of bots was facilitated by a variety of means. Participants could gain insights about bots by looking at what other communities used, seeing recommendations from community members, and both lurking and asking in hubs. Participants also often turned to support servers for the bot, where other users offered tips, guides, and support, and developers could be easily reached. Such support servers echo earlier work which found that discussion spaces for software customizations were

valuable for customization sharers and users alike [26]. Similar opportunities for engagement across communities additionally mattered for figuring out what kinds of bots existed, how particular bots worked in comparison with another, and what to do when problems arose. This was especially true for bots that support key aspects of community management, which are not always visible from the outside, rendering nuances of how third-party bots actually operate obscure unless existing documentation is extremely thorough. Ultimately, through these diverse social resources, participants were able to build bot-specific knowledge and skills as they revised their technological frames (assumptions, expectations, and knowledge) [49] about bots over time, often shifting to a more sophisticated curation of various “advanced” bots regardless of technical background.

Our findings show that the diffusion of end-user tools on social platforms should be matched with efforts to provide social scaffoldings that support exploration, learning, and user discussion of those tools. Bots and other end-user customizations enabled via APIs offer valuable opportunities to tailor sociotechnical environments [18, 19, 35, 52, 61, 64, 75]. However, not all users are able to create custom bots, and sharing tools across users can be extremely fruitful [31]. We observed that sharing of tools involved challenges not only around the effort configuring a bot but also around the work of assessing it. Our findings highlighted that these challenges can be addressed when participants can communicate and coordinate across communities to understand what tools can do, clarify how they work in practice, and learn how to set up and troubleshoot them.

On Discord, bot support servers were common and the platform itself hosted an official hub for community leaders (although other hubs existed as well). However, much of the social interaction around bots was also emergent and informal. For example, simply seeing the tools other communities used raised an occasion for participants to reach out. Participants serendipitously came across bots as a part of being communities that they enjoyed. While platforms may benefit from deliberately cultivating intercommunity interaction, we caution against too many top-down strategies to support interaction around tools, which may overengineer how people approach these user-made tools or peoples’ sense of community in intercommunity spaces. Instead, facilitating bottom-up, user-to-user interaction and support across communities is likely to cultivate the benefits observed in this study.

5.2 Bot adoption as collaborative, communal labor

Conversations with community leaders highlighted how regular community members took on supporting roles that aided them, recasting bot adoption in a more collaborative and communal light. While prior work finds that the adoption of tools is done by a small subset of skilled users [21, 31, 42], our work extends these observations by showing how regular community members also support those users: bringing effective tools to the attention of community leaders, using their differential administrative permissions to help more thoroughly test bot behaviors with community leaders, and at times even offering suggestions for fixing bot settings.

One implication of these results is that affordances which facilitate wider participation from different kinds of community members can enhance and improve the process of tool adoption. Community members were ostensibly users whose community experiences were impacted by the bots, but also offered valuable insights as technologists. As incongruent views between groups about an adopted technology can undermine its successful deployment [49], our interviews show how sharing the labor of technology adoption can foster opportunities for groups to sync with and learn from each others’ differing perspectives. This can soften any disruptions that come from adopting a new technology. However, a potential concern with having a community at large more involved is whether the subsequent visibility of management strategies associated with a tool (especially for user verification and moderation-related tasks) will make them easier for bad actors to circumvent [6, 22, 31]. Broader community participation in tool adoption should be constrained in who is brought into the fold, how long, and to what extent, in controlled environments. Participants

selectively brought trusted community members into the fold, inviting members *temporarily* to homebrewed sandboxes (i.e., dummy servers) or dedicated channels.

Sharing the labor of bot adoption may also engage communities more deeply by enabling new ways of participating in the community. Bots were a part of shaping community identity, and community members often advocated for bots that would enable them to have the kinds of interactions they wanted. Meanwhile, community members could now see the normally hidden back-end of community management. The value of a bot may not be only in how it can automate tedious tasks at scale [23] or enhance the sense of community with engagement strategies [61], but also in how it becomes an artifact around which communities collaborate and deliberate upon, cultivating engagement in of itself. These collaborative and communal dimensions seemed to shift the way participants perceived the labor that goes into adopting bots. Participants in our interviews tended to emphasize how they saw the process as fun, satisfying, and part of the community-building process, even while acknowledging the frustrations, costs, and challenges of the bot adoption that evoke ongoing questions about how platform companies profit from the invisible labor of communities [39, 40]. As participants talked about finding and sharing support and useful information in support servers and hubs with other community leaders, these frustrations, costs, and challenges were recast in a positive light by putting the focus on what participants learned and were able to now do.

