ModSandbox: Facilitating Online Community Moderation Through Error Prediction and Improvement of Automated Rules

Jean Y. Song[*]
Electrical Engineering and Computer Science, DGIST
Daegu, Republic of Korea
jeansong@dgist.ac.kr

Sangwook Lee[*]
School of Computing, KAIST
Daejeon, Republic of Korea
sangwooklee@kaist.ac.kr

Jisoo Lee
Beeble
Seoul, Republic of Korea
jisoo.lee@beeble.ai

Mina Kim
Kakao Corp
Pangyo, Republic of Korea
iamhappy537@gmail.com

Juho Kim
School of Computing, KAIST
Daejeon, Republic of Korea
juhokim@kaist.ac.kr

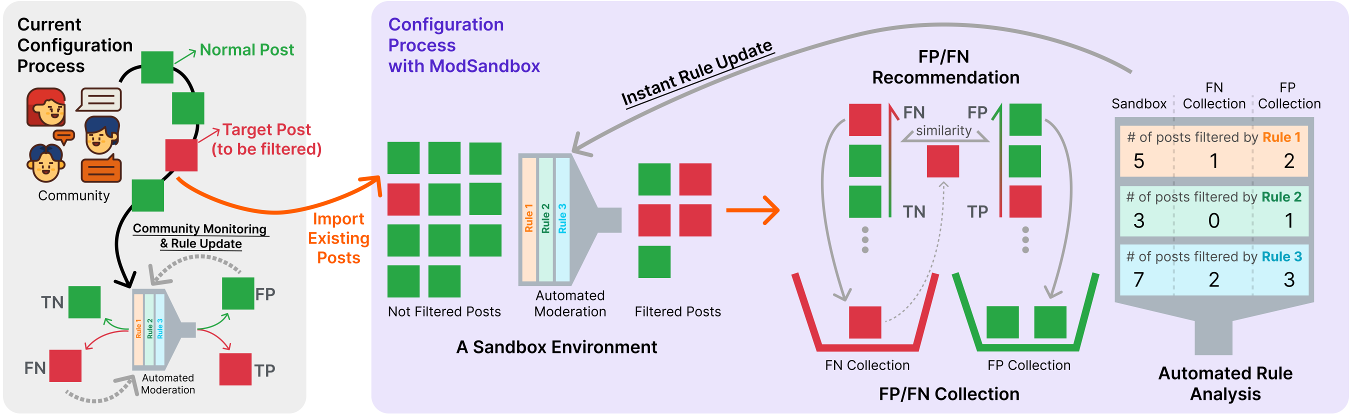


Figure 1: ModSandbox supports online community moderators with error prediction and improvement of their automated rules. The moderators currently monitor their community and update the rules to catch the target posts (posts they want to filter) based on their previous experience with false positives and negatives. ModSandbox provides features to help predict possible false positives and false negatives using existing posts (A Sandbox Environment and FP/FN Recommendation), and to improve automated rules (FP/FN Collection and Automated Rule Analysis).

ABSTRACT

Despite the common use of rule-based tools for online content moderation, human moderators still spend a lot of time monitoring them to ensure they work as intended. Based on surveys and interviews with Reddit moderators who use AutoModerator, we identified the main challenges in reducing false positives and false negatives of automated rules: not being able to estimate the actual effect of a rule in advance and having difficulty figuring out how the rules should be updated. To address these issues, we built ModSandbox, a novel virtual sandbox system that detects possible false positives and false negatives of a rule and visualizes which part of the rule is causing issues. We conducted a comparative, between-subject study with online content moderators to evaluate the effect of ModSandbox in improving automated rules. Results show that ModSandbox can support quickly finding possible false positives and false negatives of automated rules and guide moderators to improve them to reduce future errors.

CCS CONCEPTS

• **Human-centered computing → Human computer interaction (HCI); Human computer interaction (HCI).**

KEYWORDS

sociotechnical systems; moderation; automated moderation bots; online communities; virtual sandbox; human-AI collaboration

[*] Equal contribution.

CHI '23, April 23–28, 2023, Hamburg, Germany
© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-9421-5/23/04…$15.00
https://doi.org/10.1145/3544548.3581057

ACM Reference Format:
Jean Y. Song, Sangwook Lee, Jisoo Lee, Mina Kim, and Juho Kim. 2023. ModSandbox: Facilitating Online Community Moderation Through Error Prediction and Improvement of Automated Rules . In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23), April 23–28, 2023, Hamburg, Germany*. ACM, New York, NY, USA, 20 pages. https://doi.org/10.1145/3544548.3581057

1 INTRODUCTION

Communities on social platforms such as Reddit, Discord, and Twitch have a group of users who volunteer to moderate their communities, called online moderators [25, 34]. They respond to the behavior of community members that violate rules and work to improve overall interaction experiences between community members [20, 47]. Contrary to large social platform companies like Facebook and Twitter that apply machine learning algorithms to regulate user-generated content at scale [19], volunteer community moderators typically use rule-based tools. They customize their programmed conditions to suit their community's needs, which gives full control of the tool's behaviors to the moderators [22]. This helps them apply community-specific norms and transparently explain the tool's malfunction when it happens [22, 29].

These rule-based automated tools monitor posts being uploaded in real-time and remove, hide, tag, and comment on posts based on their programmed conditions [22]. For example, Reddit moderators use AutoModerator, a site-wide rule-based moderation tool that can automate tedious moderation tasks [22, 29]. According to the Reddit Transparency Report 2021, AutoModerator removed 58.9% of the removed Reddit content [39]. Discord moderators use various third-party moderation bots with rule-based moderation functions, such as word filters and user ban lists [27, 28], which more than 18.1M Discord servers use [51].

However, oftentimes a moderation tool may not work as intended – missing posts that the moderators wanted to catch (false negative) or catching the posts that the moderators did not want to catch (false positive). These errors adversely affect the community and require additional work from the moderators. For example, if the automated tool does not immediately remove hateful speech, it can increase the level of emotional stress in the community [43]. On the other hand, if it removes an innocent post, it can cause a backlash from the authors due to being seen as censorship [15, 21]. This may decrease user engagement and cause them to leave the community [2, 12, 26]. To resolve false negatives and false positives, moderators remove or approve posts manually [47], or update their automated rules as an afterthought to prevent future problems [22]. We believe that a better solution would be to predict possible false negatives and false positives *beforehand* so that moderators can minimize the errors of automated rules before their deployment.

To understand the challenges with configuring automated rules, we conducted surveys and a round of interviews with volunteer moderators on Reddit who actively use AutoModerator. From in-depth interviews with five Reddit moderators, we found four main challenges moderators encounter during a typical moderation process: 1) there is no way to estimate the actual effects of a rule in advance, 2) it is hard to detect false positives of a rule after its deployment, 3) it is hard to figure out how the rule should be updated to reduce false positives and false negatives, and 4) it is hard to understand which part of the rule is causing a problem.

Based on the identified challenges, we built ModSandbox, a sandbox system where moderators can test their automated rules by using existing community posts before the actual deployment of rules. ModSandbox has four main features corresponding to the identified challenges: 1) a sandbox to enable prompt configuration evaluation without affecting the actual posts and comments, 2) a recommendation of possible false positive and false negative posts to enable faster discovery, 3) a temporary repository feature to allow users to collect actual false positive or false negative posts to identify the common patterns in them, and 4) a visualization to analyze how the rule affects the posts. ModSandbox uses a machine learning-based sentence encoder to calculate the possibility of false positives and false negatives for each post imported into the system.

We conducted a comparative, between-subject study with 20 active online moderators to assess whether and how ModSandbox helps the configuration process of an automated moderation tool. ModSandbox was able to sort posts in the sandbox for the participants to easily detect false positives and false negatives. Participants using ModSandbox configured more consistent rules than those using the basic system. Also, with ModSandbox, participants wrote more sophisticated rules that can filter target posts precisely. We observed that the participant tried to improve their automated rules with structured and iterative processes using ModSandbox features. Finally, we compared their perceived usefulness scores of ModSandbox and its features according to the types of tasks to highlight their strengths and weaknesses.

We conclude our work by discussing how the proposed design of a system can be improved, potentially facilitate distributed governance for online communities, and reduce cognitive labor in setting up automated moderation tools.

2 RELATED WORK

We focus our review on automated content moderation on social platforms and designing systems for online content moderation. In addition, we provide background information on Reddit's AutoModerator, which we use for our user study evaluation.

2.1 Automated Content Moderation on Social Platforms

There are two levels of content moderation on social platforms: community-level moderation led by users and platform-level moderation led by platform companies [46]. Social platforms such as Meta and Twitter employ paid workers to find and remove content that violates site policies [41]. They focus on policing harmful behaviors such as spreading fake news and hate speech [4, 31], sharing unhealthy tags [8], posting violent or sexual content and using slurs and swear words. Recently, many platforms have adopted machine learning-based systems to automatically manage their content at scale [19]. For example, Facebook uses algorithms to automatically suspend accounts that do not use real names. However, Facebook had to update its algorithm regarding the real name policy because its definition of real name did not include Native Americans who have last names such as "Lone Hill" or "Brown Eyes" [52]. We note that one down side of machine learning-based content moderation is that it lacks context, having the possibility to exclude or disadvantage minor groups or small communities.

Other social platforms such as Reddit and Discord allow voluntary moderators to manage their communities themselves [34]. Typically, these moderators are elected among community members who understand the community norm or are invited by other moderators [47]. Unlike paid workers on large platforms who do not have the authority to decide or change the policy of the platform, voluntary moderators are deeply involved in establishing, determining, and executing their community rules [47]. Although the voluntary moderation opportunity increases the degree of freedom that moderators have in applying the rules for their particular community, it requires moderators to spend a lot of their time and effort on the moderation tasks. As voluntary moderators cannot spend most of their time monitoring their communities, many adopt moderation tools provided by the platform [22], third-party companies [3], and platform users [28]. These tools are mostly rule-based, which allows moderators to directly control how they operate and, if necessary, to transparently communicate with community members on the cause of moderation errors, i.e. false positives and false negatives caused by automated rules [22].

Even if these rule-based moderation tools are more straightforward and flexible than machine learning-based tools, they often do not work as the moderators intended, which requires human moderators to constantly update the configurations to reflect their intention [9]. For example, users can use abbreviations [49], intentional misspellings, and lexical variation [8] of a banned word to avoid automatically being filtered. The moderators then have to update their filter by adding these variations [22]. While these false negatives are dealt with by updating the rules, false positives are more annoying because they require moderators to manually reverse each issue [47]. In this study, we explore the effectiveness of a moderation support system that allows its users to predict false positives and false negatives of a rule-based automated content moderation tool that is applied to the content of their own community so that moderators can improve their rules to prevent future false positives and false negatives.

2.2 Designing a System for Content Moderation

In the context of online content moderation, many studies have introduced machine learning-based classifiers to detect harmful comments and malicious users in the online space. Types of classifier include the detection of cyberbullying [16], profanities and insults [50], pornographic content [48], hate speech [14], and abusive behaviors [11, 36]. As machine learning techniques evolve, recent studies made classifiers multimodal and community-specific, so that the classifiers can reflect each community's preference and culture. For example, Chancellor et al. [7] developed a multimodal classification model to detect images and text that promote eating disorders, which do not fall into the traditional category of harmful content. Furthermore, Chandrasekharan et al. [9] trained macro norms and community-specific to make the classifier more suitable for each community. While previous studies have focused on supervised learning to classify behaviors generally considered harmful, our study adopts an embedding model pretrained by a large language corpus to find comments with few examples that represent the individual moderator's intention. Our system combines the filtering results and their semantic similarities with examples to find possible false positives and false negatives.

Researchers studying data analysis and data visualization have proposed different visual representation approaches to guide the users in their investigation of data. For example, Krause et al. [30] suggested sorting the available combinations of local features of instances by the number of true positives and negatives so that the users can come up with useful hypotheses on how to improve their model. In the domain of algorithmic support for online content moderation, a few studies have proposed to visualize actual content of the community, such as comments to help configure automated rules and support the moderation process. CommentIQ [37] is an interactive visualization tool for online news comment moderators, which helps to find high-quality comments for readers. The user can filter the comments by criteria, location, and times by brushing and linking on their distribution visualization. Also, the system allows the users to reflect their preference to high-quality comments into the sorting order by setting the weights for predefined criteria. Recently, FilterBuddy [23] introduced a tool for YouTube creators to help moderate comments on their videos. The user could customize the word filters to hide or remove comments with specific words in existing filter lists. The system used existing comments to show what and how many comments were filtered, to help evaluate the performance of the filter. In this work, we focus on system design to help community moderators configure a rule-based automated tool that supports combinations of word filters to find posts that violate community rules. Our system shows the expected results of the configured tool using existing posts in a real community and visualizes the relationship between the posts and the configuration to help users analyze each filter.

2.3 Background: Reddit AutoModerator

Reddit AutoModerator is a rule-based automated moderation tool developed by one of the Reddit moderators, Chad Birch, in 2013 [22]. By configuring AutoModerator using YAML, Reddit moderators can create their own automated rules suitable for each subreddit's preference and culture. In 2015, Reddit officially integrated AutoModerator into the platform as a feature of the default moderation tools. According to Reddit transparency reports [39], AutoModerator removed about 103.6M content in 2021, which is 20.9% more than 2020, and 58.9% of all content removed by moderators.

AutoModerator works on all the posts and comments on a subreddit according to the automated rules, which a human moderator last saved in their AutoModerator. In other words, once moderators change their rules, AutoModerator applies the change to newly uploaded content, not the previous ones. Most moderators write multiple automated rules to detect profanity, slurs, and a set of posts that violate specific rules of an individual subreddit. Each rule has one or more checks and actions. The check consists of a field that AutoModerator reviews and a list of keywords, phrases, and regular expressions. The tool verifies whether the fields, such as title and body, include any words and phrases or match with regular expressions in the list. The check also supports verifying content length, the number of user reports, the account age, reputation score, and other features of the Reddit post. To exclude posts with certain conditions from being filtered, the checks can be reversed by putting a tilde notation in front of them. A human moderator can combine multiple checks to fine-tune the rule's scope as the rule filters the posts that satisfy all of the checks, i.e., the intersection of all checks' conditions. The rule also includes actions that indicate the moderation action to perform against the posts identified by the checks.

