Proceedings Paper:
Long, Kiel, Vines, John, Sutton, Selina et al. (5 more authors) (2017) "Could You Define That in Bot Terms?":Requesting, Creating and Using Bots on Reddit. In: CHI '17 Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems. 2017 ACM CHI Conference, 06-11 May 2017 CHI proceedings . ACM , pp. 3488-3500.

https://doi.org/10.1145/3025453.3025830

"Could You Define That in Bot Terms?": Requesting, Creating and Using Bots on Reddit

Kiel Long[1], John Vines[2], Selina Sutton[2], Phil Brooker[3], Tom Feltwell[2], Ben Kirman[4], Julie Barnett[3] and Shaun Lawson[2]

[1] Open Lab, Newcastle University, Newcastle upon Tyne, UK, kiel.long@newcastle.ac.uk

[2] Northumbria University, Newcastle upon Tyne, UK, {firstname.lastname}@northumbria.ac.uk

[3] University of Bath, Bath, UK, {p.d.brooker, j.c.barnett}@bath.ac.uk

[4] University of York, York, UK, ben.kirman@york.ac.uk

ABSTRACT

Bots are estimated to account for well over half of all web traffic, yet they remain an understudied topic in HCI. In this paper we present the findings of an analysis of 2284 submissions across three discussion groups dedicated to the request, creation and discussion of bots on Reddit. We set out to examine the qualities and functionalities of bots and the practical and social challenges surrounding their creation and use. Our findings highlight the prevalence of misunderstandings around the capabilities of bots, misalignments in discourse between novices who request and more expert members who create them, and the prevalence of requests that are deemed to be inappropriate for the Reddit community. In discussing our findings, we suggest future directions for the design and development of tools that support more carefully guided and reflective approaches to bot development for novices, and tools to support exploring the consequences of contextually-inappropriate bot ideas.

Author Keywords

Bots; Reddit; Online communities; Co-creation.

ACM Classification Keywords

H.5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous;

INTRODUCTION

Social media platforms have recently seen a proliferation of automated software agents, known as *bots*, which can monitor and participate in simple online communication. The current scale of online bot activity is remarkable, with bots being responsible for 24% of all activity on Twitter [54]. While their purpose on social media varies widely, bots are characterized by routine behaviours that often respond to other activity according to some designed criteria. For instance, if someone used the phrase "illegal immigrant" on Twitter in July 2015, they might have received an automated reply from "@DroptheIBot" [24] suggesting they rethink on their terminology and use the term "undocumented immigrant" instead. Bots deployed on social media platforms are often quick to gain notoriety. Through these platforms' low barriers to social interaction, bots propagate news [34] and political opinion [8]; they counter, respond to and correct statements made by users (such as @DroptheIBot); they name and shame users based on the things they say or content they share [47]; help users appeal parking tickets without needing lawyers [61]; publish generative art (e.g. @MothGenerator); give directions to mysterious places [28]; and attempt humour (e.g @AmIRiteBot). Outside of social media, bots are also an established feature of communities such as Wikipedia, Slack and Reddit, where they provide ways to automate the standard protocols that govern platform content and perform other tasks that are time-consuming for human administrators [11, 33, 36, 52]. However, poor design and implementations of bots can have negative effects. Microsoft's Tay Twitter bot [42] was quickly pulled when, influenced by mischievous human peers, it began to post highly offensive racist, sexist and homophobic material. Bots gained further notoriety in exposés of the dating site Ashley Madison in which it was revealed that many customer interactions with seemingly real users were, in fact, with bots [32].

The recent proliferation of bots has been supported by the development of new tools and services which have lowered barriers to their creation and deployment. The availability of simple, well-documented application programming interfaces (APIs) that support ever more accessible languages and frameworks opens bot creation to novice developers. Meya [63], Labnol [3] and "Cheap Bots Done Quick" [60] provide tools that allow non-developers to design, create and deploy simple bots without writing a line of code. The Weavrs platform gained widespread notoriety [2] for facilitating the creation, at the press of a button, of relatively unsophisticated but strikingly effective and subversive Twitter bots seeded with existing profiles of real celebrities. A recent well-publicised online tutorial describing *"How to Make a Twitter Bot in Under an Hour: Even if you don't code that often!"* [43] contributes to an

CHI 2017, May 06-11, 2017, Denver, CO, USA

ACM 978-1-4503-4655-9/17/05.

http://dx.doi.org/10.1145/3025453.3025830

expanding set of reference material regarding the process of making bots. Furthermore, events such as "Bot Summit" [27] and "Art of Bots" [59]) offer bot-making enthusiasts opportunities to talk about bots, collaboratively make them, swap expertise and offer help.

In this paper, we present a qualitative study of the ways in which bots are discussed, created and used by a growing bot development community on Reddit. Our study comprises an analysis of 2,284 submissions and 14,822 comments on three Reddit sub-communities, known as *subreddits*. We selected these subreddits as they explicitly aim to bring together Reddit users who not only create and program bots but also those who would like to request bots to be made. Given the perceived democratization of bot development, we were motivated to investigate why people create bots in the first place, and the ways in which novice and more expert developers discuss the practical and social issues surrounding bot creation and use. The findings from our study offer two contributions to the emerging HCI discourse on bots. First, we provide an empirical study on the collaborative creation of bots in large online communities, from which we offer insights around the social norms of bot design and use on Reddit, the challenges that novices and experts face in the collaborative creation of bots, and the misunderstandings and misconceptions around what bots can and should do. Second, grounded in our empirical findings, we contribute directions for future research around the design of applications and platforms to support the collaborative and reflective creation of bots for social platforms.

RELATED WORK

A significant amount of prior work has focused on problematic aspects of bots, such as their use in attacking websites or online services (e.g. [8, 57]) or their impersonation of humans (e.g. [10, 46]). Motivated by allegations around democratic votes in both Europe and the US, Forelle et al [17], Baker [6], and the wider research community at politicalbots.org have highlighted the widespread use of bots to disseminate and influence political opinion, 'to boost follower numbers and to retweet the content of political candidates on Twitter, to attack political opponents on Facebook, or to drown out activists' conversations on Reddit' [58, p4885]. Likewise, Larsson and Moe [30] note the need for researchers and platform developers to deepen their understandings of how bot accounts influence and propagate news and media distribution. Relatedly, Savage et al [48] present an approach for using Twitter bots as a mechanism for calling volunteers to action around social causes, highlighting the ease with which changing the tone of the language expressed by a bot can influence engagement from human social media users [48]. The growing abundance of bot code shared on platforms such as GitHub opens even further opportunities and lowers the level of expertise needed to tailor, deploy and use these software agents for personal, social or political causes [29].

As well as these wider social, technical and political implications of bot use, there is growing recognition of the important role bots play in automating otherwise burdensome and repetitive processes on platforms such as Wikipedia [11, 36] and Slack [33]. Clément and Guitton [11] categorise bots on Wikipedia into two opposing ideotypes: 'servant' bots performing laborious work in place of human volunteers (e.g. correcting grammatical error); and 'policing' bots enforcing guidelines and norms. They note how the users of these bots predictably perceive them as servant collaborators under their control, yet users disapprove of 'servant' bots unwantedly performing numerous interfering actions across a large number of pages. Furthermore, 'policing' bots are perceived to be limiting and constraining, making contributors feel as though bots were aggressively controlling purportedly voluntary decisions. This tension and potential conflict between human and bot editors on Wikipedia has been explored by Geiger both in terms of how bots fit within established roles [18] and what happens where bots that perform vital roles fail [19].