On one hand, this suggests that various social benefits — such as personal growth, learning, relationships, and interpersonal interactions — from the bot adoption process itself can mitigate the negative tensions around the labor involved. By cultivating social relations around the work of its adoption, bots offer value beyond their functional roles. On the other hand, such a perspective may be misconstrued to diminish the human labor that powers sociotechnical infrastructures like bots [15, 20, 38], especially the labor of those who are not designers or developers [30]. We caution against presuming the benefits gained by users during bot adoption signify gains that are an equivalent exchange for the value of the labor of bot adoption. Rather, we hope future work examines how the different kinds of value that emerge in the work of community leaders interact to shape outcomes like the emotional care and costs of community management [13, 73] or engagement and retention of members of community moderation teams [56].

5.3 Shaping assumptions, expectations, and knowledge about third-party tools

Our findings described how the social nature of bot adoption guided the evolution of participants' technological frames [48, 49] (their assumptions, expectations, and knowledge) about the nature, value, and use of bots: i.e., what bots are and can do, why certain bots are worth adopting, and how the bot can be and is used. While participants sought to use bots to create unique, stand-out communities by stacking bots, we highlight the implications of leaders converging on similar technological frames when trying to effectively manage their communities via shared tools.

Spaces of informal, community-based learning like those we observe in our work can often produce archetypes about concepts that shape subsequent learning and exploration [10]. Interviews with participants pointed to ways that peoples' understandings narrowed from engaging in social ecosystems around bots [50]. This may have undesirable side effects. For example, one participant noted that the popularity of a bot developer played a role in selecting a bot. While we can reason that the developer may be popular because they have been responsive and has maintained their bots well, popularity among community leaders can also be built in ways that do not correspond with the value of a bot. The question of popularity surfaced when participants talked about initially using widely-used bots that they realized later were subpar, eventually replacing them. Similarly, participants added bots that gave them affordances to look like other communities, even if they did not find those affordances particularly compelling.

The constraints of technological frames can be generative when coupled with capacity for user innovation [36]. However, adoption of *third-party* bots means that user innovation lies in the creative liberties participants take in bot stacking, rather than creating add-on scripts or extensions. Concerns that these influenced choices are limiting or non-optimal are part of the question of how to curate a “good” stack of bots (or other user-made and -facing tools). Participants in our interviews took advantage of what they could do with bots through bot stacking, and it was clear that they were often deliberate and thoughtful about what bots to add. As more user-facing tools for social platforms become available [e.g., 32, 55, 74], we note that understanding how users might recognize or overcome the constraints of existing tools will be critical. To this end, the practice of bot stacking (and similar curation of sets of user tools) is a rich area for future work. Closer examination of these practices could have concrete design implications for improving the design of user tools in general, especially by considering the technological frames made salient by bot stacks. For example, as participants cherry-pick functions across bots, understanding how resulting stacks diverge in nature, value, or use from already identified “roles” bots play in communities [58, 75] can generate new ideas for valuable tools that support community management.

More broadly, the importance of social ecosystems in how tools of community management are adopted underscores the significance of intercommunity dynamics between community leaders, or how community leaders across communities positioned themselves and the management of their communities in relation to one another. These dynamics guided decision-making during bot adoption: community leaders across communities directly discussed bots and community management together, searched through each others’ communities for inspiration, served as mods and admins in other communities, and ultimately used shared tools that constrained them in similar ways. However, these intercommunity interactions may also create space for novel ideas and uses to emerge through discussion and reflection, i.e., for new frames about bots to be reflexively developed. We suggest that further attention to interactions among community owners, moderators, bot developers, and leaders across communities will help us explain the assumptions, expectations, and assumptions around third-party tools in managing online communities. In doing so, we build on a growing body of work examining the interdependence of online communities [63, 65, 66, 70], such as the relational dimensions of how communities are managed [8, 29].

6 LIMITATIONS AND FUTURE WORK

The findings of this study are limited by the experiences of our participant pool. In our recruitment strategy, for example, we did not systematically account for factors such as community size that might affect bot use [35] and related strategies for managing communities [37, 41]. Most participants described experiences on larger communities, which is in line with quantitative findings from Kiene and Hill [35]. However, as we sought to understand bot adoption as a *process*, we focused on sampling for various experiences with bots. Our participant pool covered a wide range of community topics and scales, user account ages on Discord, and individuals’ technical backgrounds.

Our empirical focus on Discord also brings into question how our findings might generalize to other platforms. We noted in §3 that Discord sees prolific use of its API, with thousands of bots [69], and produces multiple written resources for users. This may mean that Discord is unusually mature in the ways users adopt bots, and features like community leader hubs or support servers may not be common on other social platforms. Regardless, as our data includes a variety of experiences across different bots, we believe our findings offer valuable lessons about third-party bot adoption that are applicable to many other social computing platforms with public APIs. Although some details may be particular to Discord (e.g., top.gg), broader points about the process of adopting bots such as evaluating the fit of multiple bots and having security concerns are all likely to be seen in other online community platforms.