3 INTERVIEW: CHALLENGES ENCOUNTERED DURING CONFIGURATION PROCESS

To reflect the current practices and challenges of configuring AutoModerator into the design of our system, we conducted semi-structured interviews with five Reddit moderators (Table 1) who have experience configuring AutoModerator. To recruit AutoModerator users among Reddit moderators, we sent online survey links to moderators selected from a list of popular subreddits [1] through the internal Reddit mailing system. A total of 50 moderators answered the online survey that asked for knowledge of how to configure AutoModerator and whether they configured AutoModerator themselves. Then, we sent interview recruitment emails to survey respondents who responded that they have configured AutoModerator by themselves occasionally or most of the time, and left their email addresses for a further in-depth interview.

Each interview session lasted 40-70 minutes through an online conference call and each participant was paid a $30 Amazon gift card for their participation. To extract the challenges of the configuration process from the interview transcriptions, four authors and one assistant participated in an iterative coding process through multiple pairing sessions. The authors were randomly paired for each session and coded an interview transcription. We immediately resolved any disagreement through discussion. After coding all five transcriptions, the authors gathered for four consecutive two-hour meetings to interpret and find patterns in the code and discussed until a consensus on the final codebook was reached on derived themes from the process.

According to interviews, their configuration process to update an automated moderation tool could be divided into two steps: error identification step and rule update step to avoid similar errors in the future. In the following, we describe the four challenges (C1-C4) that online community moderators face in each step.

C1. No way to estimate the actual effects of a rule in advance. When moderators want to discern errors from AutoModerator they configured, they cannot estimate the actual effects of their rules in advance. Participants said that they monitor their community and mod tools such as moderation queue, moderation logs, and Modmail to check for errors that have already occurred in their communities. P3 reported "We're actively going to look at stuff. So for instance, what I do, there's like a mod queue section, which is like reports, automated spam filter, and automoderator." The moderation queue shows the posts or comments reported by users, letting moderators notice the posts AutoModerator missed. The moderation logs and Modmail (internal mailing system for Reddit Moderators) help moderators find false positives by showing the operation history of AutoModerator and user's claim, respectively. P1 said, "As if they like, if they do get moderated and they get removed. I won't, you know, necessarily see them by default unless I go searching in the automoderator log." However, none of them supported checking automated rules in advance before AutoModerator affects community posts. P1 complained that there is no testing protocol to ensure that it works in the real world ("First, I have to use some kind of alt account to like, just essentially make posts and see if AutoModerator catches them"). P1 and P4 reported that they use fake accounts to submit test posts in their community to check the operation of AutoModerator. P4 said, "I have like a just a throwaway account that I'll like, post something real quick." However, they can test AutoModerator with only a few imaginary posts that poorly represent real-world posts. Thus, moderators face difficulty in estimating examples of possible false positives and the actual effect of AutoModerator on the community.

C2. Hard to detect false positives of the rule even after its deployment. Although moderators can search for false positives in Modmail or moderation log, they have difficulty finding false positives through those mod tools. If a user's post is removed without violating any rules, the user can appeal to moderators through Modmail. Then, the moderators can review their removed posts, which allows them to discover an issue with the AutoModerator configuration. P3 described their experience, "When a user response, like a, I think, a mistake has been made, we look at the post, we look at their profile, we look at other variables that the moderation tools give us." Alternatively, moderators can detect an issue while regularly reviewing the moderation logs, where all history of moderators' actions including AutoModerator's is saved. P2 said, "You do have to keep an eye on the moderator queue, you know, you want to have enough monitor." However, Modmail requires users to claim innocent removal of their posts, which inevitably leads to many latent false positives. Furthermore, Reddit does not have an official and individual appeal process for users whose posts are removed by AutoModerator [26], letting users give up appealing the removal due to the inconvenience of the process. On the other hand, checking the moderation log feels inefficient to the moderators. P1 said, "The harder part is always posts that do get moderated as opposed to posts that don't get moderated. [...] I won't, you know, necessarily see them by default unless I go searching in the automoderator log. Then, which I don't really do that often."

C3. Hard to figure out how the rule should be updated. When moderators update automated rules to prevent identified errors, they tend to narrow down the rule by finding additional patterns of target posts for a new rule, check, and strings, which can be difficult for novice moderators. P3 said, "We don't want to remove anything that's not supposed to be removed [...] We try to narrow it (the automated rule) down as much as possible, like, keep it effectively." P2 said "it's just pattern recognition. [...] What's different between the ones that were good, and the ones that are bad? [...] Is there a way that I can write that into a rule, you know, that the automoderator would be able to distinguish? [...] Maybe I wrote a very simple rule, and it's taken down a lot of stuff, you know, but now that I understand the pattern better I can, I can make it a little bit more complex rule, and then have the rule be a little bit more discerning." They tended to make more complex rules to be more

[1] https://www.reddit.com/r/ListOfSubreddits/wiki/listofsubreddits/

No.	Age	Moderator Periods	Gender	AutoModerator Knowledge	Configure AutoModerator?
P1	35-44	6 months	M	Yes, I'm not an expert but I know enough to use in my own sub	Yes, occasionally
P2	35-44	4 years	M	Yes, I'm an expert	Yes, most of the time
P3	35-44	3 years	M	Yes, I'm not an expert but I know enough to use in my own sub	Yes, most of the time
P4	18-24	2 years	M	Yes, I'm an expert	Yes, most of the time
P5	18-24	5 years	M	Well, I think I know a little bit	Yes, most of the time

Table 1: Background Information of Interview Participants

precise. However, this way of thinking requires recalling the errors that moderators identified during the configuration process and finding patterns that can be represented in the form of AutoModerator rules. Therefore, it can be difficult for novice moderators who lack experience with manual moderation and AutoModerator.

C4. Hard to understand which part of the rule is problematic. Moderators reported difficulty in debugging their rules. In our interview, the participants shared how they update automated rules to avoid the recurrence of the same error. Since AutoModerator can work with multiple rules, they first identify a rule that generates the error among the AutoModerator configuration. When they find the rule that catches innocent posts, they tend to eliminate a check or a keyword to avoid further false positives. However, two participants (P1, P4) responded that it is difficult to understand which part of the content triggers which rule and vice versa. P4 said, "There's a few times that it's picked out comments that I can't figure out what's the word. Every so often." In addition, he shared how he uses the action reason feature of AutoModerator. Moderators can write a rule with different action reasons that are displayed with actions in the moderation log to notice which rule was involved in the action. However, this feature does not support highlighting which part of the rule was involved and requires additional labor to manage the reasons in the automated rules.

4 MODSANDBOX: SYSTEM DESIGN

This section describes our system's design goals, which we set to resolve the challenges that are identified from the interviews. We then introduce ModSandbox (Figure 3), a sandbox environment that is built to support online community moderators to easily predict false positives and false negatives and update their automated rules to reduce them.

4.1 Design Goals

We set two high-level goals in designing our ModSandbox system as follows:

- Help moderators quickly find possible false positives and false negatives (in accordance with the error identification step in Figure 2).
- Help moderators configure more sophisticated automated rules to reduce false positives and false negatives (in accordance with the automated rule update step in Figure 2).

For each high-level goal, we present two specific design goals and how they can resolve the four challenges found in Section 3 (see also Figure 2).

4.1.1 DG1. Provide a sandbox to enable prompt configuration evaluation without affecting posts in real communities. According to our surveys and interview study, moderators do not have a way to estimate the results of an updated automated tool in the real world. A sandbox environment can be a solution, which imports real posts from the moderator's community and helps moderators evaluate the automated rules in a simulated environment without affecting posts and comments in their real community.

4.1.2 DG2. Provide a sorting feature to quickly discover false positives and false negatives posts. Interviewees reported that it is hard to recognize false positive posts unless they are reported by users because often they are buried within other posts in moderation logs. To address this, natural language processing (NLP) techniques can be used to measure the similarity between posts and sort them based on the level of similarity so that moderators can quickly spot false positives and false negatives and resolve errors in their rules. Using the semantic similarity between posts to identify false positives and false negatives could complement limitations of keyword-based filtering, e.g., only the posts that the moderators are aware of the keywords can be filtered.

4.1.3 DG3. Provide a space to collect and leverage posts to identify recurring patterns. Providing a space for moderators to collect and leverage posts such as false positives and false negatives would reduce the cognitive load in finding patterns from them. When moderators try to update automated rules, they need to find patterns of recurring errors and reflect them into updated rules. Specifically, they tend to find common features of false positives and negatives they observed during moderation to expand or narrow down the condition of their rules. Thus, we proposed the feature of collecting false positives and negatives to discover their patterns.

4.1.4 DG4. Enable intuitive visual analysis of how the rule affects posts. We suggest providing visual support to help analyze which part of a post caused the automated rule to filter it and how many posts are affected. We found that moderators struggle to recognize the relationship between automated rules and affected posts. Users' struggle with lack of visualization was also reported in a previous study by Jhaver et al. [22], where the authors discussed that visualizing the effect of each rule, such as the number of times each

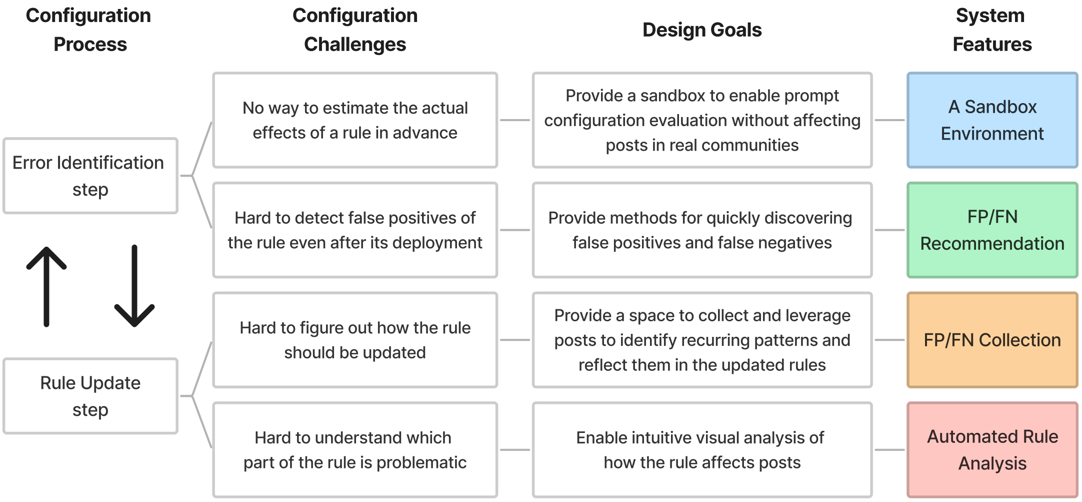

Configuration Process
Configuration Challenges
Design Goals
System Features
Error Identification step
Rule Update step
No way to estimate the actual effects of a rule in advance
Hard to detect false positives of the rule even after its deployment
Hard to figure out how the rule should be updated
Hard to understand which part of the rule is problematic
Provide a sandbox to enable prompt configuration evaluation without affecting posts in real communities
Provide methods for quickly discovering false positives and false negatives
Provide a space to collect and leverage posts to identify recurring patterns and reflect them in the updated rules
Enable intuitive visual analysis of how the rule affects posts
A Sandbox Environment
FP/FN Recommendation
FP/FN Collection
Automated Rule Analysis

Figure 2: A diagram that shows the relationship between the configuration process, challenges, design goals, and system features.

rule has been triggered, may help moderators understand the rule's behavior better.

4.2 Feature 1: A Sandbox Environment

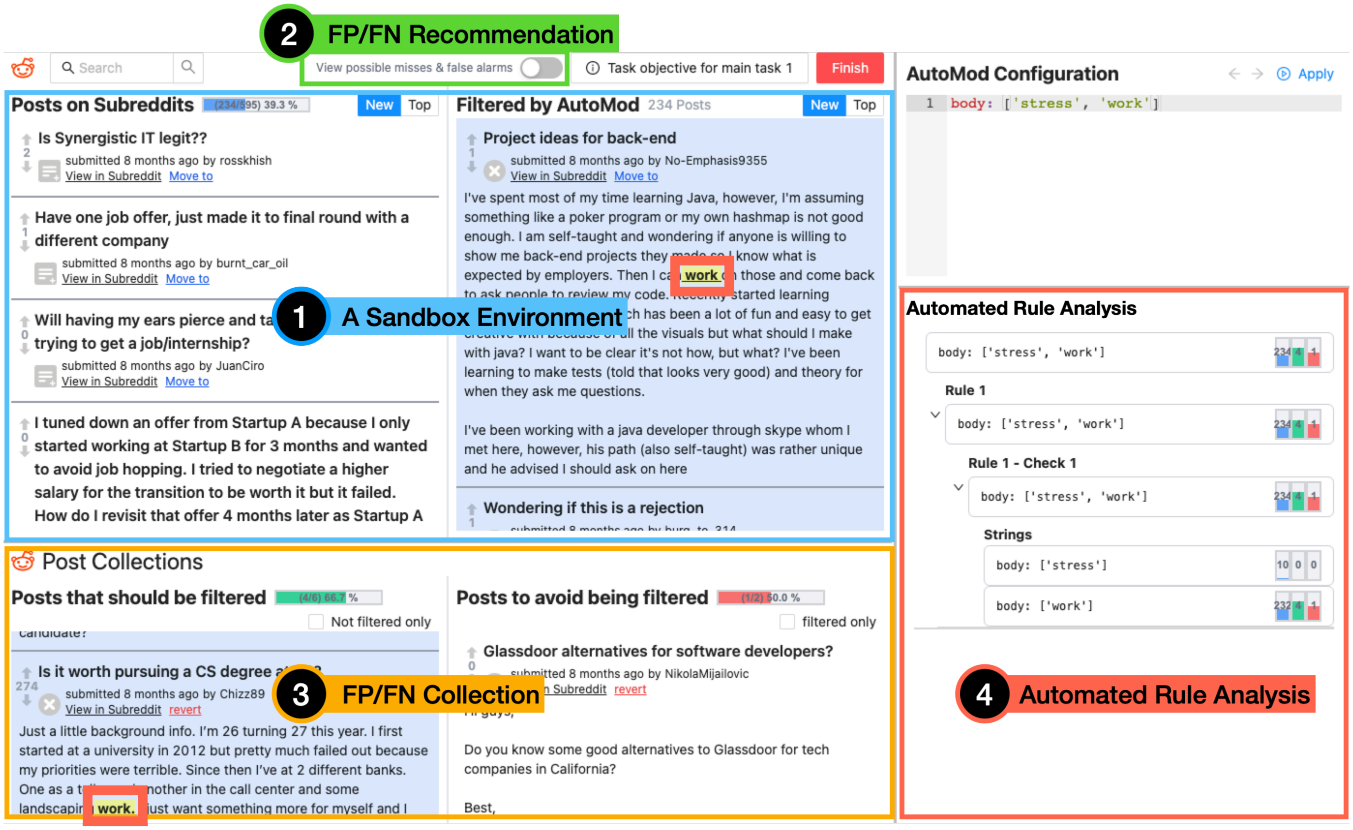
ModSandbox provides moderators with an isolated sandbox environment (❶ in Figure 3), which allows moderators to virtually test their automated rules on posts that already exist in their communities and their moderation logs. The sandbox helps moderators identify or predict any issues with the rules without affecting the posts in their actual communities. Figure 4 shows an example of the use of our sandbox environment. As an example, a moderator imports posts from a subreddit named r/cscareerquestions into the panel "Posts on Subreddits". Then they write an automated rule in the "AutoMod Configuration" panel to filter any post with the words 'IT' and 'engineer'. After they click on the "Apply" button, every post that includes the keywords turns blue to provide a visual comparison between filtered and not-filtered posts. Moderators can also see the filtered posts in the "Filtered by AutoMod" panel, which gathers them in one place for easy browsing. This rearrangement and coloring of posts help moderators understand which types of posts are affected by the rule. Additionally, a horizontal bar right next to the word "Posts on Subreddits" presents the ratio of filtered posts. In the figure, we can observe that more than 70% of the posts include the two keywords. Removing posts with these keywords may be a bad idea because it removes most of the posts in the community. That is, this ratio bar helps moderators understand the effect of automated rules so that they can assess whether the rules work as intended or harm the community.