Perhaps unsurprisingly, significant amounts of research on bots has focused on preventing them being used in the first place. Technologies such as CAPTCHA are used to stop bot accounts being automatically created [22], and much cybersecurity research has focused on enhancing these procedures (e.g. [46]). Others have studied bots to develop tools to understanding the propagation of SPAM on social media [57], and have developed techniques to detect bots based on content of tweets [10] and comparing bot-based accounts to those operated by humans [1]. This has led to the development of tools such as BotOrNot [12] which aids social media users to distinguish whether an account is operated by a person or an automated agent; and "Stweeler", which supports the analysis of the impact and influence of bots on Twitter [20].

STUDY DESIGN

While previous work has examined the influence of bots within online communities, far less is known about how and why people engage in their creation. We address this by examining a community of bot makers and bot users, and identifying key themes that emerge out of discussions around bot creation. In this case, we chose Reddit as a site of enquiry for gathering discourse on bot creation and culture due to both its active community and the ease of programmatic access to the public comments on the site.

Reddit and Bots

Reddit is an extremely popular [4] online social media, news aggregation, content rating, and discussion forum that attracts almost 250 million unique monthly users [51]. Reddit users post submissions to discussion fora, or subreddits, which focus on specific themes or topics. Subreddit names are prefixed with "/r/"; for example, `/r/funny` and `/r/WorldNews`. An example of the layout of a subreddit in a web browser is given in Figure 1. A core

characteristic of Reddit is that submissions can receive up and down votes from other users to show their relevance and value, and to facilitate social navigation [21]. The more successful a submission the more prominently it appears within a subreddit, with highly up-voted submissions appearing on the front page of Reddit itself. Submissions can also receive comments, which can also receive up or down votes, with the most up-voted comments on a submission appearing at the top of a comment tree. The votes that individual users receive, on either their submissions or comments, tallies as "karma" and reflects their contribution to the community. Through these processes and practices Reddit seeks to maintain new and relevant user submissions [50]; the site is 'arguably the internet's largest social voting community' [21, p803].

Like many online platforms, Reddit relies on volunteer moderators. However, its scale, the sheer amount of content generated [21], a number of well-publicised controversies related to the content users share on the site [25], and divisive dynamics between subreddits [45] have highlighted the particularly active role users play in the moderation and self-governance of the platform. Given the popularity of the site, a culture of using bots as automated support for human moderators has developed. Well-known bots that perform moderator tasks include `/u/Botwatchman`, which detects and removes other bots, and `/u/Automoderator`, a customizable moderation tool. As is typical with other platforms, bot development and implementation is facilitated through the openly-available Reddit API and the associated Python Reddit API Wrapper (PRAW) which both offer a range of scripted functionality [64]. This has, unsurprisingly, also led to the development of many other bots that perform tasks other than moderation; indeed, Reddit now plays host to an extremely active and diverse bot-enthusiast community who often mash together other third-party APIs, services and platforms to conceive and implement a wide range of useful, or merely entertaining bots. Popular examples include `/u/autotldr` which reduces and summarizes long text-based submissions; `/u/autowikibot` (now retired) which inserted summaries of Wikipedia content when such links were detected; and `/u/tyo-translate`, which translates a comment to the supposed writing style of a twelve-year-old.

Data Collection

We collected data from three subreddits that are explicitly focused on the discussion of bots: `/r/requestabot`, `/r/botrequests` and `/r/botwatch` (henceforth referred to as `RaB`, `BR` and `BW`). Note that due to their similarities `BR` was merged with `RaB` in 2014, with `BR` remaining in place 'for archiving purposes' [66] with 150 accessible submissions. Casual browsing (see Figure 1) of the content of these two subreddits shows that they are largely comprised of exchanges between Reddit users who have an idea for a bot and are seeking advice from skilled developers as to how to go about building it themselves, or wish to find an altruistic

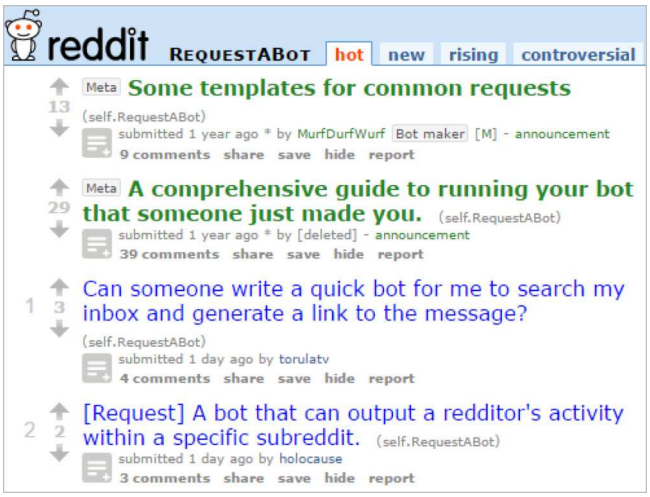

Figure 1. A recent screenshot from `/r/requestabot` showing a typical subreddit layout with up- and down-voted posts and comments. It also illustrates some typical 'bot requests'.

developer to build it for them. The welcome text attached to `BR` captures this well:

> *"Welcome to /r/botrequests. If you are a bored programmer, or a person with an idea for a bot, you can waste some of your time here. Just post a self. post explaining the idea for the bot, and someone might code it, for you."* [65]

The nature of the discussion on `BW` is somewhat broader as its own welcome text indicates:

> *"This reddit is dedicated to the continued interest, observation, discussion and study of reddit bot accounts and related topics"* [67]

We used the Reddit API [64] to construct an archive of all submissions and subsequent comments across each of the three subreddits. Data collected ranged from 15th May 2012 (the first submission on `BW`) through to the 16th June 2016. In total, we collected 2284 submissions (1344 for `RaB`, 150 for `BR` and 790 for `BW`) and 14822 comments (4984 for `RaB`, 461 for `BR` and 9377 for `BW`), which were used in their entirety as the dataset for our analysis.

Data Analysis

We used a qualitative, two stage, approach to understanding our data. First, as an entry point into this large dataset, we were interested in understanding the overall nature of the bots requested and created by this community, in particular the *issues they address* and their *technical functionalities*. We therefore first conducted a Content Analysis [16, 35], inductively coding each submission to capture initial patterns related to (i) the types of topics, issues and tasks users requested or created bots for, and (ii) the types of functionalities and technical features that underpinned requested and created bots. Second, we were interested in the discussions surrounding bots on the Reddit platform. As such, we conducted a Thematic Analysis of all submission and comment data. Following [9], we coded individual submissions and comments, when necessary at the sentence to paragraph level, to summarise content for semantic and latent meaning. Once all data was coded, we compared and

contrasted codes, grouped related codes together, and used these as the starting point for creating themes. Finally, we selected exemplar data as evidence of talk underpinning themes to be presented as part of our findings. The coding process across both stages was conducted by three researchers, who regularly met to share codes, discuss different interpretations of data, and to refine codes and subsequent themes.

In the following sections we report on the findings of our analysis. Where we use excerpts from data, these have been assigned (i) a randomised anonymised numeric username, (ii) a letter associated with when they occur in a comment thread and (iii) appended with the abbreviated subreddit label: e.g. `RaB999B` refers to a comment in subreddit `/r/requestabot` by "`user 999`" which was the second comment ("`B`") in a submission thread. Following current best practice [23], we have amended quoted data to preserve anonymity of users but to avoid altering meaning.

FINDINGS #1: BOTS REQUESTED AND CREATED

Our Content Analysis examined the issues bots that were requested and created address, as well as their technical functionalities. We provide a summary of this data in Table 1. We coded a large amount of submissions as 'Not Applicable' (806), either because they were submissions that did not explicitly refer to a requested or created bot, were repetitions of prior submissions, or were requests for creating bots for other platforms.