Future work might investigate how specific aspects of social ecosystems around bots differ across platforms and shape the ways people are able to shift and revise their technological frames. For example, on some platforms, affordances that facilitate direct communication across communities may not make sense or be normatively difficult to implement, such as in Slack workspaces. Understanding how intra-community support from community members can provide the informational benefits participants gained in intercommunity spaces in our work (e.g., looking to other communities to assess a bot's value and use) will be valuable to account for these differences when designing to support the adoption of user-facing tools more broadly. Quantitative studies might measure interpersonal dynamics in bot support servers that support successful bot adoption, or evaluate mechanisms to improve trust in third-party tools. Other work might focus on intra-community dynamics, i.e., building tools that can support troubleshooting strategies or community deliberation about bots (and other technical and infrastructural aspects of the community). Meanwhile, close examinations of bot stacking practices across users and discussions about bots in hubs can further our understandings of how users might creatively overcome the constraints of third-party tools on platforms. Research in these directions will help us better understand the consequences of the sociality of bot adoption in managing communities on platforms observed here.

7 CONCLUSION

We find that adopting third-party bots involved navigating heightened uncertainties about their security, reliability, and fit. Participants addressed these concerns through social ecosystems that emerged around bots as user-to-user networks of resources, support, and knowledge. These ecosystems helped participants revise their technological frames about bots and ultimately close knowledge gaps across diverse technical backgrounds. Given prior work highlighting barriers faced by users, this suggests that the diffusion of bots and similar tools on online community platforms be matched with efforts that facilitate interactions, social learning, and sharing about tools *across communities*. Participants also described receiving recommendations and help from community members during bot adoption, contrasting with prior work emphasizing the role of a small, skilled subset of users in bot adoption. The communal and collaborative dimensions of bot adoption both had practical implications and shifted participants' perception of the labor they were doing into something fun and satisfying. Finally, as the social ecosystems around bots shaped participant understandings about bots, our work shows that the ecosystems both supported and constrained how participants saw the value, functionalities, and possibilities of bots in their communities by aligning frames across communities. Together, these findings underscore the importance of attending to the social nature of adopting third-party bots in online communities, surfacing opportunities to better support the adoption of existing user-facing tools on platforms.

ACKNOWLEDGMENTS

We are grateful to our interview participants, who made the time to speak with us about the work they do in their communities — especially those who shared this study with their fellow community leaders. We would also like to especially thank Joseph Seering for his aid in the early stages of participant recruitment for this study; members of the Community Data Science Collective for feedback on early versions of this work; and the reviewers of this piece. Support for this project came from the National Science Foundation (IIS-1910202, DGE-2234667).

REFERENCES

- [1] Carliss Y. Baldwin and Kim B. Clark. 2000. *Design Rules: The Power of Modularity*. MIT Press, Cambridge, MA. <https://doi.org/10.7551/mitpress/2366.001.0001>