4.3 Feature 2: FP/FN Recommendation

ModSandbox provides the "FP/FN Recommendation" feature to help moderators quickly find issues with their automated rules, i.e. false positives and false negatives. When moderators activate this feature by toggling a button (❷ View possible misses & false alarms in Figure 3), possible false positives (equal to False Alarms) and false negatives (equal to Misses) are presented in the order of the most probable to the least probable. This feature helps moderators quickly find possible false positives or false negatives without having to browse all the posts in the sandbox.

Semantic similarity is often used to detect spam posts or harassment in online communities because it complements a common failure of keyword-based filtering [1, 32, 42, 44]. Keyword-based filtering may struggle to collect all the offensive posts if they do not have any matching keywords, while semantic similarity approaches can do. For example, if a malicious user tries to insult a community member without explicitly including their username in the post, keyword-based filtering may not be able to detect the insult if it only works based on the username. However, semantic similarity approaches may be able to detect the post based on its content. Motivated by previous work, we propose comparing the semantic similarities between posts to identify false positive and false negative posts. If a filtered post is semantically far from the posts that the moderators want to filter, i.e., far from the target posts, it is likely to be a false positive. In this work, we treat posts that are filtered but are different from posts in "Posts that should be filtered" as *possible false positives*. This panel is part of the "FP/FN Collection" feature described in Section 4.4. On the contrary, we treated posts that are not filtered, but are similar to posts in "Posts that should be filtered", as *possible false negatives*. For example, as shown in Figure 5, a non-filtered post with the closest distance from the posts in "Posts that should be filtered" (the reference point) comes at the top of the "Possible Misses" panel. The farthest non-filtered post from the reference point comes at the bottom of this panel. As a result, this feature lets moderators see more critical posts first and helps them quickly find clues to update their automated rules to filter these missed posts.

For the calculation of post similarity, we adopt a sentence embedding model to encode semantic features of each post into embedding vectors. The pretrained sentence-level embedding outperforms word-level embedding in various transfer tasks [13]. Also,



```
body: ['stress', 'work']
```

Is Synergistic IT legit??
2
submitted 8 months ago by rosskhish

Have one job offer, just made it to final round with a different company
1
submitted 8 months ago by burnt_car_oil

Will having my ears pierce and ta trying to get a job/internship?
0
submitted 8 months ago by JuanCiro

I tuned down an offer from Startup A because I only started working at Startup B for 3 months and wanted to avoid job hopping. I tried to negotiate a higher salary for the transition to be worth it but it failed. How do I revisit that offer 4 months later as Startup A
0

Project ideas for back-end
1
submitted 8 months ago by No-Emphasis9355

I've spent most of my time learning Java, however, I'm assuming something like a poker program or my own hashmap is not good enough. I am self-taught and wondering if anyone is willing to show me back-end projects they made so I know what is expected by employers. Then I can work on those and come back to ask people to review my code. Recently started learning ch has been a lot of fun and easy to get all the visuals but what should I make with java? I want to be clear it's not how, but what? I've been learning to make tests (told that looks very good) and theory for when they ask me questions.

I've been working with a java developer through skype whom I met here, however, his path (also self-taught) was rather unique and he advised I should ask on here

Wondering if this is a rejection
1
submitted 8 months ago by burg_to_314

candidate?

Is it worth pursuing a CS degree a
274
submitted 8 months ago by Chizz89

Just a little background info. I'm 26 turning 27 this year. I first started at a university in 2012 but pretty much failed out because my priorities were terrible. Since then I've at 2 different banks. One as a t nother in the call center and some landscapi work. just want something more for myself and I

Glassdoor alternatives for software developers?
0
submitted 8 months ago by NikolaMijailovic

Hi guys,

Do you know some good alternatives to Glassdoor for tech companies in California?

Best,

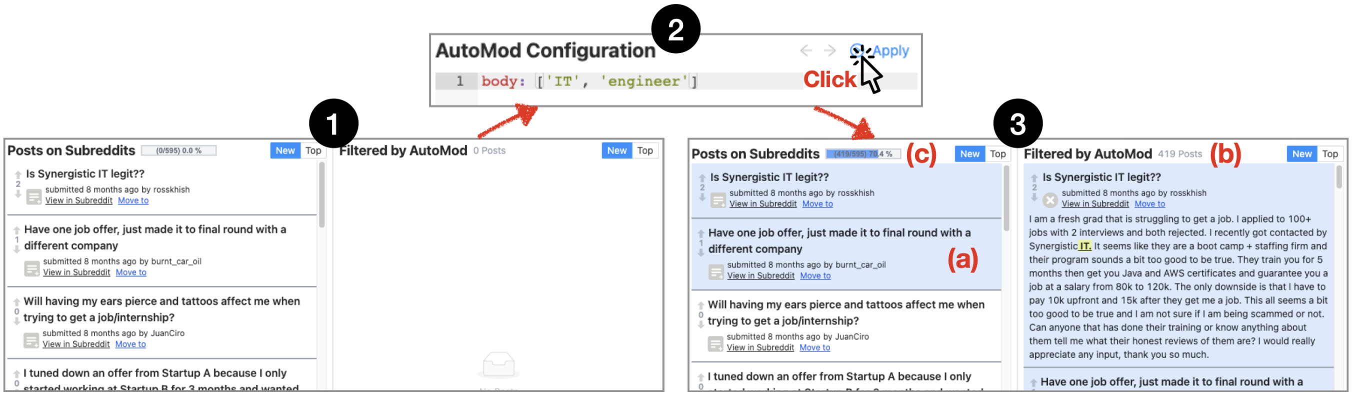
body: ['stress', 'work']	234	4	1
Rule 1			
body: ['stress', 'work']	234	4	1
Rule 1 - Check 1			
body: ['stress', 'work']	234	4	1
Strings			
body: ['stress']	10	0	0
body: ['work']	232	4	1

Figure 3: An overview of the four main features of ModSandbox. ❶ is a "Sandbox Environment" where a moderator can import all the posts from their community. ❷ is a toggle button that rearranges the posts in the sandbox area from the most "Possible misses and false alarms" to the least. It helps moderators to more quickly find possible misses (false negatives) and false alarms (false positives). ❸ is the "FP/FN Collection" area that helps moderators to collect interesting posts to find their patterns for further rule updates. ❹ is the "Configuration Analysis" panel that helps analyze how the rule affected the posts in the sandbox. It shows the number of filtered posts in "Sandbox Environment" and "Post Collections" (FP/FN Collection) with color bars and highlights the part of those filtered posts in their panels (red boxes in ❶, ❷) for macro and micro-level support of debugging each configuration.

the sentence embedding can complement the limitation of AutoModerator's word filtering by considering the context of the post to find false positives and negatives. When moderators import their community posts, our system computes and saves an embedding vector for each post using Universal Sentence Encoder [6], one of the popular open-source sentence embedding models. Then, it computes the cosine similarities between the saved vectors and the average vector of the posts in the "posts that should be filtered" and sorts the possible misses and false alarms in the order of similarity. Although the vector encoding step requires a high computation cost proportional to the number of posts, this is a one-time computation that only occurs after moderator imports posts from their community. The time to calculate the cosine similarities is also proportional to the number of posts in the system, but the calculation is much faster because it does not require deep models.

4.4 Feature 3: FP/FN Collection

The "FP/FN Collection" panel (❸ Post Collections in Figure 3) enables moderators to collect posts that are useful for evaluating their rules, such as *posts that should be filtered* (identified false negatives) or *posts to avoid being filtered* (identified false positives) Figure 6 shows an example of using the FP/FN Collection. A moderator can move the posts they want to filter with automated moderation to the "Post that should be filtered" panel ((a) in Figure 6). If the community members are active at reporting the posts, moderators can put the reported posts right into the panel. Once enough posts are collected, the moderator can use this panel in two ways. First, they can browse through the posts to find patterns that could be useful to write an automated rule, e.g., find common keywords among the posts collected. Second, they can see a green bar to see the percentage of collected posts that are being filtered by the current automated rule ((b) in Figure 6). If the number of posts being filtered is too low, they may want to update the automated rule to filter more posts.



```
1 body: ['IT', 'engineer']
```

Is Synergistic IT legit??
submitted 8 months ago by rosskhish
View in Subreddit Move to

Have one job offer, just made it to final round with a different company
submitted 8 months ago by burnt_car_oil
View in Subreddit Move to

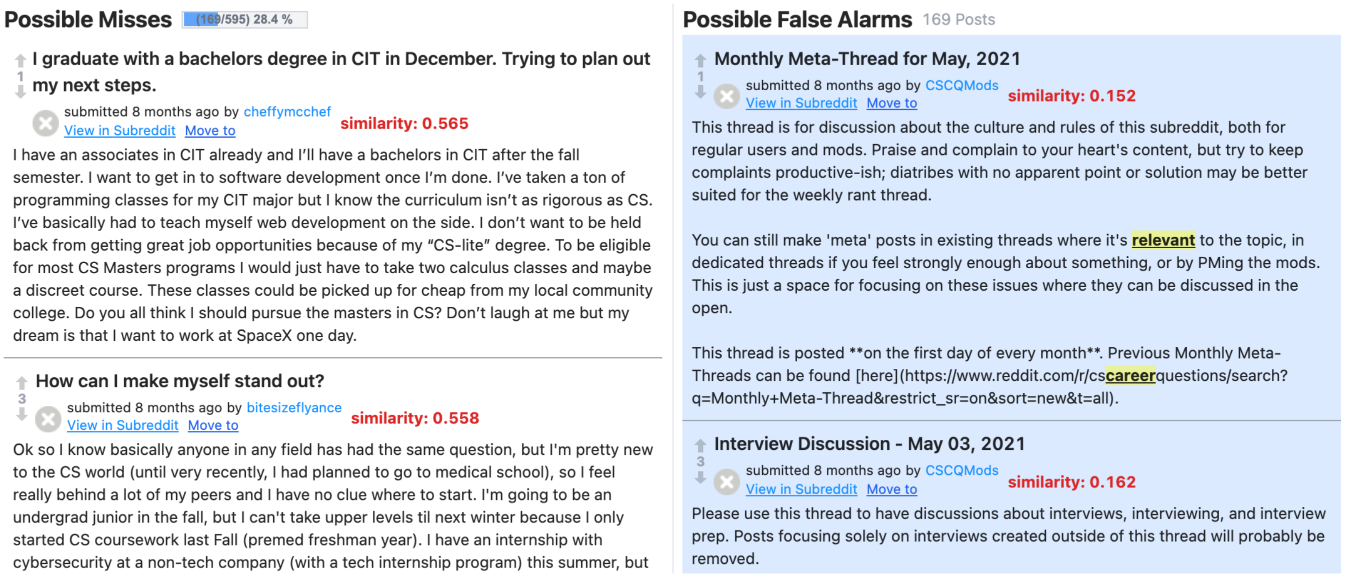
Will having my ears pierce and tattoos affect me when trying to get a job/internship?
submitted 8 months ago by JuanCiro
View in Subreddit Move to

I tuned down an offer from Startup A because I only started working at Startup B for 3 months and wanted

Is Synergistic IT legit??
submitted 8 months ago by rosskhish
View in Subreddit Move to
I am a fresh grad that is struggling to get a job. I applied to 100+ jobs with 2 interviews and both rejected. I recently got contacted by Synergistic IT. It seems like they are a boot camp + staffing firm and their program sounds a bit too good to be true. They train you for 5 months then get you Java and AWS certificates and guarantee you a job at a salary from 80k to 120k. The only downside is that I have to pay 10k upfront and 15k after they get me a job. This all seems a bit too good to be true and I am not sure if I am being scammed or not. Can anyone that has done their training or know anything about them tell me what their honest reviews of them are? I would really appreciate any input, thank you so much.

Have one job offer, just made it to final round with a

Figure 4: Show how to use a Sandbox Environment. ❶ shows the sandbox right after importing posts from a community. When a user clicks on the "Apply" button after writing their rules in the ❷ "AutoMod Configuration" panel, ❸-(a) the background turns blue for the posts filtered by the rules, ❸-(b) "Filtered by AutoMod" gathers them in a separate panel for easy browsing, and ❸-(c) a blue bar graph shows the ratio of filtered posts to imported posts.



I graduate with a bachelors degree in CIT in December. Trying to plan out my next steps.
submitted 8 months ago by cheffymcchef
View in Subreddit Move to
I have an associates in CIT already and I'll have a bachelors in CIT after the fall semester. I want to get in to software development once I'm done. I've taken a ton of programming classes for my CIT major but I know the curriculum isn't as rigorous as CS. I've basically had to teach myself web development on the side. I don't want to be held back from getting great job opportunities because of my "CS-lite" degree. To be eligible for most CS Masters programs I would just have to take two calculus classes and maybe a discreet course. These classes could be picked up for cheap from my local community college. Do you all think I should pursue the masters in CS? Don't laugh at me but my dream is that I want to work at SpaceX one day.