Issues Addressed by Bots

The vast majority of bots were proposed to support *Administration* duties (705) across the site. This included a range of bots that automated duties and tasks associated with being a moderator of subreddits, such as: scheduled posting (e.g `RedditLater (RaB116A)`), the automatic and scheduled posting of content and links from other sites (e.g posting when a Twitch stream is live `(BR071A)`) and deleting old posts (`BW065A`). After this, *Play and Humour* (278) included bots that involved the playing of games on Reddit, such as `TickTackToeBot` (`BW762A`) and `RockPaperScissorBot` (`BW223A`). Here, we also include attempts at humorous bots that would search for specific terms in users' posts and respond to these, e.g. `MonsterMathBot` (`BR032A`) and `Theyre_Minerals_Bot` (`BR035A`). *Functionality and Quality* (206) bots sought to enhance the functionality and improve the quality of content across the platform. New functionalities included currency conversion (`BR139A`), language translation (`RaB409A`) and temperature conversion (`BW598A`) tools. Others searched for specific content with a view to correcting or enhancing it, such as `gandhi_spell_bot` (`BR006A`) or converting images to be legible on mobiles (`RaB966A`). Bots intended to address *Community* issues (149) dealt with community management and moderation support in specific subreddits. Evoking the previously described underprovision of moderation on Reddit [21], this included bots that pre-banned blacklisted users (`RaB014A`) and archived posts for moderation purposes (`RaB254A`). More positive examples, however, included bots that wish users happy anniversary (`RaB119A`) and welcome them upon their first post in a subreddit (`RaB439A`). Finally, *Archiving* bots (140) were the simplest bots and involved searching for and archiving specific content, either by reposting onto new subreddits or downloading them. In these examples it was often unclear what the purpose of this archive was.

Issues addressed by Reddit bot requests and creations			
Administration	**705**	**Functionality / Quality**	**206**
Archiving	**140**	**Play / Humour**	**278**
Community	**149**	*Not Applicable*	*806*
Functions and technical features of Reddit bots			
Calling	**70**	**Querying (deleting)**	**51**
Private Messaging	**23**	**Querying (posting)**	**48**
Posting	**220**	**Querying (reposting)**	**64**
Querying (account)	**105**	**Querying (responding)**	**686**
Querying (archiving)	**74**	**Querying (messaging)**	**84**
Querying (converting)	**53**	*Not Applicable*	*806*

Table 1. Summary of findings from analysis of issues and technical functionalities addressed by bots requested and created across `RaB`, `BR` and `BW`.

Functionalities of Bots

By far the most popular functionalities of bots were those that *Queried and Responded* (686) to specific posted content across Reddit, such as posts containing particular words, or posts with the most comments in designated subreddits. After this, the second most popular functionality for a bot was automatically *Posting* (220) content from other sources, such as YouTube videos, Tweets or scheduled posts from a database. Further, there was a range of other bots with functionalities that *Queried* particular content and data for different reasons, such as: *Querying User Accounts* (105) to identify users with certain qualities (e.g. membership duration); *Querying* a specific search criteria and then *Messaging* (84) specific user accounts the results; *Querying and Archiving* on own computer (74) or *Reposting* content on another subreddit (64) based on specific search criteria; *Querying and Converting* content to be reposted or manipulated in some way (53); *Querying and Deleting* (51) content, specifically for moderation purposes; and *Querying* certain content and using this as the basis for new *Postings* (48). A further set of functionalities related to *Calling* (70), where bots respond to being "called" by users and post automated content. Finally, bots using the *Private Messaging* functions of Reddit were the smallest in our data (23); these would message a designated group of users at specific times.

FINDINGS #2: DISCUSSING BOTS

The Thematic Analysis of the submission and comment data led to five key themes: *knowledge and skills*, *technical infeasibility*, *legitimate and valuable bots*, *inappropriate and annoying bots*, and *the value of building a bot*.

Knowledge and Skills

As one would expect, a large number of the submissions from BR and RaB were requesting the creation of a bot idea. In these, some posters disclosed their lack of knowledge of how bots work and inability to program: *"I know nothing about making bots, I know nothing about the capabilities of bots, and I know nothing about hosting bots or getting them to actually work."* (RaB106A). However, typically an idea would be proposed without such disclosures and a requester's lack of expertise and knowledge would become apparent through discussion with more expert bot creators. For example, in some cases, those responding to a request would offer tips and advice on how to create the requested bot:

> *"The easiest fix would be to start with an empty string and add to it for each keyword [gives example code] […]. Use a dictionary bot as a starting point. I would encourage you to check out ReplyDict, which is a modified version of ReplyBot for doing this kind of thing."* (RaB548B)

Such suggestions, while well-meant, were often very unclear to those requesting the bot: *"Thanks, but I'm just an idiot with a stupid idea. I have no programming experience at all. Thanks for the time though"* (BR123E). Quite significantly, such naivety often greatly affected understandings of what was possible, followed by realisations that ideas requested were much more challenging to implement than first envisaged:

> *"I'm now realising this is not as easy as I thought. Can I do this myself without an education in programming, or is there someone that can walk me through it?"* (BR146E)

Similarly, there are frequent misunderstandings between bot requesters and bot creators, often due to requesters not fully understanding the technical language of a creator: *"Sorry, I'm not a programmer. I don 't really have a server. I just agreed with what you said because you are a professional so I thought you would know what you are talking about"* (RAB560C). Such instances highlighted how much of the language used by more established developers was exclusionary to novice requesters and frequently impeded understanding. The differing levels of knowledge between requesters and creators was most clearly demonstrated in situations where bot creators respond with code for the requester to use. Quite often such code would be provided by simply pasting it as text into a comment, or occasionally creators would direct a requester to a source code repository (e.g. on GitHub). However, most requesters clearly did not know what to do with this code: *"Thanks for that* [the code]*, but I really don't know how to apply this to anything."* (BR129C); *"Thanks but I haven't a clue what that all means or how I use it. If you could make it for me I would be grateful!"* (BR128E); *"I thought Git is just an old insult."* (BW754C).

Beyond not knowing what to do with the code a developer may provide, a further layer of confusion arises around how to implement it in a way that allows a bot to operate over a sustained period of time. While many creators appear happy to build a bot for free, they are reluctant to host them due to the costs incurred. Reddit bots can be run and hosted in a range of ways: most creators recommend requesters acquire server space, although bots can also be run from a personal computer. However, many requesters had little idea that their bot would have to be *"run from somewhere"* (RaB26C):

> BR94B: *For you to run a bot you must have a place to host it. It can be a computer that is always on, or you can rent one. Do you have one?*
>
> BR94C: *Oh, i guess i dont have that. i have my laptop but that isn't always on. i'm a student and poor so i can't rent one. I'll ask around and see if someone else does."*

Indeed, as was the case of BR94C, it is often the perceived costs of running a bot—or the burden of having to keep a personal machine connected to the Internet all the time—that leads to many bot creations not being fully implemented. The issue of bots not being implemented was so well known that some more knowledgeable requesters promised a minimum duration of hosting a bot someone creates for them: "*as a promise to the person who makes me this, I guarantee I will host it for at least six months*" (RaB1212A).