- [2] Richard Bentley and Paul Dourish. 1995. Medium versus Mechanism: Supporting Collaboration through Customisation. In *Proceedings of the Fourth Conference on European Conference on Computer-Supported Cooperative Work (ECSCW'95)*. Kluwer Academic Publishers, USA, 133–148.
- [3] Wiebe E Bijker, Thomas Parke Hughes, and T. J Pinch (Eds.). 1987. *The Social Construction of Technological Systems: New Directions in the Sociology and History of Technology*. MIT Press, Cambridge, MA.
- [4] Virginia Braun and Victoria Clarke. 2021. To Saturate or Not to Saturate? Questioning Data Saturation as a Useful Concept for Thematic Analysis and Sample-Size Rationales. *Qualitative Research in Sport, Exercise and Health* 13, 2 (March 2021), 201–216. <https://doi.org/10.1080/2159676X.2019.1704846>
- [5] Stevie Chancellor, Eric P. S. Baumer, and Munmun De Choudhury. 2019. Who Is the "Human" in Human-Centered Machine Learning: The Case of Predicting Mental Health from Social Media. *Proceedings of the ACM on Human-Computer Interaction* 3, CSCW (Nov. 2019), 1–32. <https://doi.org/10.1145/3359249>
- [6] Stevie Chancellor, Jessica Annette Pater, Trustin Clear, Eric Gilbert, and Munmun De Choudhury. 2016. #thyghgapp: Instagram Content Moderation and Lexical Variation in pro-Eating Disorder Communities. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing (CSCW '16)*. ACM, New York, NY, USA, 1201–1213. <https://doi.org/10.1145/2818048.2819963>
- [7] Eshwar Chandrasekharan, Chaitrali Gandhi, Matthew Wortley Mustelie, and Eric Gilbert. 2019. Crossmod: A Cross-Community Learning-Based System to Assist Reddit Moderators. *Proceedings of the ACM on Human-Computer Interaction* 3, CSCW (Nov. 2019), 1–30. <https://doi.org/10.1145/3359276>
- [8] Eshwar Chandrasekharan, Mattia Samory, Shagun Jhaver, Hunter Charvat, Amy Bruckman, Cliff Lampe, Jacob Eisenstein, and Eric Gilbert. 2018. The Internet's Hidden Rules: An Empirical Study of Reddit Norm Violations at Micro, Meso, and Macro Scales. *Proc. ACM Hum.-Comput. Interact.* 2, CSCW (2018), 32:1–32:25. <https://doi.org/10.1145/3274301>
- [9] Kathy Charmaz. 2015. *Constructing Grounded Theory: A Practical Guide through Qualitative Analysis* (2nd ed.). SAGE, Thousand Oaks, California.
- [10] Ruijia Cheng, Sayamindu Dasgupta, and Benjamin Mako Hill. 2022. How Interest-Driven Content Creation Shapes Opportunities for Informal Learning in Scratch: A Case Study on Novices' Use of Data Structures. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22)*. Association for Computing Machinery, New York, NY, USA, 1–16. <https://doi.org/10.1145/3491102.3502124>
- [11] Juliet Corbin and Anselm Strauss. 2015. *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory* (fourth edition ed.). SAGE Publications, Inc, Los Angeles.
- [12] Mark Diaz, Isaac Johnson, Amanda Lazar, Anne Marie Piper, and Darren Gergle. 2018. Addressing Age-Related Bias in Sentiment Analysis. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, Montreal QC Canada, 1–14. <https://doi.org/10.1145/3173574.3173986>
- [13] Bryan Dosono and Bryan Semaan. 2019. Moderation Practices as Emotional Labor in Sustaining Online Communities: The Case of Aapi Identity Work on Reddit. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. Association for Computing Machinery, Glasgow, Scotland Uk, 1–13. <https://doi.org/10.1145/3290605.3300372>
- [14] Brianna Dym. 2022. What Teens See in Closed Online Spaces like the Discord App. <http://theconversation.com/what-teens-see-in-closed-online-spaces-like-the-discord-app-178741>.
- [15] Nadia Eghbal. 2016. *The Unseen Labor behind Our Digital Infrastructure*. Technical Report. Ford Foundation. 143 pages.
- [16] Patricia I. Fusch and Lawrence R. Ness. 2015. Are We There yet? Data Saturation in Qualitative Research. *The Qualitative Report* 20, 9 (Feb. 2015), 1408–1416.
- [17] R. Stuart Geiger. 2014. Bots, Bespoke, Code and the Materiality of Software Platforms. *Information, Communication & Society* 17, 3 (March 2014), 342–356. <https://doi.org/10.1080/1369118X.2013.873069>
- [18] R. Stuart Geiger. 2016. Bot-Based Collective Blocklists in Twitter: The Counterpublic Moderation of Harassment in a Networked Public Space. *Information, Communication & Society* 19, 6 (June 2016), 787–803. <https://doi.org/10.1080/1369118X.2016.1153700>
- [19] R. Stuart Geiger and Aaron Halfaker. 2013. When the Levee Breaks: Without Bots, What Happens to Wikipedia's Quality Control Processes?. In *Proceedings of the 9th International Symposium on Open Collaboration (OpenSym '13)*. ACM, New York, NY, 6:1–6:6. <https://doi.org/10.1145/2491055.2491061>
- [20] R. Stuart Geiger, Dorothy Howard, and Lilly Irani. 2021. The Labor of Maintaining and Scaling Free and Open-Source Software Projects. *Proceedings of the ACM on Human-Computer Interaction* 5, CSCW1 (April 2021), 175:1–175:28. <https://doi.org/10.1145/3449249>
- [21] R. Stuart Geiger and David Ribes. 2010. The Work of Sustaining Order in Wikipedia: The Banning of a Vandal. In *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work*. Association for Computing Machinery, New York, NY, USA, 117–126. <https://doi.org/10.1145/1718918.1718941>
- [22] Ysabel Gerrard. 2018. Beyond the Hashtag: Circumventing Content Moderation on Social Media. *New Media & Society* 20, 12 (Dec. 2018), 4492–4511. <https://doi.org/10.1177/1461444818776611>