How can I make myself stand out?
submitted 8 months ago by bitesizeflyance
View in Subreddit Move to
Ok so I know basically anyone in any field has had the same question, but I'm pretty new to the CS world (until very recently, I had planned to go to medical school), so I feel really behind a lot of my peers and I have no clue where to start. I'm going to be an undergrad junior in the fall, but I can't take upper levels til next winter because I only started CS coursework last Fall (premed freshman year). I have an internship with cybersecurity at a non-tech company (with a tech internship program) this summer, but

Monthly Meta-Thread for May, 2021
submitted 8 months ago by CSCQMods
View in Subreddit Move to
This thread is for discussion about the culture and rules of this subreddit, both for regular users and mods. Praise and complain to your heart's content, but try to keep complaints productive-ish; diatribes with no apparent point or solution may be better suited for the weekly rant thread.

You can still make 'meta' posts in existing threads where it's relevant to the topic, in dedicated threads if you feel strongly enough about something, or by PMing the mods. This is just a space for focusing on these issues where they can be discussed in the open.

This thread is posted **on the first day of every month**. Previous Monthly Meta-Threads can be found [here](https://www.reddit.com/r/cscareerquestions/search?q=Monthly+Meta-Thread&restrict_sr=on&sort=new&t=all).

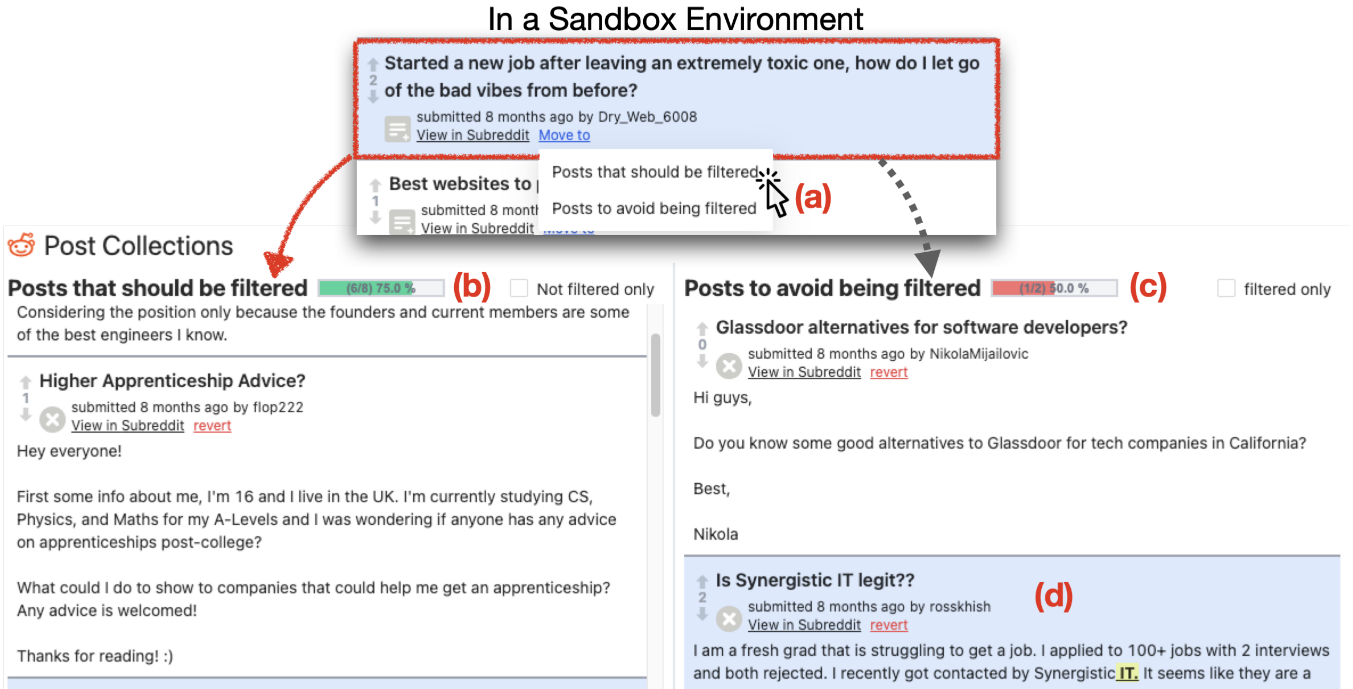
Interview Discussion - May 03, 2021
submitted 8 months ago by CSCQMods
View in Subreddit Move to
Please use this thread to have discussions about interviews, interviewing, and interview prep. Posts focusing solely on interviews created outside of this thread will probably be removed.

Figure 5: Example of possible misses (false negatives) and false alarms (false positives) of the configured rules in Task 2 of our main user study. Participants were guided to detect posts about asking whether or how to get CS-relevant jobs without CS-relevant degrees. The more probable posts that are being missed are listed at the top (e.g., similarity 0.565 is larger than 0.558), and the opposite happens for the false alarms (similarity 0.152 is smaller than 0.162). The similarity values are hidden in the actual interface.

A similar practice could be applied to using the "Post to avoid being filtered" panel. A moderator can collect posts that should not be filtered in this panel to find common patterns among them. Then they can write an automated rule that would avoid filtering these posts. The moderator can monitor the red bar ((c) in Figure 6) in this panel to see if the current rule is successfully avoiding filtering posts in this panel. For example, in this figure, since 50% of the posts in this panel are being filtered, the moderator might want to edit their automated rule to reduce this number.

4.5 Feature 4: Automated Rule Analysis (DG4)

ModSandbox helps moderators analyze the impact of their complex automated rules through the features of "Automated Rule Analysis" (❹ in Figure 3). First, "Automated Rule Analysis" panel shows rules in a hierarchical structure, allowing moderators to easily analyze them one by one. Similar to other automated rule generators, AutoModerator supports multiple rules, and each rule has one or more checks. A check is a line of code that represents a single condition for filtering the posts. It consists of an attribute of the posts, such as body or title, and a single list of strings. For example, *body: ['red', 'blue']* makes a rule to catch the posts with body that



Started a new job after leaving an extremely toxic one, how do I let go of the bad vibes from before?
2
submitted 8 months ago by Dry_Web_6008

Best websites to
1
submitted 8 mont

Posts that should be filtered (6/8) 75.0 %

Considering the position only because the founders and current members are some of the best engineers I know.

Higher Apprenticeship Advice?
1
submitted 8 months ago by flop222

Hey everyone!

First some info about me, I'm 16 and I live in the UK. I'm currently studying CS, Physics, and Maths for my A-Levels and I was wondering if anyone has any advice on apprenticeships post-college?

What could I do to show to companies that could help me get an apprenticeship? Any advice is welcomed!

Thanks for reading! :)

Posts to avoid being filtered (1/2) 50.0 %

Glassdoor alternatives for software developers?
0
submitted 8 months ago by NikolaMijailovic

Hi guys,

Do you know some good alternatives to Glassdoor for tech companies in California?

Best,

Nikola

Is Synergistic IT legit??
2
submitted 8 months ago by rosskhish

I am a fresh grad that is struggling to get a job. I applied to 100+ jobs with 2 interviews and both rejected. I recently got contacted by Synergistic IT. It seems like they are a

Figure 6: Show how to use FP/FN Collection. (a) The users can move posts from the Sandbox Environment to one of the Post Collections panels: "Posts that should be filtered (red solid arrow)" and "Posts to avoid being filtered (gray dashed arrow)". (b, c) The green and red bars show the ratio of the filtered ones. (d) The filtered posts by the automated rules are marked blue in the Post Collections panel.

includes 'red' or 'blue'. Using the "Configuration Analysis" feature, moderators can assess the impact of each rule, check, and string individually. The three bar graphs in blue, green and red on the right panel of Figure 7 indicate how each part of the rules affects posts in "Posts on Subreddits", "Posts that should be filtered" and "Posts to avoid being filtered", respectively. Looking at the three bar graphs in Figure 7(d), the rule 1 filter more posts in the "Posts to avoid being filtered (red)" than the posts in "Posts that should be filtered (green)". In this case, the moderator can expand Rule 1 for a deeper analysis by clicking on it and find that the keyword 'work' in Rule 1 ((e)in Figure 7) is the one causing a lot of unwanted posts to be filtered. Then, they may choose to remove or update that keyword to reduce false positives.

"Automated Rule Analysis" also presents quick highlights for all filtered posts showing which part of the post is being affected by the automated rules (e.g., the word in the post that triggers the AutoModerator) and which part of the rule is being triggered by the post (e.g., the keyword that was triggered in the filter). This feature helps moderators quickly and easily find the reason for false positives from automated rules. For example, if an automated rule is set to filter posts containing the word "work" in the body, then in every post that contains the word "work", the word "work" is highlighted in yellow (see Figure 7). Reversely, when a user hovers a cursor over one of the highlighted words in a post, the system highlights the triggered rule, check, and string so that the human moderator can understand which part of the rule is related to filtering the post.

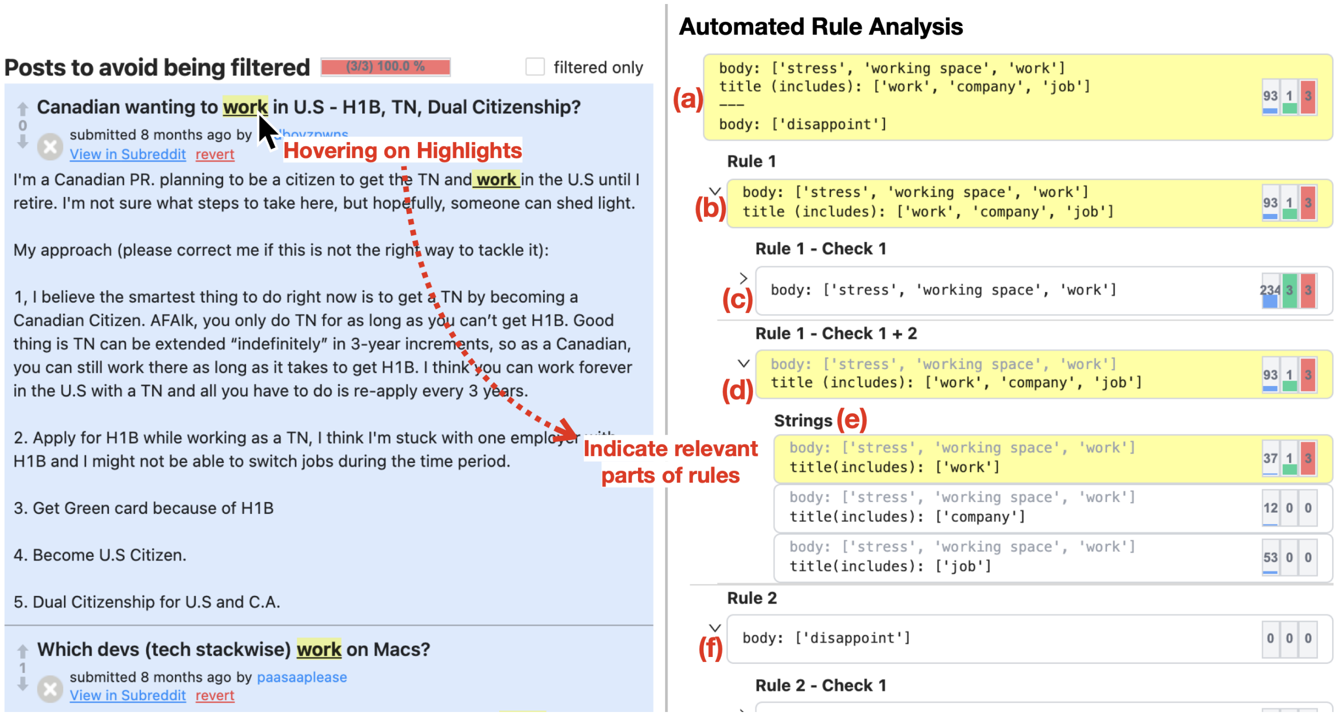
5 USER STUDY

To observe how ModSandbox can improve the AutoModerator configuration process, we conducted a controlled between-subject user study with 20 active moderators of the online community through Zoom [2]. We divided the participants into an experimental group and a control group, where the control group was added to ensure that any benefits observed when using ModSandbox is not just coming from nudging participants to repeat the task. The experimental group used a basic system first and then ModSandbox, and the control group used the basic system first and then the basic system again.

The basic system (Figure 8) was built to simulate a general process of creating Reddit AutoModerator rules. With the basic system, users can do the typical things they would do when moderating their subreddits: browse community posts, search posts by words or phrases, and sort posts by newest and highest votes. Then we built ModSandbox by adding the features proposed in Section 4 to the basic system.

Five hundred and ninety-five posts to be used in the user study were crawled from a subreddit named r/cscareerquestions from May 1st 2021 to May 7th 2021, a subreddit where members post questions about computer science careers. Our criteria to select a

[2] https://zoom.us/



Posts to avoid being filtered (3/3) 100.0 % □ filtered only

Canadian wanting to work in U.S - H1B, TN, Dual Citizenship?

0

submitted 8 months ago by

View in Subreddit revert

I'm a Canadian PR. planning to be a citizen to get the TN and work in the U.S until I retire. I'm not sure what steps to take here, but hopefully, someone can shed light.

My approach (please correct me if this is not the right way to tackle it):

1, I believe the smartest thing to do right now is to get a TN by becoming a Canadian Citizen. AFAIk, you only do TN for as long as you can't get H1B. Good thing is TN can be extended "indefinitely" in 3-year increments, so as a Canadian, you can still work there as long as it takes to get H1B. I think you can work forever in the U.S with a TN and all you have to do is re-apply every 3 years.

2. Apply for H1B while working as a TN, I think I'm stuck with one employer with H1B and I might not be able to switch jobs during the time period.

3. Get Green card because of H1B

4. Become U.S Citizen.

5. Dual Citizenship for U.S and C.A.

Which devs (tech stackwise) work on Macs?