Technical Feasibility

A frequent occurrence across the data were discussions between requesters and potential bot creators around the feasibility of proposed ideas. There was often a great amount of confusion from requesters about what a bot is, and specifically what functions a bot can perform on the Reddit platform. In some cases those requesting a bot were requesting relatively simple functions that many users could already access via Reddit's inbuilt admin tools:

> RaB1273A: *Is it possible for a bot to tag images as NSFW, even if it doesn't find the text NSFW in a post?*
>
> RaB1273B: *Hang on, so you want every single post in a subreddit tagged at NSFW? Isnt that a simple moderation option for your sub?*"

In other cases, requesters would be asking for bots to do relatively simple automatic procedures (e.g. keyword search (RaB218A) or posting content from other sites (RaB208A)) and would be directed to search services or other solutions. However, while there were situations where requests could be covered by other types of services, the vast majority of difficulties were due to the infeasibility of proposed ideas. Many requesters appear self-aware of the potential oversights within their proposals: *"This might not be possible or even legal. but it would be awesome if it was!"* (RaB012A); *"To be honest this is just a proof of concept, and I bet there are a lot of issues that would cause problems."* (RaB347A). A frequent point of contention were requests that required levels of contextual interpretation that the simple algorithms underpinning bots are unable to support. For example, RaB1332A requested an adaptation to an existing bot (FallacyBot) so that it searched for posts *"containing fallacies"*, quoted these, and provided an explanation of why this was a fallacy. After some discussion about the proposal, a commenter responds:

> *"You know, _FallacyBot_ cant do that. It cant actually detect fallacies, it searches for words like 'ad hominem' and 'red herring' etc and responds to them. Bots can't understand context, so what you want is actually impossible."* (RaB1332F)

In a similar vein, BR089A requests a JudgeJudy bot: "*This bot will be used when a user is an absolute twat within the courts of reddit. I want the bot to say SILENCE!"*. A respondent asks: *"Could you define what absolute twat means in quantitative terms?"* (BR089B), while another explains:

> *"The issue here is that bots are stupid and only do exactly what you tell them to do. You cannot tell a bot to determine whether someone is an absolute twat unless there are certain words or phrases related to being a twat that trigger the bot."* (BR089C)

The complexities of context and language often arose around requests for 'pedantic' and 'humorous' bots that intended to correct spelling and grammar: *"In theory you've got a good idea, but in reality it corrects things that aren't mistakes at all. You know, "Anxious to" is perfectly fine. It's not a misuse of "anxious".* (BW412D). Others highlighted the complexity of such seemingly simple bots: "*It will have to differentiate between correct use of both valid spellings."* (BR007C). The potential for bots like these to deviate from their intended function and suggest wrongful corrections (and thus becoming spam bots) was an oft-cited concern: *"You want to correct isles to aisles, but hey, what if someone is actually talking about isles?! That's a spam bot. Would you appreciate a message saying you suck at spelling? No. No you would not"* (BR009B). These examples illustrate how bot requesters frequently attributed a level of sophistication and complexity to bots that are highly unrealistic.

Unexpectedly high expectations around technical feasibility were not just limited to context awareness. Other frequently occurring issues including ambitious ideas that would require large amounts of data processing (*"I think you'll find scanning all the data on reddit, twitter and youtube is quite a large task"* (RaB025B)) or uncertainty over the types of search terms a bot might query the Reddit API for (*"I see what you want, but finding keywords for that is complicated. Your words would be detected in A LOT of posts, and would be different in meaning*" (BR148B)). It was common for potential creators to question the nature of the data that would be inputted into a bot (*"What is the bot looking for and then what does it do? Where does it get its information?"* (RaB1330)) or the unfeasibility of accessing required data in the first place: "*A bot can't just scrape information from an article, It needs an organised file that's updating every day. This is not doable.*" (RaB117E). Other ideas were quickly discarded as being completely unfeasible based on the functionality of Reddit: such as a bot that only replied to bots masquerading as human users (*"How on earth would you ever be able to determine which accounts were bots and which humans?"* (RaB213C)); and a bot to perform shopping tasks on an external website (*"Well, ok, this is way beyond building reddit bots. But good luck, I guess, but I don't really think this will be feasible"* (BR030D).

As we see here, there are a range of situations where technically novice or naive requesters of bots are unable to understand the complexity of the ideas being suggested. Somewhat echoing Nass and Moon's canonical work on computers as social actors [39], to some requesters what appears to be relatively simple tasks most human beings can do should be easily attributable to what appear to be complex automated agents.

Legitimate and Valuable Bots

Although there were many examples where bot requests are not developed due to technical infeasibility, some ideas would be explored further and developed. In some cases, requesters gave detailed 'requirements' for their idea:

> *"I'm looking for a coder who can write and host an automatic bot. But if you can only create the bot and not host it then that is ok. What does the bot have to do? It will be a broadcast bot. This means it will send private messages to people on a list. The list of people to be broadcast to is chosen by the users via PMs […] What and when should the bot broadcast? The bot should broadcast any private messages that [user] sends to it. It should these when it gets those messages from me. Finally, if possible that bot should also respond to people enabling and disabling private messages with the message: [example]."* (RaB104A)

While direct and perhaps presumptuous, such a level of detail is usually well-received by potential bot creators. Listing a request like this suggested that requesters had already researched into the feasibility of their idea, and that while they lacked knowledge of how to implement the bot they knew a little about what needed to be done and why. Furthermore, in such requests, as in the above, requesters also clearly infer who is going to use the bot, and for what purposes (e.g. *"PS: This bot will be used on my subreddit where I am a mod, see:* [link]*."* (RaB004)). Adding such details made requests appear more legitimate and have purpose, providing a sense of scale of the need and, thus, the value to others of the bot being made:

> *"We have a complicated flair system over at [subreddit], here is our [stylesheet] and here is our flair [wiki page]. We made the system when we were a lot smaller. Now we have around 7,000 subs a day and we can't keep up."* (RaB1132)

Relatedly, those with more ambiguous ideas were often questioned on their intentions: "*You haven't provided a lot of detail. We know you want a bot that replies with -something- to a post that has -some phrase- in a link title that appears in -some subreddit-*" (RaB218C). Respondents would ask for elaboration on the specifics of the bot. While this had a functional and technical value, it was also an attempt to explore the requester's motivations for wanting the bot in the first place: "*Please explain why you want this? It can be easily abused*" (BR141D). In this example, the requester wanted a configurable bot that could post specific information to a particular subreddit at a specific time each day. The specificity of the request, matched with ambiguity around the details (e.g., what information, for what subreddit) led to suspicion around their motivation. However, the requester evidenced that they were the moderator for the subreddit in question and linked to specific examples of the content they wished to post. The revealing of these specific details—and a requester with credentials—resulted in a more positive engagement with the request.

A further way in which community members engaged in exploring the legitimacy and value of bot ideas was by using already existing bots as a starting point for discussion.

This might involve the requester referring to an existing bot and asking for it to be tweaked for their own purposes:

> *"This would be similar to [bot] apart that it would check for a parent comment by the opening poster of the link post after 6 hours. If no parent comment is found, then the post would be removed until said comment is left by the OP."* (RaB11A)

Again, providing examples of already existing bots added legitimacy to ideas and provided exemplar functionality without the need for in-depth technical description. Likewise, commenters and creators would provide examples in response to requests in order to probe what the requester was seeking to achieve: *"What you're asking for is what* [bot] *does. Except it takes the fastest rising story every hour."* (BW030B). Such examples would often include creators referencing source-code repositories or subreddits dedicated to specific bots for requesters to look at, or in some cases creators directing the requester to a creator's own pre-made code repositories ("*I've got just the thing for you* [GitHub link]" (BR004B)). This quite often led to positive responses from requesters: "*Your countries bot looks very close to what I need*." (BR021C). However, the provision of code related to existing bots resulted in similar responses to those noted earlier—enthusiasm around the response their idea received, but uncertainty on what to do next with the code.

Inappropriate and Annoying Bots

The legitimacy and value of a bot idea was also established in regards to how a request might comply (or does not comply) with the terms and norms that the Reddit community, and specific subreddits, operate on. Certain ideas would fail to comply with the terms and conditions of the Reddit API. Issues such as number of request calls or amount of data needed to be queried would often be a limiting factor of bot development: "*You want every comment on reddit? Crazy. The account would be banned in minutes*" (BR127C). In those cases where bots were feasible via the API, but still involved querying for and responding to large amounts of content across the community, distinctions between what was acceptable and inacceptable were blurred. For example, while many requests aimed to conform to *"the spirit"* (RaB567G) of the community and even seen as fun (e.g. RaB567A's popular request for a bot responding to most popular Gif images with a version with frames reversed) their automated nature (operating continuously and unrequested across all subreddits) was seen to be potentially invasive and inappropriate. Similarly, some bots were seen to be *"following the rules but not the spirit"* (BR143D): *"This is easy to build, but I question if it's a good idea."* (RaB133C). Many suggestions were discouraged and rejected as they would simply annoy or upset people, such as bots to correct spelling or grammar: "*I like this idea, but it would mean the bot *commenting* on each post with corrected grammar, which is a bigger nuisance than the original errors*." (RaB003D).