- [23] Tarleton Gillespie. 2018. "The Scale Is Just Unfathomable". <https://logicmag.io/scale/the-scale-is-just-unfathomable/>.
- [24] Tarleton Gillespie. 2020. Content Moderation, AI, and the Question of Scale. *Big Data & Society* 7, 2 (July 2020), 2053951720943234. <https://doi.org/10.1177/2053951720943234>
- [25] Aaron Halfaker, R. Stuart Geiger, Jonathan T. Morgan, and John Riedl. 2013. The Rise and Decline of an Open Collaboration System: How Wikipedia's Reaction to Popularity Is Causing Its Decline. *American Behavioral Scientist* 57, 5 (May 2013), 664–688. <https://doi.org/10.1177/0002764212469365>
- [26] Mona Haraty, Joanna McGrenere, and Andrea Bunt. 2017. Online Customization Sharing Ecosystems: Components, Roles, and Motivations. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing (CSCW '17)*. Association for Computing Machinery, New York, NY, USA, 2359–2371. <https://doi.org/10.1145/2998181.2998289>
- [27] Daniel G Heslep and PS Berge. 2021. Mapping Discord's Darksides: Distributed Hate Networks on Disboard. *New Media & Society* 00, 0 (Dec. 2021), 1–22. <https://doi.org/10.1177/14614448211062548>
- [28] Jane Hsieh, Joselyn Kim, Laura Dabbish, and Haiyi Zhu. 2023. "Nip It in the Bud": Moderation Strategies in Open Source Software Projects and the Role of Bots. *Proceedings of the ACM on Human-Computer Interaction* 7, CSCW2 (Sept. 2023), 1–29. <https://doi.org/10.1145/3610092>
- [29] Sohyeon Hwang and Aaron Shaw. 2022. Rules and Rule-Making in the Five Largest Wikipedias. *Proceedings of the International AAAI Conference on Web and Social Media* 16 (May 2022), 347–357. <https://doi.org/10.1609/icwsm.v16i1.19297>
- [30] Lilly C. Irani and M. Six Silberman. 2016. Stories We Tell about Labor: Turkopticon and the Trouble with "Design". In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. Association for Computing Machinery, New York, NY, USA, 4573–4586. <https://doi.org/10.1145/2858036.2858592>
- [31] Shagun Jhaver, Iris Birman, Eric Gilbert, and Amy Bruckman. 2019. Human-Machine Collaboration for Content Regulation: The Case of Reddit Automoderator. *ACM Trans. Comput.-Hum. Interact.* 26, 5 (July 2019), 31:1–31:35. <https://doi.org/10.1145/3338243>
- [32] Shagun Jhaver, Quan Ze Chen, Detlef Knauss, and Amy X. Zhang. 2022. Designing Word Filter Tools for Creator-Led Comment Moderation. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22)*. Association for Computing Machinery, New York, NY, USA, 1–21. <https://doi.org/10.1145/3491102.3517505>
- [33] Shagun Jhaver, Alice Qian Zhang, Quan Ze Chen, Nikhila Natarajan, Ruotong Wang, and Amy X. Zhang. 2023. Personalizing Content Moderation on Social Media: User Perspectives on Moderation Choices, Interface Design, and Labor. *Proceedings of the ACM on Human-Computer Interaction* 7, CSCW2 (Oct. 2023), 289:1–289:33. <https://doi.org/10.1145/3610080>
- [34] Jialun "Aaron" Jiang, Charles Kiene, Skyler Middler, Jed R. Brubaker, and Casey Fiesler. 2019. Moderation Challenges in Voice-Based Online Communities on Discord. *Proceedings of the ACM on Human-Computer Interaction* 3, CSCW (2019), 23. <https://doi.org/10.1145/3359157>
- [35] Charles Kiene and Benjamin Mako Hill. 2020. Who Uses Bots? A Statistical Analysis of Bot Usage in Moderation Teams. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems (CHI EA '20)*. Association for Computing Machinery, Honolulu, HI, USA, 1–8. <https://doi.org/10.1145/3334480.3382960>
- [36] Charles Kiene, Jialun "Aaron" Jiang, and Benjamin Mako Hill. 2019. Technological Frames and User Innovation: Exploring Technological Change in Community Moderation Teams. *Proceedings of the ACM on Human-Computer Interaction* 3, CSCW (Nov. 2019), 44:1–44:23. <https://doi.org/10.1145/3359146>
- [37] Charles Kiene, Andrés Monroy-Hernández, and Benjamin Mako Hill. 2016. Surviving an "Eternal September": How an Online Community Managed a Surge of Newcomers. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, 1152–1156. <https://doi.org/10.1145/2858036.2858356>
- [38] Charlotte P. Lee, Paul Dourish, and Gloria Mark. 2006. The Human Infrastructure of Cyberinfrastructure. In *Proceedings of the 2006 20th Anniversary Conference on Computer Supported Cooperative Work (CSCW '06)*. Association for Computing Machinery, New York, NY, USA, 483–492. <https://doi.org/10.1145/1180875.1180950>
- [39] Hanlin Li, Leah Ajmani, Moyan Zhou, Nicholas Vincent, Sohyeon Hwang, Tiziano Piccardi, Sneha Narayan, Sherae Daniel, and Veniamin Veselovsky. 2022. Ethical Tensions, Norms, and Directions in the Extraction of Online Volunteer Work. In *Companion Publication of the 2022 Conference on Computer Supported Cooperative Work and Social Computing (CSCW'22 Companion)*. Association for Computing Machinery, New York, NY, USA, 273–277. <https://doi.org/10.1145/3500868.3560923>
- [40] Hanlin Li, Brent Hecht, and Stevie Chancellor. 2022. All That's Happening behind the Scenes: Putting the Spotlight on Volunteer Moderator Labor in Reddit. In *Proceedings of the International AAAI Conference on Web and Social Media*, Vol. 16. AAAI, Atlanta, Georgia, 584–595.
- [41] Zhiyuan Lin, Niloufar Salehi, Bowen Yao, Yiqi Chen, and Michael S. Bernstein. 2017. Better When It Was Smaller? Community Content and Behavior after Massive Growth.. In *Eleventh International AAAI Conference on Web and Social Media*. AAAI, Montreal, Canada, 132–141.