1

submitted 8 months ago by paasaaplease

View in Subreddit revert

Automated Rule Analysis

(a)

```
body: ['stress', 'working space', 'work']
title (includes): ['work', 'company', 'job']
---
body: ['disappoint']
```

93 1 3

Rule 1

(b)

```
body: ['stress', 'working space', 'work']
title (includes): ['work', 'company', 'job']
```

93 1 3

Rule 1 - Check 1

(c)

```
body: ['stress', 'working space', 'work']
```

234 3 3

Rule 1 - Check 1 + 2

(d)

```
body: ['stress', 'working space', 'work']
title (includes): ['work', 'company', 'job']
```

93 1 3

Strings (e)

```
body: ['stress', 'working space', 'work']
title(includes): ['work']
```

37 1 3

```
body: ['stress', 'working space', 'work']
title(includes): ['company']
```

12 0 0

```
body: ['stress', 'working space', 'work']
title(includes): ['job']
```

53 0 0

Rule 2

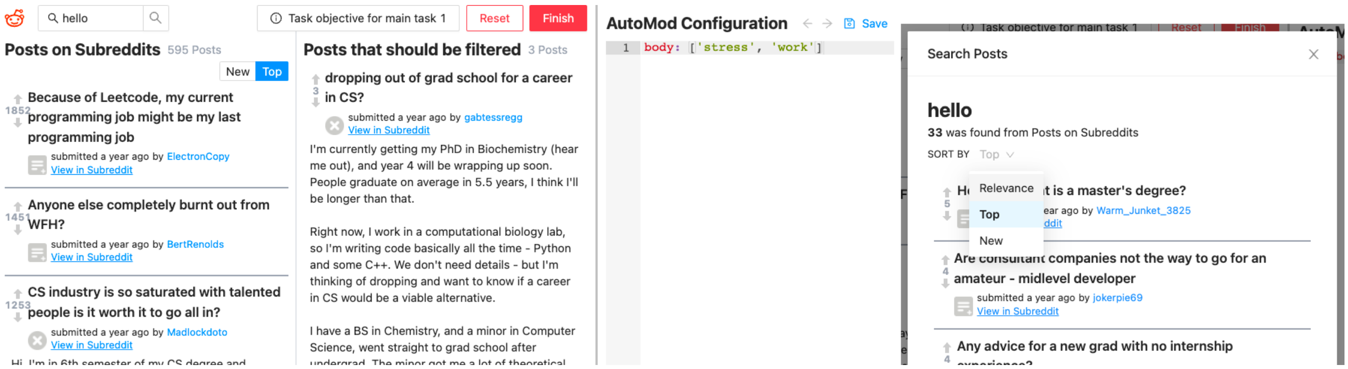
(f)

```
body: ['disappoint']
```

0 0 0

Rule 2 - Check 1

Figure 7: An example of the Automated Rule Analysis feature. Each labeled box with rounded corners on the right side represents a part of a configured rule. The three embedded vertical bar graphs on the right side of each rounded box show the number and ratio of filtered posts in three different types of posts: "Posts on Subreddits", "Posts that should be filtered", and "Posts to avoid being filtered", respectively.(a) shows an AutoModerator configuration that consists of multiple rules: (b) and (f). Rule (b) detects intersection of posts detected by checks (c) and (d). Check (c) finds the posts that have any of 'stress', 'working space', and 'work' in the body. Among the posts detected by check (c), check (d) detects the posts that includes any of 'work', 'company', and 'job' on the title. (e) shows the impact of individual strings in the check (d). The Highlight feature emphasizes specific part of posts affected by the configuration. As shown in the left side of the figure, when a user hovers the cursor on the word "work" in the post title, relevant items (the rounded boxes) are highlighted on Configuration Analysis panel. In this case, the first string in (e) got involved in the detection of "work" in the title. Thus, the system highlights the check (d), rule (b), and whole configuration (a) that includes the first string in (e)



Posts on Subreddits 595 Posts

New Top

Because of Leetcode, my current programming job might be my last programming job

1852

submitted a year ago by ElectronCopy

View in Subreddit

Anyone else completely burnt out from WFH?

1451

submitted a year ago by BertRenolds

View in Subreddit

CS industry is so saturated with talented people is it worth it to go all in?

1253

submitted a year ago by Madlockdoto

View in Subreddit

Posts that should be filtered 3 Posts

dropping out of grad school for a career in CS?

3

submitted a year ago by gabtessregg

View in Subreddit

I'm currently getting my PhD in Biochemistry (hear me out), and year 4 will be wrapping up soon. People graduate on average in 5.5 years, I think I'll be longer than that.

Right now, I work in a computational biology lab, so I'm writing code basically all the time - Python and some C++. We don't need details - but I'm thinking of dropping and want to know if a career in CS would be a viable alternative.

I have a BS in Chemistry, and a minor in Computer Science, went straight to grad school after

AutoMod Configuration ← → Save

```
body: ['stress', 'work']
```

Search Posts ×

hello

33 was found from Posts on Subreddits

SORT BY Top

Relevance
Top
New

...t is a master's degree?

5

...ear ago by Warm_Junket_3825

Are consultant companies not the way to go for an amateur - midlevel developer

4

submitted a year ago by jokerpie69

View in Subreddit

Any advice for a new grad with no internship

4

Figure 8: An overview of a basic system for our user study. The system provides similar features that moderators have during the moderation process in Reddit. The participants can see the community posts in "Posts on Subreddits", sort them by New & Top, see the example target posts in "Posts that should be filtered", and search posts by words or phrases with the pop-up window on the right side.

subreddit included (1) whether the community is active, (2) whether posts are mostly text-based as our scope focuses on keyword-based moderation, and (3) whether it is easy to make a plausible hypothetical community rule to use for the user studies.

Through the user study, we aimed to evaluate whether ModSandbox could help moderators identify the possible errors of AutoModerator and improve their automated rules. Furthermore, our goal was to analyze their process, perceived usefulness, and feedback to propose the direction for the future system that supports the automated moderation tool configuration process.

- RQ1: Can ModSandbox support the configuration process of the automated moderation tool?
  - RQ1-1: Can ModSandbox support moderators to detect false positives and false negatives more easily?
  - RQ1-2: Can ModSandbox help moderators update automated rules to reduce false positives and false negatives?
- RQ2: How does ModSandbox support the configuration process of the automated moderation tool?
  - RQ2-1: How do participants use the features of ModSandbox for the configuration process?
  - RQ2-2: How do participants perceive the usefulness of ModSandbox in the configuration process?

5.1 Participants

We recruited 10 participants for an experimental group and 10 participants for a control group (Table 2). The experimental group used a basic system first and then our ModSandbox, while the control group used the basic system for both of the trials. The purpose of the comparative between-subject design was to verify whether ModSandbox actually helps moderators *improve* their rule in the second trial. If a rule can be improved by giving them a second chance to refine it, then the rule will also be enhanced in the control group where the basic system is repeated.

More details of the experimental procedures are described in Section 5.2. In the experimental group, seven Reddit moderators (five males and two females) were from the United States. The other three were non-Reddit moderators (three females) and were in charge of Korean online communities on Facebook Groups. These three were proficient in English, thus participated in the English-based user study as the U.S. participants did. Participants in the control group were all Reddit moderators from Asia, Europe, and the United States. We sent a recruitment advertisement to Reddit moderators through mod mail, which is a message system within the Reddit platform. We contacted moderators of subreddits randomly sampled from the same list for our interview recruitment and excluded the interview participants. The non-Reddit moderators in South Korea were recruited by word of mouth. We expected the non-Reddit moderators to represent voluntary moderators outside Reddit. Although they may not be familiar with Reddit, we confirm their moderation practices and challenges aligned with those on Reddit, while they might have unique moderation experiences.

Additionally, we ensure that we have moderators both with and without experience using AutoModerator. Half of the participants (P1, P5, P7, P8, P10 in the experimental group; P12, P14, P16, P19, P20 in the control group) had experience configuring AutoModerator while the others (P2, P3, P4, P6, P9 in the experimental group; P11, P13, P15, P17, P18 in the control group), including Korean community moderators, had little or no AutoModerator experience. The recruitment method and the study design were approved by our institution's IRB policy.

5.2 Study Procedure

Each study lasted about two hours, and each participant received a $30 Amazon gift card per hour or 30,000 KRW per hour as compensation. Before the study, the participants filled out a consent form and answered their background information in Table 2

5.2.1 *Tutorial on How to configure AutoModerator (20-30 minutes).*

Before entering the main task, we explained the process of user study and how to write AutoModerator rules that are available in the user study. The available rules were restricted to detecting or excluding specific words and phrases in the title or body. For the experimental group, we gave an additional walk-through tutorial on how to use the features of ModSandbox.

After the tutorial session, we asked participants to solve quizzes about the study to ensure that they understood how to configure AutoModerator. If they got incorrect answers, we helped them find the right answer and then checked if they understood correctly. This step ensured that everyone had rule authoring skills that were sufficient to perform the main tasks. We also provided them with two reference documentation: the description of ModSandbox features (only for the experimental group) and the AutoModerator rule syntax, which was freely accessible during the main tasks.

5.2.2 *Main Tasks (60-80 minutes).*

Each participant was given two different tasks where they write the AutoModerator rules for a given hypothetical moderation scenario. Moderation according to the actual rules can expose users to mentally abusive posts including slurs and swear words. Thus, we created a novel moderation objective instead of using the subreddit's actual rules.

The two main scenarios that we showed to the participants were as follows:

- Task A: Many people without CS relevant degrees post questions asking whether or how to get CS relevant jobs on r/cscareerquestions. Because r/cscareerquestions has a FAQ page that contains answers to those questions, moderators want to configure AutoMod to automatically leave a comment with a link to the FAQ page on posts asking whether and how to get CS-relevant jobs without the related degrees.
  - **Objective:** Write AutoMod rules to detect posts asking whether or how to get CS-relevant jobs without CS-relevant degrees.
- Task B: The moderators of r/cscareerquestions want to leave a comment saying "Your post includes keywords related to Covid-19. If you need any help with the current global pandemic situation related to medical, mental, or economical crisis, please contact xxx for further information." on posts relevant to "covid".
  - **Objective:** Write AutoMod rules to detect the posts that the moderator should leave comments according to the above.

No.	Condition	Age	Gender	Moderator periods	Prior experiene		
					Platform	Programming	AutoModerator
P1	Experimental	35-44	M	over 5 years	Reddit	basic concepts	Experienced
P2	Experimental	18-24	F	6 months - 1 year	Facebook	basic concepts	Novice
P3	Experimental	25-34	F	under 6 months	Facebook	No knowledge	Novice
P4	Experimental	18-24	F	6 months - 1 year	Facebook	frequently	Novice
P5	Experimental	25-34	F	1 - 2 years	Reddit	basic concepts	Experienced
P6	Experimental	25-34	M	under 6 months	Reddit	No knowledge	Novice
P7	Experimental	25-34	M	2 - 3 years	Reddit	frequently	Experienced
P8	Experimental	18-24	M	2 - 3 years	Reddit	a few programs	Experienced
P9	Experimental	45-54	M	6 months - 1 year	Reddit	a few programs	Novice
P10	Experimental	18-24	M	1 - 2 years	Reddit	a few programs	Experienced
P11	Control	25-34	M	1 - 2 years	Reddit	No knowledge	Novice
P12	Control	45-54	X	over 5 years	Reddit	a few programs	Experienced
P13	Control	45-54	M	over 5 years	Reddit	basic concepts	Novice
P14	Control	35-44	F	over 5 years	Reddit	a few programs	Experienced
P15	Control	25-34	F	3 - 4 years	Reddit	basic concepts	Novice
P16	Control	25-34	M	1 - 2 years	Reddit	frequently	Experienced
P17	Control	18-24	M	under 6 months	Reddit	No knowledge	Novice
P18	Control	25-34	M	over 5 years	Reddit	No knowledge	Novice
P19	Control	18-24	M	2 - 3 years	Reddit	basic concepts	Experienced
P20	Control	25-34	F	6 months - 1 year	Reddit	frequently	Experienced

Table 2: Background information of study participants. Experienced participants in Prior experience (with) AutoModerator column are moderators who configure AutoModerator occasionally or most of the time by themselves. P2, P3, and P4 are moderators of Korean communities on different Facebook Groups. P12 preferred not to say their gender.

These two tasks represent two different scenarios of content moderation in online communities. The first task (CS-relevant degrees) represents a more community-specific moderation scenario, where the rule only applies to the specific community alone. This scenario also represents the cases where the targeted posts have semantically similar content, which makes it easier for natural language processing models to work. The second task (COVID-19) represents a more general scenario in which unexpected external events affect the community.

For each task, we provided three target example posts as samples that meet each moderation objective. The participants were informed that those three example posts had already been manually filtered by other virtual peer moderators. Because moderation task is somewhat subjective, we expect that the given example posts would help participants have similar criteria on how they evaluate whether a post should be moderated or not. The authors selected this type of example posts from the posts that two external annotators regarded as targeted for the moderation objective. This process is further explained in Section 5.3.1.

Each participant in the experimental group first used the basic system to draft automated rules and then moved on to ModSandbox to improve the rules using the given features of the system. On the other hand, the same basic system was offered twice for the control group. Task A and B were offered in a randomized order for each participant. To ensure that participants have reasonable rules to start with, we emphasized that the rules written in the basic system should be in the form of their best attempt in both groups. Their monitor screens were shared and recorded with their consent to analyze how participants used the systems during the main task.

5.2.3 Post Surveys (10-20 minutes). After the main tasks for the experimental group, the participants took part in a survey about their experience. They answered a 7-point Likert scale and open-ended questions on how useful the features of ModSandbox were in each main task and the overall usefulness of ModSandbox. We also asked them about their strategies using ModSandbox and feedback on how the system could be improved.

5.3 Measures

To answer the research questions, we observe the following:

The distribution of target posts to be filtered within other posts under different sorting conditions (RQ1-1). This shows how much more effective "FP/FN Recommendation" (FP/FN) is than sorting by the newest and highest votes (NEW and TOP). We expected our system to help users find false positives and false negatives more easily by showing more probable false positives and negatives on top. Therefore, we visualized the cumulative numbers of target posts using different sorting methods. For comparison, we created a set of ground-truth (GT) target posts to be filtered for the two main tasks. The detailed procedure for obtaining these GT target posts is described in Section 5.3.1.

The average complexity of the automated rules using ModSandbox. (RQ1-2) This shows how ModSandbox can help participants build more sophisticated rules to reflect their moderation intention and avoid false positives and false negatives. We compared the number of rules, checks, and strings they wrote in the basic system and ModSandbox. The number of rules can describe how many subgoals they considered for a given moderation scenario. The number of checks and strings can represent how accurately the rules catch the posts that the participants intended.