While much of the discussion on RaB and BR was oriented towards halting spam bots before they are created, commentators noted that certain ideas not necessarily considered likely to spam were also causing problems. On occasions, bots requested and created were viewed as spam-like because of the accidental use of wrong code or poor implementation: "*Dude, your bot is spamming all over* [subreddit]. *We like the idea but it's looping on itself."* (BW865D); *"Really sorry. I've stopped it and will debug it"* (BW865E). In other cases, however, oversights in a bot's design caused well-meant bots to become spam bots. One of the more notorious examples of these was FallacyBot, which was the first "exemplar" bot request:

> *"FallacyBot is the first bot made as a request. Bots that are requested should follow some unwritten guidelines that FallacyBot follows. It is simple, designed to do a small number of tasks, and is not used maliciously."* (RaB1348A)

However, while intended to act as an example of good practice for others to follow, this bot caused discontent:

> *"We're having a discussion over in [subreddit] over whether something is, or is not, a loaded question. Your bot has spammed the thread like ten times or more with the definition. It's frustrating as hell, and it will get banned. Can you change it to only post once per thread?"* (RaB1348B)

> *"This is pretty much the worst. It's so annoying. We really, honestly, don't need a bot telling us what a strawman is 400 times. I hate the bot."* (RaB1348F)

Over time, FallacyBot was modified; initially to only post a maximum of two times on a submission, then to work only when summoned. Eventually, it was announced that it had *"died"* (BW676A): *"Glad to hear it."* (BW676B). While FallacyBot was a very prominent example, many bots—both proposed or created—were criticised because of a similar sense of inappropriateness. For example, BoobBot3000 (BW34A) would respond to users who say the word 'boobs' with *"hehe… you said boobs!"*:

> *"If this is a "noobs" bot, well done for learning to code, but now learn how to keep a bot relevant and avoid people from being annoyed. Especially, avoid certain subreddits where the audience won't appreciate it like /r/askwomen."* (BW34D)

It was clear across the data that the precise nature of how appropriateness is determined in relation to bot use was hard to define. The same bots that some find hilarious, were also seen to be annoying, insulting and deeply offensive depending on the context of where they appear:

> *"Hey, I saw a bot tell someone in /r/RaisedByNarcissists to "turn your frown upside down". That is not helpful advice, especially in a context where people discuss abuse. I really hope that bot doesn't pull that in /r/Depression."* (BW407B)

In cases like this, bots created to *"add a bit of joy"* (BW407F) had troubling capabilities for certain audiences and in particular contexts. However, for the most part the problematic qualities of these bots were unintentional. In a smaller number of contrasting examples, requests made for bots that explicitly targeted specific communities or individuals were identified as particularly problematic. Bots that replied to only specific users whenever they posted a comment (BR025A's bot requested to *"annoy his sister"*), that spammed specific subreddits (RaB1259's RelevancyBot) or those that targeted *"repeat offenders"* of

grammatical pickiness (RaB1337) were all highly questioned. In a further set of examples, across two separate submissions different requesters asked for a bot that "*looks through a female users post history to try and find out if they have any gonewild submissions"* (RaB1252A) and *"a bot that checks whether a girl has submitted a link to r/gonewild"* (BR096A). These requests—which involve scraping links to images on a subreddit dedicated to *'open-minded Adult Redditors to exchange their nude bodies for karma'* made by users who may post elsewhere on Reddit—were responded to negatively by some: *"sounds like creeperbot"* (BR096B); *"That isn't very nice."* (RaB1252B). However, the idea also received endorsements from many more users: *"good idea"* (RaB1252G), and *"this is awesome!"* (RaB1252J) and was implemented.

Bots that are seen to be inappropriate and annoying often raise much discussion, with a focus on the context of their use. While many bots were not intended to offend, without a nuanced understanding of context they can easily be seen to do so. Similarly, a lack of understanding of context, inside humour and sub-community norms can lead to well-meant ideas being discarded, or labelled offensive.

The Value of Building a Bot

Our final theme discusses the ways those who create and build bots that are requested express the value they get from their creations. Along with the technical feasibility and appropriateness of a bot, it was also clear that potential creators took into account a range of other factors when deciding whether a bot was worth creating or not. For seemingly more confident and experienced bot creators, those bots that were seen as offering value tended to be those that had perceived utility across a broad audience or a range of communities. This included bots that calculated currency conversions (RaB1004A), or bots that translated queried terms into other languages (BW273A). Creators of bots express a desire to program agents that had not only visibility across the community, but were seen to be useful and thus be used by others: *"that's a neat party trick, but what value does it add?"* (BW140G). This is not to say seemingly frivolous bots were not created however. Some were built because creators *"liked this idea"* (BR138B on `XKCD bot`), found it *"incredibly interesting"* (BW626D on `Godwin's Law bot`) or thought it was *"hilarious"* (RaB1207A on a bot that attempted to engage in disjointed conversation with other users). However, there was still an undertone here that large audiences for such bots were envisioned: *"I can see this going down well over a r/humour"* (BW626F).

The creation of bots was also seen as important to less experienced programmers. Some creators treated a requested bot as a project: "*I'll have this bot project*" (RaB420B), "*This sounds like a very interesting project!*" (RaB287C). A bot project was treated as an opportunity to try out or develop programming skills: *"I 'm looking at this as a fun project to learn a little python (I 've been a frontend guy for some time)"* (RaB222). Bot projects were seen as a good way of entering the world of programming, and provided an opportunity to *"play about"* and *"learn some new skills"* (BR070A) in a safe, low-risk, environment: "*I want to make a bot that gets some info* […] *then I would like to teach myself to add other bits to it as well, but learning the table, praw and Reddit API at once seems enough for the first take*" (RaB1029A); "*That sounds pretty easy. You've made me want to learn python and just do it*" (RaB573C). In a similar vein, the creation of bots was also an opportunity for some more experienced programmers to push themselves. Some respondents to requests noted they were *"up for the challenge"* (BR03B).

Perhaps because of the perceived value that bot creators get from having their work be seen across the platform and be used by members of the community, there was often great disappointment when a project failed to be implemented or was poorly received. One of the longest discussions on `BW` related to `TLDRBot`, a bot that summarises long comments in 4 sentences. The bot was very well received by other users of `BW`, and was implemented in a way to be summoned. However, although initially well-received, the owner of it *"shut it down"*, noting that *"it wasn't liked, they even me banned from a subreddit"* (BW402G). The creator of `TARDIS-BOT`—a bot that would randomly choose to respond to an archived post—was *"bemused"* by the *"profanities thrown at my poor bot. Several times someone down voted everything on the front page of the sub."* (BW4008A). Similarly, `BlackjackBot` received a huge amount of praise from the `BW` community but its creator received a lot of *"hate"*:

> *"Wow. A lot of hate and a lot of bans. […]. I guess this bot sucks. […] I don't play xbox live, so I'm not used to being called a "faggot" so frequently." (BW756F)*

Finally, while bots might be built by creators, they might not go on to be implemented properly. It is common to see submission threads where people ask *"does this bot still work"* (BR612D), or note that they *"think the bot died"* (BW498) or they *"shut it down"*: "*I have shut the bot down, the script is no longer running. This means that the bot will no longer work. Thank you to everybody who gave it a try."* (BW383G). Final comments in submission threads would frequently be where creators announce a completed bot, which seemingly was never implemented. It was thus expected that bots had a limited *"shelf life"*: *"I saw a bot do that once* […] *It was a couple of months ago so it is probably dead now"* (BR134AF). As such, coming back to our earlier points around the re-use of old bots, great enthusiasm occurred when a creator was presented with a new requester who wanted to bring their bot *"back from the dead"*: "*Yes, please use it! The other guy never put it on a server!*" (RaB627U).