- [42] Kiel Long, John Vines, Selina Sutton, Phillip Brooker, Tom Feltwell, Ben Kirman, Julie Barnett, and Shaun Lawson. 2017. "Could You Define That in Bot Terms?": Requesting, Creating and Using Bots on Reddit. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 3488–3500. <https://doi.org/10.1145/3025453.3025830>
- [43] Ming-Cheng Ma and John P. Lalor. 2020. An Empirical Analysis of Human-Bot Interaction on Reddit. In *Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020)*. Association for Computational Linguistics, Online, 101–106. <https://doi.org/10.18653/v1/2020.wnut-1.14>
- [44] Mykola Makhortykh, Aleksandra Urman, Felix Victor Münch, Amélie Heldt, Stephan Dreyer, and Matthias C Kettemann. 2022. Not All Who Are Bots Are Evil: A Cross-Platform Analysis of Automated Agent Governance. *New Media & Society* 24, 4 (April 2022), 964–981. <https://doi.org/10.1177/14614448221079035>
- [45] Andrés Monroy-Hernández, Benjamin Mako Hill, Jazmin Gonzalez-Rivero, and danah boyd. 2011. Computers Can't Give Credit: How Automatic Attribution Falls Short in an Online Remixing Community. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. Association for Computing Machinery, New York, NY, USA, 3421–3430. <https://doi.org/10.1145/1978942.1979452>
- [46] Claudia Müller-Birn, Leonhard Dobusch, and James D. Herbsleb. 2013. Work-to-Rule: The Emergence of Algorithmic Governance in Wikipedia. In *Proceedings of the 6th International Conference on Communities and Technologies*. ACM, New York, NY, USA, 80–89. <https://doi.org/10.1145/2482991.2482999>
- [47] Sabine Niederer and José van Dijk. 2010. Wisdom of the Crowd or Technicity of Content? Wikipedia as a Sociotechnical System. *New Media & Society* 12, 8 (Dec. 2010), 1368–1387. <https://doi.org/10.1177/1461444810365297>
- [48] Wanda J. Orlikowski. 1992. Learning from Notes: Organizational Issues in Groupware Implementation. In *Proceedings of the 1992 ACM Conference on Computer-supported Cooperative Work (CSCW '92)*. ACM, New York, NY, USA, 362–369. <https://doi.org/10.1145/143457.143549>
- [49] Wanda J. Orlikowski and Debra C. Gash. 1994. Technological Frames: Making Sense of Information Technology in Organizations. *ACM Trans. Inf. Syst.* 12, 2 (April 1994), 174–207. <https://doi.org/10.1145/196734.196745>
- [50] Trevor J. Pinch and Wiebe E. Bijker. 1984. The Social Construction of Facts and Artefacts: Or How the Sociology of Science and the Sociology of Technology Might Benefit Each Other. *Social Studies of Science* 14, 3 (Aug. 1984), 399–441. <https://doi.org/10.1177/030631284014003004>
- [51] Lev Poretski and Ofer Arazy. 2017. Placing Value on Community Co-Creations: A Study of a Video Game 'modding' Community. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing (CSCW '17)*. Association for Computing Machinery, New York, NY, USA, 480–491. <https://doi.org/10.1145/2998181.2998301>
- [52] Hector Postigo. 2007. Of Mods and Modders: Chasing down the Value of Fan-Based Digital Game Modifications. *Games and Culture* 2, 4 (Oct. 2007), 300–313. <https://doi.org/10.1177/1555412007307955>
- [53] Rifat Rahman, Md. Rishadur Rahman, Nafis Irtiza Tripto, Mohammed Eunus Ali, Sajid Hasan Apon, and Rifat Shahriyar. 2021. Adolescentbot: Understanding Opportunities for Chatbots in Combating Adolescent Sexual and Reproductive Health Problems in Bangladesh. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. ACM, Yokohama Japan, 1–15. <https://doi.org/10.1145/3411764.3445694>
- [54] Discord Safety. 2020. Discord Transparency Report: July – Dec 2020. <https://discord.com/blog/discord-transparency-report-july-dec-2020>.
- [55] Nathan Schneider, Primavera De Filippi, Seth Frey, Joshua Z. Tan, and Amy X. Zhang. 2021. Modular Politics: Toward a Governance Layer for Online Communities. *Proceedings of the ACM on Human-Computer Interaction* 5, CSCW1 (April 2021), 1–26. <https://doi.org/10.1145/3449090>
- [56] Angela M. Schöpke-Gonzalez, Shubham Atreja, Han Na Shin, Najmin Ahmed, and Libby Hemphill. 2022. Why Do Volunteer Content Moderators Quit? Burnout, Conflict, and Harmful Behaviors - Angela M. Schöpke-Gonzalez, Shubham Atreja, Han Na Shin, Najmin Ahmed, Libby Hemphill, 2022. *New Media & Society* 0(0) (Dec. 2022), 25.
- [57] Joseph Seering, Brianna Dym, Geoff Kaufman, and Michael Bernstein. 2022. Pride and Professionalization in Volunteer Moderation: Lessons for Effective Platform-User Collaboration. *Journal of Online Trust and Safety* 1, 2 (Feb. 2022), 1–20. <https://doi.org/10.54501/jots.v1i2.34>
- [58] Joseph Seering, Juan Pablo Flores, Saiph Savage, and Jessica Hammer. 2018. The Social Roles of Bots: Evaluating Impact of Bots on Discussions in Online Communities. *Proc. ACM Hum.-Comput. Interact.* 2, CSCW (Nov. 2018), 157:1–157:29. <https://doi.org/10.1145/3274426>
- [59] Joseph Seering, Tony Wang, Jina Yoon, and Geoff Kaufman. 2019. Moderator Engagement and Community Development in the Age of Algorithms. *New Media & Society* 21, 7 (Jan. 2019), 1–24. <https://doi.org/10.1177/1461444818821316>
- [60] Nancy Shaw, Lee-Partridge Eng, and James Ang. 1997. Understanding End-User Computing through Technological Frames. *ICIS 1997 Proceedings* 37 (1997), 8.
- [61] C. Estelle Smith, Irfanul Alam, Chenhao Tan, Brian C. Keegan, and Anita L. Blanchard. 2022. The Impact of Governance Bots on Sense of Virtual Community: Development and Validation of the Gov-BOTs Scale. *Proceedings of the ACM on*