The semantic similarities between the example posts and filtered posts. (RQ1-2) Since we provided example posts that represent each moderation goal, we can use the semantic similarities of the posts with the example posts to represent how each post is semantically close to false positives and false negatives. For example, if the filtered posts are semantically far from the example posts, they are likely to be false positives (or vice versa). Using this metric, we compare the similarities between example posts and filtered posts in the basic system and ModSandbox. If the system helps reduce false positives, their distribution will increase. We applied the algorithm that was used in "FP/FN Recommendation" feature to calculate semantic similarities.

The consistency of filtered posts among moderators (RQ1-2). Since we gave clear and concrete task objectives, we can measure the consistency of filtered posts among participants for each condition. It represents whether the system helps write automated rules that can catch the posts in which the majority of them agree to accord with the user study task goal. We note that consistency gauges how much they have reached an agreement, not how accurate their rules are. We used Fleiss' Kappa [18], a statistical measure of inter-rater reliability among multiple raters, to measure the consistency among ten moderators in control and experimental groups, respectively.

System usage patterns of participants and answers to a rule-making strategies (RQ2-1) Two authors reviewed screen recordings to observe how participants use a basic system and ModSandbox and found patterns of using ModSandbox features together to improve their automated rules. Also, we asked their own rule-making strategies while using ModSandbox through the post surveys.

Perceived usefulness of each features (RQ2-2) We calculated the average usefulness score of ModSandbox and its features for each task. We then analyze the answers to open-ended questions to understand why they gave those scores.

Finally, we directly asked for their feedback to improve ModSandbox to set the direction for the future system.

5.3.1 *Creating a set of Ground-truth Target Posts to be Filtered.*

We hired two external annotators from our university campus to label posts that must be filtered for the given moderation scenarios. Both were international students who are proficient in English and familiar with the online community like Reddit. Based on the scenarios, they were asked to label 1 on the posts to be filtered and 0 on the posts not to be. The inter-rater reliability measured with Cohen's Kappa was 0.45 for Task A and 0.67 for Task B. The scores were low even after having an asynchronous discussion session via email to reach agreement. This was because each annotator had different internal criteria for each scenario. For example, Annotator 2 considered that any post that mentions the usefulness of enrolling in a "bootcamp" should be filtered in Task A, while Annotator 1 did not agree with it. Although task B was more objective, the annotators still had disagreements between their labels. For example, Annotator 1 considered that any post that mentions "lockdown" should be filtered in Task B, while Annotator 2 did not agree with it. Thus, we did not directly compare user study participants configuration results with the ground truth dataset. The ground truth dataset was only used to assess the performance of the sorting algorithm.

5.4 Results

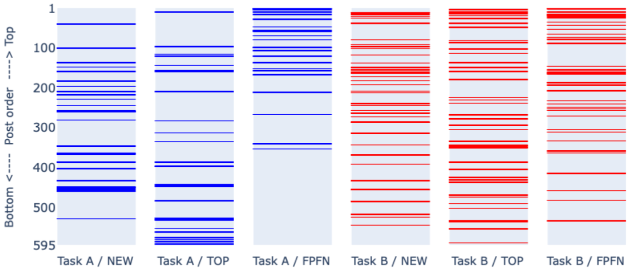


Figure 9: Locations of target posts to be filtered using different sorting methods in Task A and Task B. The target posts to be filtered that are labeled by two external annotators are marked in colors. *FP/FN sorting* for Task A (the third column) concentrates the target posts to be filtered at the top of the list so that the users can more easily see them.

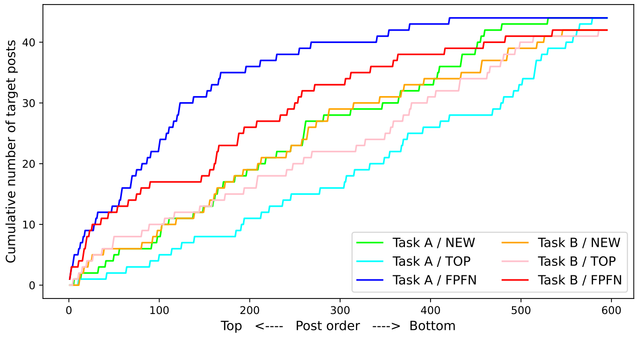


Figure 10: The cumulative numbers of target posts using different sorting methods in Task A and Task B. The target posts labeled by any of two external annotators are marked in cold colors (blue, sky-blue, and green) for Task A; and warm colors (red, pink, and orange) for Task B. It is shown that sorting by *FP/FN* recommendation (blue and red) concentrates the target posts to be filtered at the top of the list so that the users can see them earlier.

5.4.1 *RQ1-1: Can ModSandbox support moderators with detecting false positives and false negatives more easily?*

To verify how the "FP/FN recommendation" feature works on the main tasks of user study, we compared the order of posts among the three different sorting methods: *NEW*, *TOP*, and *FP/FN* (recommendation). The sorting by *NEW* and *TOP* are Reddit's default sorting methods that show the most recently published posts and the highest vote count posts, respectively. The results of these different sorting methods are shown in Figure 9 and 10. We used the labeled dataset we created with the two external annotators (Section 5.3.1).

In Figure 9, each post is marked as a line in blue and red. In Figure 10, the cumulative numbers of target posts from Task A and B are marked as three lines in cold (blue, sky-blue, and green) and warm (red, pink, and orange) colors, respectively. We found that the performance of FP/FN recommendation varies according to the moderation tasks. For Task A of filtering posts asking about getting CS-relevant jobs without CS-relevant degrees, many target posts to be filtered were located at the top when using *FP/FN* recommendation and thus were first shown to the users (third column in Figure 9, blue line in Figure 10). This contrasts with the sorting by *NEW* or *TOP* (first and second column in Figure 9, green and sky-blue lines in Figure 10). However, for Task B of filtering posts mentioning about COVID-19, *FP/FN* sorting (sixth column in Figure 9, red line in Figure 10) showed less noticeable differences from other sorting methods (fourth and fifth column in Figure 9, orange and pink lines in Figure 10).

5.4.2 RQ1-2: Can ModSandbox help moderators update the automated rules to reduce the false positives and false negatives? We analyzed the characteristics of rules and filtered posts after using the basic system and ModSandbox. We also compared them with a control group that had a second chance to update the rules with the same basic system. The results show that the benefits observed using ModSandbox were not just coming from having a second chance to improve the rules, but were actually coming from using the features of ModSandbox. Individual results of all participants are shown in Figure 12 and Figure 11, respectively. The first three rows of bar plots represent the complexity of the automated rules with the number of rules, checks, and strings. The other two rows indicate the results of the rules with the number of filtered posts and the similarity with the three example posts, which represent the targeted posts to be filtered. We arranged the x-axis order of participants with their experience with AutoModerator; the yellow background color represents the experienced moderators.

In the control group (Figure 11), six participants did not update their rule in the second chance in both tasks, although they were given additional time to refine their rules. Therefore, the measures from the filtered posts were not changed. In the post interview, they told us that the rules from the first try were already satisfying, and the second try with the same system felt like doing exact same thing again. P16 and P11 added a new rule to extend the filtering range in task A, their rules eventually filtered so many posts that the filtered posts became semantically far from the example posts. In task B, no participants added a new rule, so there was no significant change in their filtered posts except P11. Overall, participants did not make significant changes to the rules, but some took more time trying to improve the rules. The final rules in the control study resulted in too many posts being filtered and average semantic similarity being reduced. The consistency of filtered posts among participants in Table 3 also dropped in the second trial of the control condition.

	Control Group			Experimental Group		
System	basic-1	basic-2	Δ	basic	mod	Δ
Task A	0.038	0.016	(-0.022)	0.037	0.156	(+0.119)
Task B	0.773	0.676	(-0.097)	0.614	0.732	(+0.118)

Table 3: The consistency of filtered posts among participants in Fleiss' Kappa. Δ is the difference between the tries (second try − first try)

In the experimental group (Figure 12), we observed that participants start with relatively simple rules and make it to be more sophisticated as they use ModSandbox. The average numbers of rules, checks, and strings (horizontal lines on Figure 11 and 12) increased more with ModSandbox than the second try with the basic system. Starting with the basic system, most participants made a single list of keywords and phrases rather than advanced combinations of units. Specifically, five moderators from Task A and nine moderators from Task B submitted a single-rule and single-check configuration. For example, P5 submitted a rule that detects any 'change,' 'degree,' 'machine learning,' and 'worth' in the posts for Task A. When comparing the changes in the average number of rule, check, and string between control and experimental groups, the participants in the experimental group tend to update their primitive rules to have more rule, check, and string after using ModSandbox. This difference shows that ModSandbox lets participants try more sophisticated rules to accurately target the posts they want to filter.

As a result, most participants noticeably adjusted the number of filtered posts. In task A, six participants wrote the rules that filter over 50 posts with a basic system and updated them to filter less with ModSandbox. On the other hand, other participants updated the rules to filter more when they checked that their original rules filtered under 50 posts. It is observed that for P1, P2, P4, P5, P6, P7, P9, and P10, the semantic similarity increased after using ModSandbox. The changes of P1, P5, and P6 were statistically significant when tested using a t-test with Bonferroni correction (p=0.000, p=0.04, p=0.001, respectively). By running repeated measures ANOVA, we found improvement in the semantic similarity across all participants, and the result was marginally significant (p=0.056). In task B, the number of filtered posts was changed, but there were no significant differences in semantic similarity. This implies that using ModSandbox, the participants not only adjusted the number of posts being filtered but indeed the remaining filtered posts had high similarity with the targeted posts to be filtered.

As seen in Table 3, ModSandbox enhanced the consistency of filtered posts among participants compared to the control condition where the basic system was used again for the second trial. It indicates that ModSandbox helped moderators update their rules to filter similar sets of posts regardless of their prior knowledge and subjectivity to the moderation goals. However, we note that consistency should not be used as a metric on its own in a moderation context because there could be cases where people make the same mistakes in selecting the filtering words with ModSandbox. The mistakes would increase the consistency while the selected filtering words are not valid.

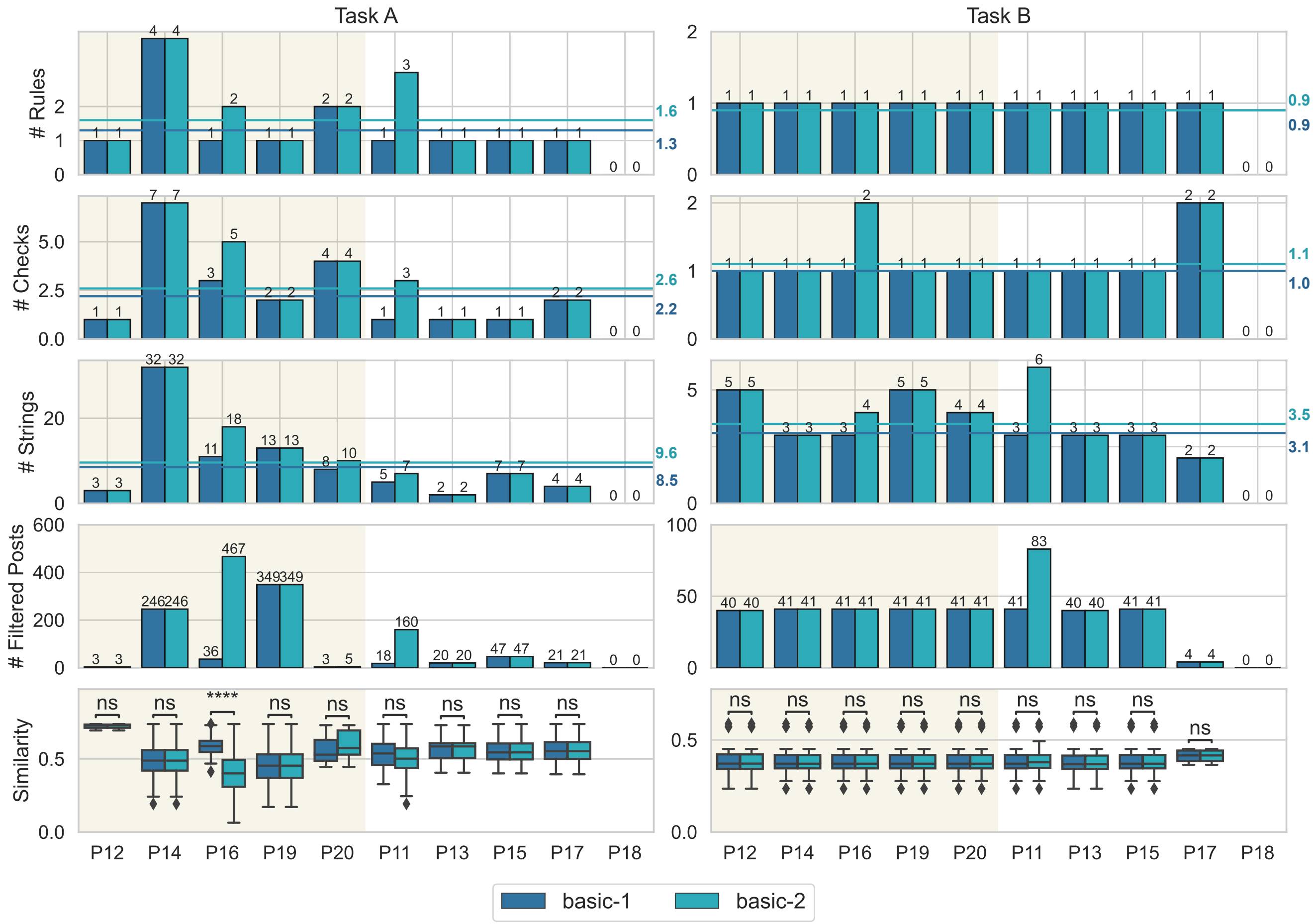


Figure 11: The plots display the number of rules, checks, strings, filtered posts, and their semantic similarities with three examples posts for each participant in the first (basic-1, blue color) and second (basic-2, cyan color) tries with the basic system for Task A and B. The yellow background indicates the participants who experienced the AutoModerator configuration. The blue and cyan horizontal lines in the first three rows of plots indicate the average number of rules, checks, and strings for each trial.

5.4.3 *RQ2-1: How do participants use the features of ModSandbox for the configuration process?*

We looked at each participant's system usage patterns and rule-making strategies in ModSandbox to understand the system's usefulness.

System usage pattern in ModSandbox. We found that they used ModSandbox to evaluate and update AutoModerater rules in a structured way; they created several routines of using ModSandbox's features.