DISCUSSION

In our study we have explored the motivations, challenges and opportunities around bot creation among communities of interest on Reddit. Although we have examined the use of bots on a platform with its own functional qualities and social norms, it nonetheless offers a valuable set of insights into the ways in which bots—and simple software agents more generally—are imagined, produced and valued by online communities.

Given the huge presence of bots on the web, the increasing numbers of bots being used on social media, and the increasing accessibility of tools for bot creation, it is important to explore how wider audiences and users of social platforms understand what they are and how they work. This is all the more important as bots start traversing into new domains and contexts, which in some situations (as in the Ashley Madison bot) can be a potential source of exploitation, manipulation and abuse. Furthermore, as bots become increasingly employed to propagate political messages (e.g. [6, 17, 48])—or used as ways of publically experimenting with new forms of artificial intelligence (as per Tay [42])—it is increasingly important that the wider public is supported in understanding how these technologies work, how they propagate their messages and how, if necessary, they can be questioned and countered.

Our findings are valuable for understanding these challenges further. We saw how many people simply have no idea what bots *are* at a functional level, how they act and react to data, or how sophisticated they can and cannot be. This included little understanding of the technical fundamentals of bots; budding botsmiths did not appreciate the additional infrastructure required to host and operate bots, nor understand the most basic of programming and networking competencies required to realise their idea. There were also issues around the possibilities and limitations of the APIs that serve as the lifeblood of any bot. In addition to these technical matters, there were issues of expectation around richness of interaction, such as a bots' ability to understand context or subtleties of language, perhaps in part due to the cultural mythology that surrounds robots and artificial intelligence [55]. This speaks to wider contemporary issues where the public have little awareness or understanding of the software processes that shape and constrain the things we see online [9]. In many respects, these raise deep concerns about the potential to engage wider publics in probing and questioning the roles software agents play in society.

At the same time, however, we also saw examples where discussions on Reddit exposed opportunities for engaging with and learning about the underlying software processes through which bots operate. We observed nuanced co-creation processes surrounding proposing, discussing, motivating, releasing and maintaining a bot. We saw the ways in which the suggestion of bot ideas, the receiving of feedback on these ideas from a supportive community of developers, and then the iteration of these through discussion, enabled lay community members to see their initial ideas come to life. We also observed how existing bots became exemplars and prototypes on which new ideas were built. They became ways to peek behind the curtain of bot design and implementation. In some cases this was through watching the trial and error of a creator. In others it was through learning new coding skills themselves via the help and support of more experienced peers. As such, while there was in some cases no exchange of code the interactions observed were akin to forms of legitimate peripheral participation [31] observed in open source communities [13, 68] and in situated studies of learning programing skills [13, 49]. Therefore, while there were clearly challenges associated with the creation of bot ideas requested by novices, there is great potential to see acts of making bots as sites where the mysterious "black box" of technology can be unpicked and understood. In the following sections we discuss three areas where future work might support such bot making, and reflect on the value of studying bots as a way of understanding online communities. In discussing these future directions, we connect with and build on related issues in the fields of end-user development and collaborative design.

Expressing Ideas for Bots

As reported, there were many examples where those who were proposing bots struggled to express their ideas, could not elaborate on initial propositions, or simply had a very limited knowledge of what was technically feasible. There is potential here, however, for exploring the ways in which online environments might be designed or reconfigured to support the expression of ideas around software agents. Inspiration might be found here in literature that has examined the ways in which other online platforms support exchanges of knowledge and advice between expert and novice coders [26]. For example, Asaduzzaman et al [5] have examined why requests for help get left unresolved on platforms like StackOverflow—they highlight that those questions that are 'too short, unclear, vague or hard to follow', too specific or perceived to involve too much work often get ignored. Contrastingly, Nashei et al [38] highlight that good questions on the same platform tend to encourage continued discussion where the initial problem gets redefined. Similarly, those responses that work best tend to be those that are concise, refer back to the question asked, highlight key elements, and provide step-by-step instructions and explore multiple solutions. Furthermore, studies of expert online communities of programmers have highlighted the importance of timely responses, both from those asking a question and those proposing answers [40].

These works offer interesting parallels with many of the issues that the bot requesters and creators faced. Those requests that were most well-received tended to be those that offered detailed 'requirements', or where requesters responded in a timely and open-minded way to those who were offering to create their bot. However, there were clearly distinct issues associated with these being interactions between 'lay' users and more expert programmers. This included impenetrable language (for both parties, but especially novices), requests that were technically flawed from the start, and then creations that fail to be implemented due to a lack of understanding or resource from its requester. As such, building on the above work, we might imagine ways in which online environments carefully guide those who have a bot idea to express their ideas in a more organized way. Critically,

drawing on recent work on end-user toolkits [41], it would be important to guide users with natural language and notation to reflect on both the idea being proposed and the contextual and interpretative limitations of bots. For example, we might imagine a bot requester wishing to create a "humorous" bot being prompted with: "As a bot I can't understand jokes, but I can tell you one". The benefit of such prompts would be to gently direct those with ideas to understanding the limitations of these simple software entities, and also to spur new ideation.

Experimenting with Bots

While a significant amount of collaborative practice around design in HCI orientates itself towards enabling conditions where participants can be open-minded [15], withhold judgement [56] and imagine radically different futures [14], in many respects the opposite issue is the matter of concern here. The challenge was to shape requesters' ideas in ways to fit their imaginations to the capabilities of simple bots. It is often claimed that digital technologies can be "black boxes" that are 'impermeable, inflexible, and unviewable' [37]. Indeed, consumer products are critiqued for creating conditions where these hidden innards disable people from appropriating, re-making and engaging in their own practices of design [53]. Therefore, we might take inspiration from attempts to support DIY practices around maker technologies and kits for experimentation with input and output platforms [53]. Simple tools that provide exemplar functions and types of data that act as bot "building blocks" might offer a starting point for guided experimentation. One way of doing so might be to present already existing and typical designs that require completing (as suggested by [44]) through adding or taking away component blocks that represent specific functionality. In doing so, we might more carefully scaffold learning through safe experimentation, and also extend the current positive examples we saw where exemplar bots would be used as a starting point to inspire or ideate a new bot creation by more novice users.

Future Ramifications of Bots

Due to the unique space that bots occupy on social platforms and their automated power, it is imperative that connections are made between bot design and user consequences. In the situations we observed, experienced bot developers would provide guidance based on knowledge of the community, its spirit, and its 'unspoken rules'. Their experience also allowed for the safeguarding of the community through identifying possible negative consequences of proposed designs and alerting both requesters and creators of how where a bot operates and how it responds can quickly make a "fun" bot offensive. Such safeguarding of communities and promotion of maintaining social norms to novice developers is particularly critical in self-moderating and bot facilitating communities such as Reddit. We might imagine the ways in which these supportive, protective and mentoring practices might also be embedded in bot creation tools. For instance, services like *If This Then That* [62] provide ways of ideating and automating functions and actions within and across platforms; however, we might go further and say that there is a need for ideating the consequences of actions, especially with bots on social platforms. Future tools could incorporate expressively richer rule sets and operate within a number of dynamic requirements, similar to recent proposals for Internet of Thing end-user programming kits that are considerate to diverse and 'fuzzy' conditions [7]. At one level, sandpit type environments where bot ideas can be functionally tested against APIs and formal policies are a good start; providing clear, common language feedback to creators about why their creation has failed. However, there is also a need to test the social consequences of bots: "if I do this" then "this is how they will react" or "this is how they might feel". Such environments would give clues as to why bots would be inappropriate, which audiences may and may not react well to them, and, potentially, scaffold the creation of more positive, joyful bots in the future.