- Human-Computer Interaction* 6, CSCW2 (Nov. 2022), 462:1–462:30. <https://doi.org/10.1145/3555563>
- [62] Susan Leigh Star. 1999. The Ethnography of Infrastructure. *American Behavioral Scientist* 43, 3 (Nov. 1999), 377–391. <https://doi.org/10.1177/00027649921955326>
- [63] Chenhao Tan and Lillian Lee. 2015. All Who Wander: On the Prevalence and Characteristics of Multi-Community Engagement. In *Proceedings of the 24th International Conference on World Wide Web (WWW '15)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, 1056–1066. <https://doi.org/10.1145/2736277.2741661>
- [64] Nicoletta Tancred, Selen Turkay, Nicole Vickery, Peta Wyeth, and Anna McCoombe. 2020. Understanding Women Modders Using the Serious Leisure Perspective. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–13. <https://doi.org/10.1145/3313831.3376351>
- [65] Nathan TeBlunthuis and Benjamin Mako Hill. 2022. Identifying Competition and Mutualism between Online Groups. In *International AAAI Conference on Web and Social Media (ICWSM 2022)*, Vol. 16. AAAI, Atlanta, Georgia, USA, 993–1004.
- [66] Nathan TeBlunthuis, Charles Kiene, Isabella Brown, Laura (Alia) Levi, Nicole McGinnis, and Benjamin Mako Hill. 2022. No Community Can Do Everything: Why People Participate in Similar Online Communities. *Proceedings of the ACM on Human-Computer Interaction: Computer Supported Cooperative Work* 6 (April 2022), 1–25. <https://doi.org/10.1145/3512908>
- [67] Jan E. Trost. 1986. Statistically Nonrepresentative Stratified Sampling: A Sampling Technique for Qualitative Studies. *Qualitative Sociology* 9, 1 (March 1986), 54–57. <https://doi.org/10.1007/BF00988249>
- [68] Eric von Hippel and Ralph Katz. 2002. Shifting Innovation to Users via Toolkits. *Management Science* 48, 7 (2002), 821–833. jstor:822693
- [69] Tom Warren. 2021. Discord Is Quietly Building an App Empire of Bots. <https://www.theverge.com/2021/11/17/22787018/discord-bots-app-discovery-platform>.
- [70] Galen Weld, Amy X. Zhang, and Tim Althoff. 2021. What Makes Online Communities 'Better'? Measuring Values, Consensus, and Conflict across Thousands of Subreddits. *arXiv:2111.05835 [cs]* 16 (Nov. 2021), 1121–1132. [arXiv:2111.05835 \[cs\]](https://arxiv.org/abs/2111.05835)
- [71] Mairieli Wessel. 2020. Leveraging Software Bots to Enhance Developers' Collaboration in Online Programming Communities. In *Conference Companion Publication of the 2020 on Computer Supported Cooperative Work and Social Computing (CSCW '20 Companion)*. Association for Computing Machinery, New York, NY, USA, 183–188. <https://doi.org/10.1145/3406865.3418368>
- [72] Yvette Wohn. 2019. Volunteer Moderators in Twitch Micro Communities: How They Get Involved, the Roles They Play, and the Emotional Labor They Experience. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI'19)*. ACM, Glasgow, UK, 1–13.
- [73] Bingjie Yu, Joseph Seering, Katta Spiel, and Leon Watts. 2020. "Taking Care of a Fruit Tree": Nurturing as a Layer of Concern in Online Community Moderation. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems (CHI EA '20)*. Association for Computing Machinery, New York, NY, USA, 1–9. <https://doi.org/10.1145/3334480.3383009>
- [74] Amy X. Zhang, Grant Hugh, and Michael S. Bernstein. 2020. PolicyKit: Building Governance in Online Communities. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*. ACM, Virtual Event USA, 365–378. <https://doi.org/10.1145/3379337.3415858> [arXiv:2008.04236](https://arxiv.org/abs/2008.04236)
- [75] Lei (Nico) Zheng, Christopher M. Albano, Neev M. Vora, Feng Mai, and Jeffrey V. Nickerson. 2019. The Roles Bots Play in Wikipedia. *Proc. ACM Hum.-Comput. Interact.* 3, CSCW (Nov. 2019), 215:1–215:20. <https://doi.org/10.1145/3359317>