First, all participants preferred to activate the FP / FN Recommendation feature (Feature 3) rather than sorting by *NEW* and *TOP* throughout the study session to help find the posts that are likely to be false positives or false negatives. Their process to update rules mostly began with finding false positives and false negatives using the "FP/FN Recommendation" feature (Feature 3). The most popular process was as follows. Seven participants (P1, P4, P5, P6, P8, P9, P10) first reviewed possible misses and false alarms that our feature recommended to find false positives and false negatives. Next, they moved the identified false positives and negatives into the "FP/FN Collection" panel. Then, they updated the AutoModerator configuration to resolve the collected posts on the FP/FN Collection panel.

We observed two patterns in collecting problematic posts (usually actual false positives and false negatives) in FP/FN Collection (Feature 2). A group of participants (P1, P5, P6, P8, P9, P10) collected several problematic posts at once and updated their rules by referring to all of them at once. However, some of them (P6, P8, and P9) failed to use this pattern because they could not find a breakthrough rule to resolve the collected false positives and false negatives at once. Otherwise, Two participants (P4, P5) tried to collect posts one by one in the "FP/FN Collection" panel (Feature 2), followed by a rule update after each collection. This resulted in fine-tuning the rules to resolve each and every post being collected in the "FP/FN Collection" panel. An exceptional pattern was observed from P2 and P7, where they did not use the "FP/FN Collection" panel at all

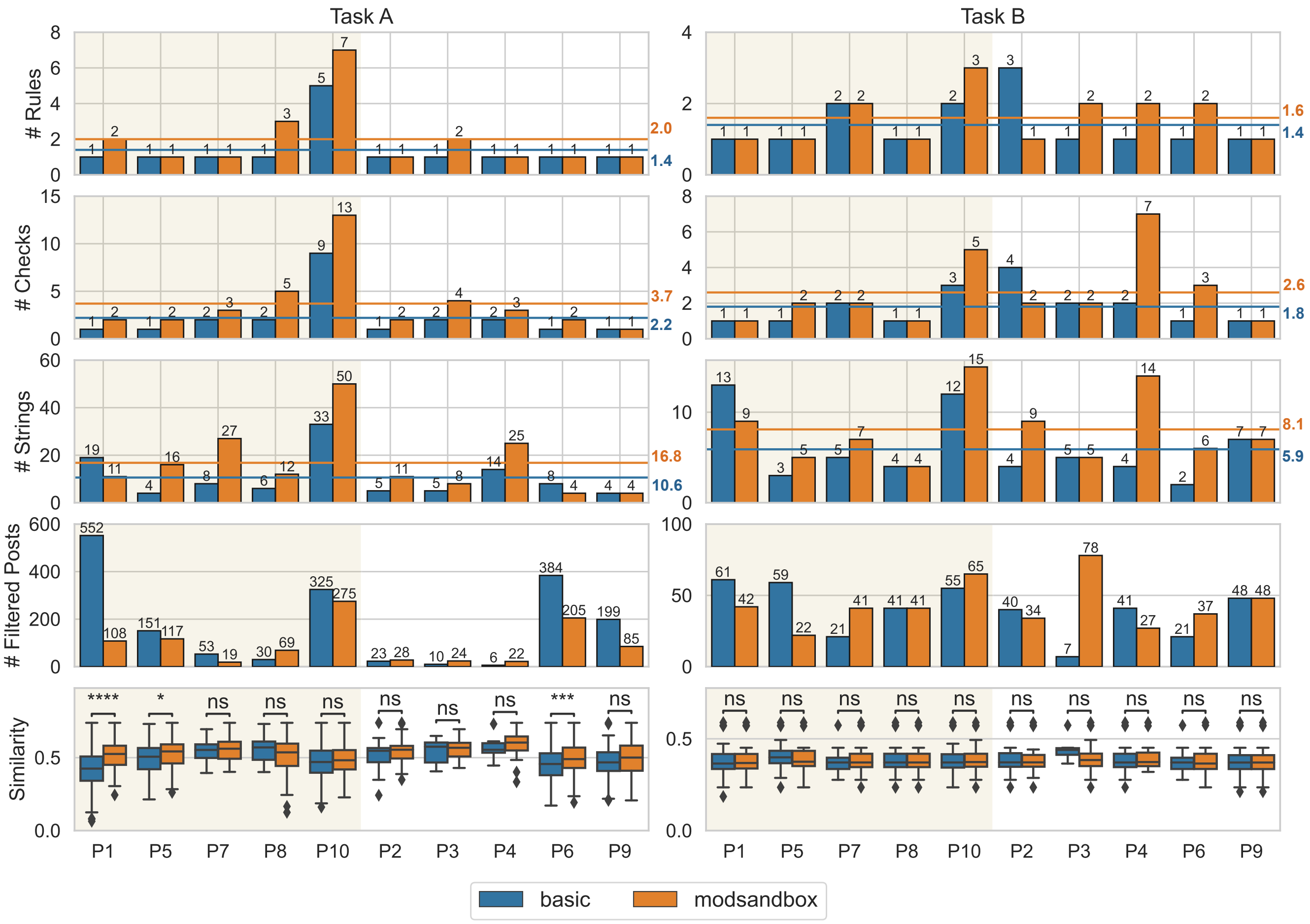


Figure 12: The plots display the number of rules, checks, strings, filtered posts, and their semantic similarities with three examples posts for each participant with the basic system (blue color) and ModSandbox (orange color) for Task A and B. The yellow background indicates the participants who experienced the AutoModerator configuration. The blue and orange horizontal lines in the first three rows of plots indicate the average number of rules, checks, and strings for each system.

and directly reflected the false positives she found from the Possible False Alarms Panel. P2 reported that it was cumbersome to move posts to the FP/FN Collection because she was able to just quickly deal with the false positives she found without having to move them. All participants(P2, P4, P5, P7) using this one-by-one strategy presented a common strategy to update rules, which is writing a filter with a large number of keywords at first and then adding white-list keywords to exclude false positives. Six participants (P1, P5, P7, P8, P9, P10) used the Automated Rule Analysis Panel (Feature 4) as an extra supporting tool to understand how the rules are working. Four of them (P1, P5, P9, P10) refer to this feature to quickly find which part of the rules are filtering the false positives and false negatives in "FP/FN Collection" panel. The other two (P7 and P8) checked the number of posts that each rule and keyword affected. They first checked whether their rules were catching too many posts or not by looking at the ratio bar in the Sandbox Panel and used mouse hovering on highlights to remove the relevant part of the rules.

Rule-making strategies in ModSandbox. Participants elaborated their rule-making strategies while using ModSandbox. Four participants (P2, P4, P6, P8) first created a rule with a large list of keywords to catch targeted posts and update the rules to reduce the false positives. To be specific, P2, P4, and P8 added some white-list keyword filters to exclude the false positives. P8 described *"First I skimmed through the posts and the task for keywords that might be able to match what I want. I then checked the false positives to find additional keywords that I could add to the rules to reduce the amount of false positives".* P6 followed the same strategy, but he told that the configuration became so muddled that he was getting too many false positives and regretted that he should have thought a lot more different than he had at the beginning in trying to filter his words. P7 introduced his impressive strategy. He first tried to think of a simple algorithm that can catch or reject target posts in this head, e.g., find all posts including word X and word Y, but not any posts including word Z. Then he thought of keywords that fulfill that logic. Finally, he made multiple rules with three checks: a filtering list with basic keywords, a list with additional keywords to narrow down the range, and a list with keywords that should not be filtered. Each rule has the range of $(\mathbf{X} \cap \mathbf{Y}) \cap \mathbf{Z}^c$.

5.4.4 RQ2-2: How do participants perceive the usefulness of ModSandbox in the configuration process? After the user study, we asked all participants how useful each feature was and how they think they can be improved. We summarize the responses, highlighting the difference in main strength of each feature according to the moderation task and user's condition.

Feature 1. A Sandbox Environment: Valuable to see what posts are being filtered. Six participants (P2, P3, P4, P6, P8, P9) responded that a sandbox environment was valuable because they could see what posts are being filtered in real-time. P1 said that he gave a high score for this function because it can show the results of AutoModerator applied to the community without affecting the community. However, three participants (P4, P7, P9) said that the Sandbox UI showed so much data compared to Feature 3: FP/FN Recommendation, that they did not like it. P7 said *"There are a lot of posts shown at a time, which makes it less useful when compared to the features with fewer posts shown."*

Feature 2. FP/FN Collection: More useful to participants who were familiar with configuring AutoModerator. Six participants(P1, P2, P6, P7, P8, P10) mentioned that seeing the posts manually gathered in the Post Collections panel was useful for the user study task. Specifically, P6, P7, and P8 noted its usefulness in finding proper rules. P6 reported *"It was great to actually see which posts are false negatives and false positives so that it was easier to look for keywords that are more relevant to the current topic."* However, Three novice moderators (P3, P6, P9) pointed out they are difficult to use and gave lower usefulness scores to this feature (Table 4). P9 said *"It was useful in that is showed me how the keywords were being used but it left me wondering how to apply this."*

Feature 3. FP/FN Recommendation: The most useful feature for everyone, but only works well if posts have semantic similarities. Five participants (P1, P4, P5, P8, P10) liked this feature because it allowed them to grasp *probable* false positives and false negatives, and thus quickly find *actual* false positive and negative posts. P10 mentioned *"The possible misses, false alarms was very helpful in showing what things I missed with my filter. It definitely saved me tons of time of scrolling through matches to find bad ones."* However, P2, P3, and P10 felt that possible misses are less accurate in Task B. P3 wrote *"This feature is so convenient, but I think there were many articles in the Possible Misses that did not seem to be included in the task"*. Interestingly, P7 doubted the accuracy of the algorithm *"I'm unsure how good the algorithm is and I'd be afraid that focusing on these will miss important posts"*. They evaluated this function as less useful in Task B, but as most useful function overall (Table 4). We note that this feature was indeed less accurate in Task B because each post mentioning COVID-19 had very different semantic and context compared to Task A. Targeted posts in Task A shared similar topics, but targeted posts in Task B had varying topics.

Feature 4. Automated Rule Analysis and Highlights: More useful with more complex rules. "Automated Rule Analysis" panel helped Four experienced moderators (P5, P7, P8, P10) when they analyze the code and determine which rules or words are good and bad for the task. P5 answered that it is helpful to see how each code impacts on the filtered results, thus making it easier for them to remove keywords that were yielding too many false results. Two experienced moderators (P1, P8) stated that they were able to understand how the rules work but they did not feel the need to use the panel much. Interestingly, P1 suggested a novel way to utilize what is seen in the Automated Rule Analysis panel. He pointed out that it is easily readable data that could be presented to other moderators as evidence to discuss the flaws and strengths of each rule. Four novice moderators (P2, P3, P4, P9) preferred "Highlights" because it helps notice where the keywords in the rules are. Furthermore, P4 felt confident that she could identify why certain keywords were filtered or not.

Feedback to improve ModSandbox. Three moderators (P1, P4, P7) provided feedback to improve ModSandbox through the post-survey. P1 and P4 mentioned that the UI could be more simplified so that novice or casual moderators could also easily use it. P4 and P7 suggested analyzing word frequency in the "FP/FN Collection" panel so that the most frequent words can be used as recommended keywords when writing keyword-based automated rules.

6 DISCUSSION

In this work, we investigate the challenges that online content moderators faced when configuring automated moderation tools and presented a novel approach to help them quickly find false positives and negatives and improve their automated rules. In the following, we discuss the impact of intelligent NLP algorithms that help find false positives and false negatives, the potential impact of recommending concrete methods on how to update automated rules, supporting efficient collaboration between moderators, reducing emotional labor for online content moderators using ModSandbox, and ModSandbox being a practical solution for other platforms beyond Reddit.

6.1 The Impact of Intelligent Algorithms on Finding False Positives and False Negatives

In our user study, the accuracy of the algorithms in detecting possible false positives and negatives had a significant impact on the trust of participants and the perceived usefulness of the system. The experimental result in Figure 10 shows that our algorithm was not as effective as Task A in Task B. Due to this, three participants commented on the unreliability of the given functionality. P7 reported distrust of the algorithm: *"I'm unsure how good the algorithm is and I'd be afraid that focusing on these will miss important posts."* While the targeted posts to be filtered in Task A asked to filter posts with a similar context asking about getting CS-related jobs, Task B asked to filter posts that contain any keyword related to COVID-19, which may appear in various different contexts. These posts could have any topic spanning from talking about the impact of anti-vaccine protests in the job market to having to work remotely due to quarantine. In addition, the Universal Sentence Encoder may not be suitable for Task B because it was pre-trained with sources from

Condition	Task	Sandbox	FP/FN	Collections	Analysis	System
Experienced	A	4.8(1.8)	**6.0(0.7)**	**5.2(1.9)**	5.2(0.8)	**6.2(0.8)**
	B	4.6(1.8)	5.4(1.1)	**6.2(0.8)**	5.2(1.1)	
Novice	A	5.0(1.6)	**5.8(0.8)**	4.6(1.6)	5.0(1.6)	5.4(2.1)
	B	**6.0(1.5)**	5.4(0.4)	5.0(0.9)	5.0(2.0)	
	Total	5.1(1.6)	**5.7(0.9)**	5.2(1.6)	5.1(1.4)	

Table 4: Average usefulness scores (and standard deviation) of each feature in ModSandbox

Wikipedia, web news, web question-answer pages and discussion forums before the COVID-19 pandemic [6].

The algorithm we adopted to predict false positives and negatives calculates semantic similarities of posts in the level of sentences, not keywords. Therefore, other algorithms could be tested to see the impact on tasks similar to Task B in our user study. For example, a word embedding model [35] or a language model pre-trained with recent social media content [33] may be more effective for similar tasks. For the current system, we only use a single algorithm, but it may be possible to improve the system by supporting alternative algorithms to recommend possible false positives and negatives, and let moderators compare the performance between them and apply what works best for them. Another way to improve the feature to find possible false positives and false negatives is to expand the range of imported data. ModSandbox extracts the possible misses and false alarms from only posts on a subreddit. The system can potentially use posts from multiple similar subreddits that share similar norms [10] or an AI-generated virtual community that has the same topic and rules [38]. A more significant number of posts can help moderators make a concrete and preventive configuration by providing various examples that reflect prospective behavior from their communities [5].