CONCLUSION

This paper has examined the user practices of requesting and creation of bot ideas on Reddit. While this is a specific context of bot use, it speaks to wider issues around bots as they continue to become an everyday feature of online communication. As a large online community open to bots through its API, Reddit is a noteworthy case as we can observe both the community reaction to new bots on the service, but also the process by which new bots are made. In particular, our study has examined the interactions between developers and novice bot creators, as they negotiate and discuss the features of new bots and their potential positive and negative implications. This exposes some of the practical issues around bot creation, such as understanding technical capabilities of bots, but also the importance of social responsibility in bot development, and what constitutes appropriate behaviour of bots within this social context. As such, by observing community members negotiate appropriateness in automation we gain a valuable lens on the ideas, values and matters of concern for that community.

ACKNOWLEDGEMENTS

We would like to thank Kellie Morrissey for edits and comments on the paper. This work was supported by the RCUK grant ES/M003574/1 "CuRAtOR: Challenging online feaR And OtheRing". CuRAtOR is funded through the Empathy and Trust in Online Communicating (EMoTICON) funding call administered by the Economic and Social Research Council in conjunction with the RCUK Connected Communities, Digital Economy and Partnership for Conflict, Crime and Security themes, and supported by the Defence Science and Technology Laboratory (Dstl) and the Centre for the Protection of National Infrastructure (CPNI). Data underpinning this paper is available from the authors

REFERENCES

1. Norah Abokhodair, Daisy Yoo, and David W. McDonald. 2015. Dissecting a Social Botnet: Growth, Content and Influence in Twitter. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing* (CSCW '15). ACM, New York, NY, USA, 839-851. DOI: http://dx.doi.org/10.1145/2675133.2675208
2. Sally Adee. 2012. Digital doppelgangers: building an army of you. *New Scientist*, *215*(2877), 38-41.
3. Amit Agarwal. 2016. *How to Write a Twitter Bot in 5 Minutes* www.labnol.org/internet/write-twitter-bot/27902/
4. Alexa – statistics for Reddit.com http://www.alexa.com/siteinfo/reddit.com
5. Muhammad Asaduzzaman, Ahmed Shah Mashiyat, Chanchal K. Roy, and Kevin A. Schneider. 2013. Answering questions about unanswered questions of stack overflow. In *Proceedings of the 10th Working Conference on Mining Software Repositories (MSR '13)*. IEEE Press, Piscataway, NJ, USA, 97-100.
6. Vicky Baker. 2015. Battle of the bots: Fake social media accounts on the attack. *Index on Censorship*, 44 (2), 127-129. DOI: 10.1177/0306422015591470
7. Barbara Rita Barricelli, and Stefano Valtolina. 2015. Designing for end-user development in the internet of things. In *International Symposium on End User Development*, pp. 9-24.
8. Yazan Boshmaf, Ildar Muslukhov, Konstantin Beznosov, and Matei Ripeanu. 2011. The socialbot network: when bots socialize for fame and money. In *Proceedings of the 27th Annual Computer Security Applications Conference (ACSAC '11)*. ACM, New York, NY, USA, 93-102. DOI: http://dx.doi.org/10.1145/2076732.2076746
9. Virginia Braun and Victoria Clarke. 2006. Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3(2), 77-101.
10. Zi Chu, Steven Gianvecchio, Haining Wang, and Sushil Jajodia. 2010. Who is tweeting on Twitter: human, bot, or cyborg?. In *Proceedings of the 26th Annual Computer Security Applications Conference (ACSAC '10)*. ACM, New York, NY, USA, 21-30. DOI: http://dx.doi.org/10.1145/1920261.1920265.
11. Maxime Clément and Matthieu J Guitton. 2015. Interacting with bots online: Users' reactions to actions of automated programs in Wikipedia. *Computers in Human Behaviour* 50 (2015), 66–75
12. Clayton Allen Davis, Onur Varol, Emilio Ferrara, Alessandro Flammini, and Filippo Menczer. 2016. BotOrNot: A System to Evaluate Social Bots. In *Proceedings of the 25th International Conference Companion on World Wide Web (WWW '16 Companion). International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland*, 273-274. DOI: http://dx.doi.org/10.1145/2872518.2889302
13. Nicolas Ducheneaut. 2005. Socialization in an open source software community: A socio-technical analysis. *Computer Supported Cooperative Work (CSCW)* 14, no. 4, 323-368.
14. Pelle Ehn, and Morten Kyng, 1991. Cardboard computers in Design at Work: Cooperative Design of Computer Systems. eds Greenbaum. J. & Kyng, M Lawrence Erlbaum Associates, Hillsdale, NJ.
15. Haakon Faste, Nir Rachmel, Russell Essary, and Evan Sheehan. 2013. Brainstorm, Chainstorm, Cheatstorm, Tweetstorm: new ideation strategies for distributed HCI design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, New York, NY, USA, 1343-1352. DOI: http://dx.doi.org/10.1145/2470654.2466177
16. Uwe Flick, Ernst von Kardoff, and Ines Steinke, (eds.) 2004. *A companion to qualitative research*. Sage
17. Michelle C. Forelle, Philip N. Howard, Andrés Monroy-Hernández, and Saiph Savage. 2015. Political Bots and the Manipulation of Public Opinion in Venezuela (July 25, 2015) Available at SSRN 2635800
18. R. Stuart. Geiger, 2011. The lives of bots. In *G. Lovink and N. Tkacz (eds.) In Wikipedia: A Critical Point of View. Amsterdam: Institute of Network Cultures* 78-93.
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* (WikiSym '13). ACM, New York, NY, USA, DOI: http://dx.doi.org/10.1145/2491055.2491061
20. Zafar Gilani, Liang Wang, Jon Crowcroft, Mario Almeida, and Reza Farahbakhsh. 2016. Stweeler: A Framework for Twitter Bot Analysis. In *Proceedings of the 25th International Conference Companion on World Wide Web (WWW '16 Companion). International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland*, 37-38. DOI: http://dx.doi.org/10.1145/2872518.2889360
21. Eric Gilbert. 2013. Widespread underprovision on Reddit. In *Proceedings of the 2013 conference on Computer supported cooperative work* (CSCW '13). ACM, New York, NY, USA, 803-808. DOI: http://dx.doi.org/10.1145/2441776.2441866
22. Philippe Golle and Nicolas Ducheneaut. 2005. Preventing bots from playing online games. *Comput. Entertain.* (July 2005), 3-3. DOI: http://dx.doi.org/10.1145/1077246.1077255