A DESCRIPTION OF BOTS

Bot	Description	P#s
MEE6	Moderation (e.g. filtering) alongside content and engagement, e.g., user roles and alerts for Twitter content, that make it a more multipurpose tool	P4, P5, P6, P7, P12
Dyno	Moderation with a simple dashboard UI alongside some simple fun features such as random pug images.	P1, P3, P4, P5, P11 P12, P13, P16
Carlbot	Multipurpose: moderation, logging, polling, welcome messages, leaderboards, etc.	P2, P6, P11
Tatsu	Community engagement with games and leveling (gamify interactions as points).	P7-10
YAGPDB	Multipurpose: moderation, user roles, connecting social media feeds, cat facts, etc.	P8, P11, P16
Bot#1	Utility focus on logging activity, e.g. user joining or being banned.	P5, P10
Bot#2	Advanced customizable moderation, geared for large-scale communities.	P3, P4, P12, P13
Bot#3	Multipurpose: administration, moderation, games, survey-building, trivia, etc.	P1, P14
Bot#4	Anti-spam, specifically to block spam bots that “raid” communities.	P15, P9, P13
Bot#5	Community engagement with games and leveling.	P2, P8
Bot#6	Smart link detection and filtering for anti-scam and anti-virus defense.	P12
Bot#7	Utility and admin support such as archiving, spam configuration, and user information.	P10, P13
Bot#8-10	Moderation bots with more advanced configuration than a toggle-option dashboard.	P1, P6, P14, P16
Bot#11	Bot enabling live-time music sharing.	P4, P14,
Bot#12	ID verification of users to deter bots and alternative accounts of banned users.	P7

Table 2. Descriptions for key bots discussed by participants in the study. The table includes bots that were either mentioned frequently or in interesting cases to contextualize the findings. We note the participants who most prominently discussed these bots.

Received January 2023; revised October 2023; accepted January 2024.