6.2 Further Recommending Concrete Ideas on How to Update the Automated Rules

Going further from just showing the possible false positive and false negative posts to the users, recommending concrete action items on how to update the automated rule may be helpful to the users. During the user study, three participants (P3, P6, P9) found it difficult to extract meaningful patterns to be written in a rule when using the "FP/FN Collection" panel. They lacked ideas to update the rules using these patterns because they were unable to identify the appropriate keywords. As a solution, we can leverage the "FP/FN Collection" panel to suggest concrete directions to improve the configuration. In the study, two participants (P4, P7) suggested showing frequently occurring keywords and inverse frequency analysis, which is a method to measure how much information each word provides. This approach may help find useful keywords based on the collected posts to improve the rules. Furthermore, ModSandbox can potentially suggest a single regular expression that detects these useful keywords. On the other hand, ModSandbox can adopt the sorting idea from previous studies [30] to prune the rules, where sorting rules, checks, and strings based on their ratio of filtered posts in the rule analysis panel could help quickly find the best configuration among many choices.

The patterns of rule updates observed in the user study can guide the design of recommendations for future automated rules. Some participants added keywords they found in the possible false positive examples as white-list keywords. Furthermore, P10 started with a single check with a list of keywords and then added an additional normal check or reverse check, which is a condition that the post must not meet. These structured procedures can become a framework to help guide the writing of a better AutoModerator configuration. That is, we believe that guiding moderators to make informed updates to their AutoModerator configuration is a promising next step. The system can potentially recommend effective rules based on keyword extraction results and rule-update patterns. However, such data-driven recommendations may sometimes suggest rules that humans cannot interpret. To overcome this, the system may list promising options for changing current rules so that moderators can build more accurate and interpretable rules.

6.3 Facilitating Distributed Governance for Online Communities

In the user study, P1, who moderates a high-traffic subreddit, said ModSandbox *"would not only allow for refinement of rules, but presentation thereof"*. P1 meant that one could use ModSandbox to demonstrate the expected results of AutoModerator configurations to peer moderators during discussions that are conducted before any moderation decision. P1 suggested that such use of ModSandbox can help casual moderators become more involved in the configuration process. A previous study [22] showed that only a few moderators actively configure AutoModerator due to its difficulty in learning how to use it. Therefore, they suggested that an automated system could be designed to make it easier for moderators to understand how to use it. Tools that can visualize moderation rules and their results, such as ModSandbox, can be a promising solution to support many non-tech-savvy moderators to participate in automated tools. In addition, ModSandbox can support them in learning to use a regular expression in the configuration by testing it in the sandbox environment. We expect this line of work to reduce the barriers for novice moderators by providing a learning opportunity.

Furthermore, ModSandbox has the potential to serve a team of moderators and community users in their distributed decision-making scenarios. We can extend ModSandbox to support multiple moderators in collaborative writing of rules while discussing the expected impact of automated rules on their community. We could even give these capabilities to community users, increasing moderation transparency and awareness. Recent studies([24, 45, 54]) have introduced software infrastructures and strategies to support distributed governance for online communities. For example, moderators can run a poll to make a decision about a change in an automated rule, showing statistics from ModSandbox, not the rule itself. In this way, ModSandbox contributes a special purpose software infrastructure and governance layer for algorithmic moderation to this research thread.

6.4 Reducing Cognitive Labor in Setting Up Automated Moderation Tools

The feature "FP/FN Recommendation" can help reduce the cognitive labor of moderators when configuring automated tools. For Task A in our study, we observed that this feature could help participants identify false positives and negatives earlier without skimming through all posts imported into the system (Figure 10). P10 mentioned that ModSandbox saved time in finding problematic posts compared to scrolling through a large number of community posts. Furthermore, moderators can avoid being exposed to toxic and harassing posts during moderation by using this feature. Although the typical moderation process requires emotional labor for moderators because they are exposed to these toxic posts while skimming through posts [17, 29, 40, 41], using the "FP/FN Recommendation" feature creates a separate space for moderators to focus only on posts related to the current moderation task. Facebook recently built an AI-supported moderation system to reduce the number of posts paid moderators should review by automatically excluding obviously harmful content and first sorting ambiguous content [53]. In a similar vein, ModSandbox also benefits volunteer moderators by reducing the number of posts that they need to review. The sandbox feature can also help reduce the cognitive load because the moderators do not have to worry about any malfunction of a rule that may affect their community values negatively. With the sandbox, moderators can give an easy trial for any rules they want to test.

7 CONCLUSION

This paper proposes ModSandbox, a virtual sandbox system for online content moderation, which supports human moderators in predicting and preventing false positives and false negatives of automated rules for their communities (e.g., filtering innocent posts or missing posts that should be filtered). ModSandbox was built by investigating the four main challenges that moderators face during the configuration of their automated rules. The four main features driven from and corresponding to each challenge help moderators analyze their current automated rules and improve them by referring to the patterns found from the collected targeted posts to be filtered. Our user study with community moderators from various platforms demonstrates that ModSandbox can help configure automated rules that reflect the detailed intentions of the moderators. Features like "FP/FN Recommendation" can reduce cognitive labor in setting up automated moderation rules because human moderators do not have to be exposed to toxic posts while analyzing their rules. Potential extended use cases of ModSandbox include supporting collaboration between moderators by sharing the results of the system and reducing the barriers for novice moderators by providing a learning opportunity inside ModSandbox with real community data.

ACKNOWLEDGMENTS

This work was supported in part by the National Research Foundation of Korea (NRF) grant funded by the Ministry of Science and ICT (NRF-2020R1A2C2005479, NRF-2022R1C1C1003123, NRF-2020R1I1A1A01072385, ERC (NRF-2018R1A5A1059921), and the DGIST Start-up Fund Program (2021070007)).

REFERENCES

[1] S Abarna, JI Sheeba, S Jayasrilakshmi, and S Pradeep Devaneyan. 2022. Identification of cyber harassment and intention of target users on social media platforms. *Engineering applications of artificial intelligence* 115 (2022), 105283.

[2] Fernando Alfonso III. 2014. Reddit strips r/technology from its homepage following moderator and censorship drama. https://www.dailydot.com/unclick/reddit-censorship-technology-drama-default/

[3] Jie Cai and Donghee Yvette Wohn. 2019. Categorizing Live Streaming Moderation Tools: An Analysis of Twitch. *International Journal of Interactive Communication Systems and Technologies (IJICST)* 9, 2 (2019), 36–50.

[4] Twitter Help Center. 2021. *Hateful conduct policy.* Retrieved 2021 from https://help.twitter.com/en/rules-and-policies/hateful-conduct-policy

[5] Damon Centola. 2010. The spread of behavior in an online social network experiment. *science* 329, 5996 (2010), 1194–1197.

[6] Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, et al. 2018. Universal sentence encoder for English. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. 169–174.

[7] Stevie Chancellor, Yannis Kalantidis, Jessica A Pater, Munmun De Choudhury, and David A Shamma. 2017. Multimodal classification of moderated online pro-eating disorder content. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. 3213–3226.

[8] 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*. 1201–1213.

[9] Eshwar Chandrasekharan, Chaitrali Gandhi, Matthew Wortley Mustelier, 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 (2019), 1–30.

[10] 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. *Proceedings of the ACM on Human-Computer Interaction* 2, CSCW (2018), 1–25.

[11] Eshwar Chandrasekharan, Mattia Samory, Anirudh Srinivasan, and Eric Gilbert. 2017. The bag of communities: Identifying abusive behavior online with preexisting internet data. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. 3175–3187.

[12] Kevin Collier. 2014. Reddit's r/technology has a secret list of about 50 words you can't use in headlines. *The Daily Dot* (Apr 2014). https://www.dailydot.com/unclick/reddit-technology-banned-words/

[13] Alexis Conneau, Douwe Kiela, Holger Schwenk, Loic Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. *arXiv preprint arXiv:1705.02364* (2017).

[14] Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *Proceedings of the international AAAI conference on web and social media*, Vol. 11. 512–515.

[15] Nicholas Diakopoulos. 2016. Accountability in algorithmic decision making. *Commun. ACM* 59, 2 (2016), 56–62.

[16] Karthik Dinakar, Roi Reichart, and Henry Lieberman. 2011. Modeling the detection of textual cyberbullying. In *Proceedings of the International AAAI Conference on Web and Social Media*, Vol. 5. 11–17.

[17] 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*. 1–13.

[18] Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin* 76, 5 (1971), 378.

[19] Robert Gorwa, Reuben Binns, and Christian Katzenbach. 2020. Algorithmic content moderation: Technical and political challenges in the automation of platform governance. *Big Data & Society* 7, 1 (2020), 2053951719897945.

[20] James Grimmelmann. 2015. The virtues of moderation. *Yale JL & Tech.* 17 (2015), 42.

[21] Shagun Jhaver, Darren Scott Appling, Eric Gilbert, and Amy Bruckman. 2019. " Did You Suspect the Post Would be Removed?" Understanding User Reactions to Content Removals on Reddit. *Proceedings of the ACM on human-computer interaction* 3, CSCW (2019), 1–33.

[22] Shagun Jhaver, Iris Birman, Eric Gilbert, and Amy Bruckman. 2019. Human-machine collaboration for content regulation: The case of Reddit Automoderator.

ACM Transactions on Computer-Human Interaction (TOCHI) 26, 5 (2019), 1–35.

[23] Shagun Jhaver, Quan Ze Chen, Detlef Knauss, and Amy X Zhang. 2022. Designing Word Filter Tools for Creator-led Comment Moderation. In *CHI Conference on Human Factors in Computing Systems*. 1–21.

[24] Shagun Jhaver, Seth Frey, and Amy Zhang. 2021. Designing for Multiple Centers of Power: A Taxonomy of Multi-level Governance in Online Social Platforms. *arXiv preprint arXiv:2108.12529* (2021).

[25] 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), 1–23.

[26] Prerna Juneja, Deepika Rama Subramanian, and Tanushree Mitra. 2020. Through the Looking Glass: Study of Transparency in Reddit's Moderation Practices. *Proceedings of the ACM on Human-Computer Interaction* 4, GROUP (2020), 1–35.

[27] 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*. 1–8.

[28] 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 (2019), 1–23.

[29] 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*. 1152–1156.

[30] Josua Krause, Aritra Dasgupta, Jordan Swartz, Yindalon Aphinyanaphongs, and Enrico Bertini. 2017. A workflow for visual diagnostics of binary classifiers using instance-level explanations. In *2017 IEEE Conference on Visual Analytics Science and Technology (VAST)*. IEEE, 162–172.

[31] PJS Kumar, Polagani Rama Devi, N Raghavendra Sai, S Sai Kumar, and Tharini Benarji. 2021. Battling Fake News: A Survey on Mitigation Techniques and Identification. In *2021 5th International Conference on Trends in Electronics and Informatics (ICOEI)*. IEEE, 829–835.

[32] Cheng Hua Li and Jimmy Xiangji Huang. 2012. Spam filtering using semantic similarity approach and adaptive BPNN. *Neurocomputing* 92 (2012), 88–97.

[33] Daniel Loureiro, Francesco Barbieri, Leonardo Neves, Luis Espinosa Anke, and Jose Camacho-Collados. 2022. Timelms: Diachronic language models from twitter. *arXiv preprint arXiv:2202.03829* (2022).

[34] J Nathan Matias. 2016. Going dark: Social factors in collective action against platform operators in the Reddit blackout. In *Proceedings of the 2016 CHI conference on human factors in computing systems*. 1138–1151.

[35] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

[36] Chikashi Nobata, Joel Tetreault, Achint Thomas, Yashar Mehdad, and Yi Chang. 2016. Abusive language detection in online user content. In *Proceedings of the 25th international conference on world wide web*. 145–153.

[37] Deokgun Park, Simranjit Sachar, Nicholas Diakopoulos, and Niklas Elmqvist. 2016. Supporting comment moderators in identifying high quality online news comments. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 1114–1125.

[38] Joon Sung Park, Lindsay Popowski, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2022. Social Simulacra: Creating Populated Prototypes for Social Computing Systems. In *In the 35th Annual ACM Symposium on User Interface Software and Technology (UIST '22)* (Bend, OR, USA) *(UIST '22)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3526113.3545616

[39] Reddit. 2021. Transparency Report 2021. https://www.redditinc.com/policies/transparency-report-2021

[40] Sarah T Roberts. 2016. Commercial content moderation: Digital laborers' dirty work. (2016).

[41] Sarah T Roberts. 2019. *Behind the screen*. Yale University Press.

[42] Mozhgan Saeidi, Samuel Bruno da S Sousa, Evangelos Milios, Norbert Zeh, and Lilian Berton. 2019. Categorizing online harassment on Twitter. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 283–297.

[43] Koustuv Saha, Eshwar Chandrasekharan, and Munmun De Choudhury. 2019. Prevalence and psychological effects of hateful speech in online college communities. In *Proceedings of the 10th ACM conference on web science*. 255–264.

[44] Vlad Sandulescu and Martin Ester. 2015. Detecting singleton review spammers using semantic similarity. In *Proceedings of the 24th international conference on World Wide Web*. 971–976.

[45] 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 (2021), 1–26.

[46] Joseph Seering. 2020. Reconsidering Self-Moderation: the Role of Research in Supporting Community-Based Models for Online Content Moderation. *Proceedings of the ACM on Human-Computer Interaction* 4, CSCW2 (2020), 1–28.

[47] 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 (2019), 1417–1443.

[48] Monika Singh, Divya Bansal, and Sanjeev Sofat. 2016. Behavioral analysis and classification of spammers distributing pornographic content in social media. *Social Network Analysis and Mining* 6, 1 (2016), 1–18.

[49] Sara Sood, Judd Antin, and Elizabeth Churchill. 2012. Profanity use in online communities. In *Proceedings of the SIGCHI conference on human factors in computing systems*. 1481–1490.

[50] Sara Owsley Sood, Elizabeth F Churchill, and Judd Antin. 2012. Automatic identification of personal insults on social news sites. *Journal of the American Society for Information Science and Technology* 63, 2 (2012), 270–285.

[51] Top.gg. 2022. Mee6 discord bot: The #1 discord bot list. https://top.gg/bot/159985870458322944

[52] Kristen Vaccaro, Christian Sandvig, and Karrie Karahalios. 2020. " At the End of the Day Facebook Does What ItWants" How Users Experience Contesting Algorithmic Content Moderation. *Proceedings of the ACM on Human-Computer Interaction* 4, CSCW2 (2020), 1–22.

[53] James Vincent. 2020. Facebook is now using AI to sort content for quicker moderation. *The Verge* (Nov. 2020). https://www.theverge.com/2020/11/13/21562596/facebook-ai-moderation

[54] 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*. 365–378.