23. Claire Hewson, and Tom Buchanan. 2013. Ethics guidelines for internet-mediated research. *The British Psychological Society*
24. Patrick Hogan, and Jorge Rivas. 2015. *We built a Twitter bot that corrects people who say 'illegal immigrant'*. http://fusion.net/story/165089/we-built-illegal-immigrant-bot/ (accessed 16/09/2016)
25. Elias Isquith. 2015. *Reddit's ugly, racist secret: How it became the most hateful space on the Internet.* From: http://www.salon.com/2015/03/18/reddits_ugly_racist_secret_how_it_became_the_most_hateful_space_on_the_internet/
26. M. Cameron Jones and Elizabeth F. Churchill. 2009. Conversations in developer communities: a preliminary analysis of the yahoo! pipes community. In *Proceedings of the fourth international conference on Communities and technologies* (C&T '09). ACM, New York, NY, USA, 195-204. DOI: http://dx.doi.org/10.1145/1556460.1556489
27. Darius Kazemi. 2016. *Bot Summit 2016*. http://tinysubversions.com/botsummit/2016/
28. Ben Kirman, Conor Linehan, and Shaun Lawson. 2012. Get lost: facilitating serendipitous exploration in location-sharing services. In *CHI '12 Extended Abstracts on Human Factors in Computing Systems* (CHI EA '12). ACM, New York, NY, USA, 2303-2308. DOI: http://dx.doi.org/10.1145/2212776.2223793
29. Bence Kollanyi. 2016. Automation, Algorithms, and Politics| Where Do Bots Come From? An Analysis of Bot Codes Shared on GitHub. *International Journal Of Communication,* 10, 20.
30. Anders Olof Larsson, and Moe Hallvard. 2015. Bots or journalists? News sharing on Twitter. *Communications - The European Journal of Communication Research*, 40 (3), pp. 361-370.
31. Jean Lave and Etienne Wenger. 2002. Legitimate peripheral participation in communities of practice. *Supporting lifelong learning.* 1. 111-126.
32. Ben Light. 2016. The rise of speculative devices: Hooking up with the bots of Ashley Madison. *First Monday*.21 (6). DOI: http://dx.doi.org/10.5210/fm.v21i6.6426
33. Bin Lin, Alexey Zagalsky, Margaret-Anne Storey, and Alexander Serebrenik. 2016. Why Developers Are Slacking Off: Understanding How Software Teams Use Slack. In *Proceedings of the 19th ACM Conference on Computer Supported Cooperative Work and Social Computing Companion* (CSCW '16 Companion). ACM, New York, NY, USA, 333-336. DOI: http://dx.doi.org/10.1145/2818052.2869117
34. Tetyana Lokot, and Nicholas Diakopoulos. 2015. News Bots: Automating news and information dissemination on Twitter. *Digital Journalism*. 1-18.
35. Annette N. Markham and Nancy K. Baym. 2008. *Internet Inquiry: Conversations about Method*. Sage Publications, Inc., Thousand Oaks, CA, USA.
36. 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* (C&T '13). ACM, New York, NY, USA, 80-89. DOI: http://dx.doi.org/10.1145/2482991.2482999.
37. Bonnie Nardi and Jannis Kallinikos. 2007. Opening the Black Box of Digital Technologies: Mods in *World of Warcraft. In Proceedings of 23rd EGOS Colloquium*.
38. Seyed Mehdi Nasehi, Jonathan Sillito, Frank Maurer, and Chris Burns. 2012. What makes a good code example?: A study of programming Q&A in StackOverflow. In *28th IEEE International Conference on Software Maintenance (ICSM)*, 25-34
39. Clifford, Nass and Youngme Moon. 2000. Machines and mindlessness: Social responses to computers. *Journal of social issues* 56, no. 1. 81-103.
40. Kawin Ngamkajornwiwat, Dongsong Zhang, A. Gunes Koru, Lina Zhou, and Robert Nolker. 2008. An exploratory study on the evolution of oss developer communities." In *, Proceedings of the 41st Annual Hawaii International Conference on System Sciences*. 305-305.
41. Charith Perera, Saeed Aghaee, and Alan Blackwell. 2015. Natural Notation for the Domestic Internet of Things. In *International Symposium on End User Development*, pp. 25-41.
42. Sarah Perez. 2016. *Microsoft silences its new A.I. bot Tay, after Twitter users teach it racism [Updated]*. www.techcrunch.com/2016/03/24/microsoft-silences-its-new-a-i-bot-tay-after-twitter-users-teach-it-racism/
43. Daniel Peterschmidt. 2016. *How to Make a Twitter Bot in Under an Hour Even if you don't code that often!* From www.medium.com/science-friday-footnotes/how-to-make-a-twitter-bot-in-under-an-hour-259597558acf#.5ewopdxjt
44. Alexander Repenning, and Andri Ioannidou. 2006. What makes end-user development tick? 13 design guidelines. In *End User Development*, pp. 51-85.
45. Roopika Risam. 2015. Toxic femininity 4.0. *First Monday*, 20(4). DOI: 10.5210/fm.v20i4.5896
46. Yong Rui and Zicheg Liu. 2003. Excuse me, but are you human?. In *Proceedings of the eleventh ACM international conference on Multimedia* (MULTIMEDIA '03). ACM, New York, NY, USA,

462-463. DOI: http://dx.doi.org/10.1145/957013.957113

47. Aviva Rutkin. 2014. Twitter bots grow up and take on the world. *New Scientist.* 223(2980). 20-21.
48. Saiph Savage, Andres Monroy-Hernandez, and Tobias Höllerer. 2016. Botivist: Calling Volunteers to Action using Online Bots. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing* (CSCW '16). ACM, New York, NY, USA, 813-822. DOI: http://dx.doi.org/10.1145/2818048.2819985
49. Shilpa V. Shukla, and David F. Redmiles. 1996. Collaborative learning in a software bug-tracking scenario. In *Workshop on Approaches for Distributed Learning through Computer Supported Collaborative Learning*, pp. 16-20.
50. Philipp Singer, Fabian Flöck, Clemens Meinhart, Elias Zeitfogel, and Markus Strohmaier. 2014. Evolution of reddit: from the front page of the internet to a self-referential community?. In *Proceedings of the 23rd International Conference on World Wide Web* (WWW '14 Companion). ACM, New York, NY, USA, 517-522. DOI: http://dx.doi.org/10.1145/2567948.2576943
51. Statista - statistics for Reddit.com http://www.statista.com/statistics/443332/reddit-monthly-visitors/
52. Thomas Steiner. 2014. Bots vs. wikipedians, anons vs. logged-ins. In *Proceedings of the 23rd International Conference on World Wide Web* (WWW '14 Companion). ACM, New York, NY, USA, 547-548. DOI: http://dx.doi.org/10.1145/2567948.2576948
53. Cristiano Storni. 2014. The problem of de-sign as conjuring: empowerment-in-use and the politics of seams. In *Proceedings of the 13th Participatory Design Conference: Research Papers - Volume 1* (PDC '14), Vol. 1. ACM, New York, NY, USA, 161-170. DOI: http://dx.doi.org/10.1145/2661435.2661436
54. Sysomos.com. 2009. An In-Depth Look at the Most Active Twitter User Data from https://sysomos.com/inside-twitter/most-active-twitter-user-data
55. Alex S. Taylor. 2009. Machine intelligence. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '09). ACM, New York, NY, USA, 2109-2118. DOI: http://dx.doi.org/10.1145/1518701.1519022
56. John Vines, Mark Blythe, Stephen Lindsay, Paul Dunphy, Andrew Monk, and Patrick Olivier. 2012. Questionable concepts: critique as resource for designing with eighty somethings. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '12). ACM, New York, NY, USA, 1169-1178. DOI: http://dx.doi.org/10.1145/2207676.2208567
57. Alex Hai Wang. 2010. Don't follow me: Spam detection in twitter. In *Proceedings of the 2010 International Conference on Security and Cryptography (SECRYPT)*. 1-10. IEEE
58. Samuel C. Woolley and Philip N. Howard. 2016. Automation, Algorithms, and Politics| Political Communication, Computational Propaganda, and Autonomous Agents—Introduction. *International Journal of Communication* 10 9.
59. www.andfestival.org.uk/events/art-of-bots-london/
60. www.cheapbotsdonequick.com
61. www.donotpay.co.uk
62. www.ifttt.com
63. www.meya.ai
64. www.reddit.com/dev/api
65. www.reddit.com/r/botrequests
66. www.reddit.com/r/botrequests/comments/29z0nt/merging_with_rrequestabot
67. www.reddit.com/r/botwatch/
68. Yunwen Ye and Kouichi Kishida. 2003. Toward an understanding of the motivation Open Source Software developers. In *Proceedings of the 25th International Conference on Software Engineering* (ICSE '03). IEEE Computer Society, Washington, DC, USA, 419-429.