Extracting and Visualizing User Engagement on Wikipedia Talk Pages

Carlin MacKenzie
The University of Edinburgh
Edinburgh, United Kingdom
s1724780@ed.ac.uk

John R. Hott
The University of Virginia
Charlottesville, United States of America
jrhott@virginia.edu

ABSTRACT

As Wikipedia has grown in popularity, it is important to investigate its diverse user community and collaborative editorial base. Although all user data, from traffic to user edits, are available for download under a free and open license, it is difficult to work with this data due to its scale.

In this paper, we demonstrate how consumer hardware can be used to create a local database of Wikipedia's full edit history from their public XML data dumps. Using this database, we create and present the first visualizations of how editing on talk pages differs between user groups. Our visualizations demonstrate that low quality edits are primarily performed by IP users, rather than blocked users, and that overall engagement with talk pages has plateaued over the last 10 years across all user groups. Finally, we investigate the feasibility of classifying blocked users using this dataset as an example of future research directions. However, we demonstrate the difficulty of this task and find that additional data or a more advanced model would be needed to classify them, as our approach didn't provide sufficient information to do this.

We anticipate that our visualizations and data extraction process are of interest to the community and will provide researchers with the tools needed to use Wikipedia's valuable data when resources are limited.

CCS CONCEPTS

• **Information systems → Data extraction and integration**; *Wikis*; • **Human-centered computing → Information visualization**.

KEYWORDS

Wikipedia, classification, data visualization, talk pages, data extraction

ACM Reference Format:
Carlin MacKenzie and John R. Hott. 2021. Extracting and Visualizing User Engagement on Wikipedia Talk Pages. In *17th International Symposium on Open Collaboration (OpenSym 2021), September 15–17, 2021, Online, Spain.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3479986.3479995

OpenSym 2021, September 15–17, 2021, Online, Spain

ACM ISBN 978-1-4503-8500-8/21/09…$15.00
https://doi.org/10.1145/3479986.3479995

1 INTRODUCTION

Twenty years after its release, Wikipedia needs no introduction. Started as an experiment in anonymous, public collaboration, it is now the largest and most popular reference work on the Internet [1]. Additionally, its talk pages feature some of the most high-quality debate on the Internet which provides fertile ground for research [4][21]. However, too little attention has been paid to the engagement of users on talk pages as a vehicle for understanding and approximating user and group activity across Wikipedia as a whole [15]. We extract user interactions on talk pages throughout the entire edit history, along with group membership among the users, in order to visualize and depict engagement patterns across and among the users and groups. We anticipate these visualizations will provide (i) additional insights on how editors engage with each other on these pages and (ii) demonstrate whether talk page interactions provide a comprehensive enough depiction of the overall diversity in editor engagement to classify users into membership in one of these groups.

Analyzing Wikipedia data presented some challenges, due to several limiting factors. Like Wikipedia itself, documentation about accessing the data is community generated, with little top-down guidance of best practices or official tools. We found multiple publicly curated lists of tools[1], each providing a combination of maintained and abandoned projects. None of the tools found provided an efficient mechanism for procuring user engagement data and edit history.

Although the data is made as public as possible, the direct exports of the database tables are not published. Instead, the full edit history of all pages is released as a set of ≈ 600 archives, each of which extract to ≈ 50 GB XML files. Other options are provided to access individual files or limited revisions, however none address the need to consume the entire edit history. For example, alternative versions of the XML data files, denoted as "multistream" [19] and compressed using bzip2 [16] with multiple bzip2 streams per file, are provided to allow indexing and extracting a particular page without the need to decompress the entire file. Additionally, if a smaller section of Wikipedia is of interest, such as a category or a set of pages, the Special:Export[2] tool can be used. Unfortunately this tool is limited to the 1,000 oldest or newest revisions.

Since our work seeks to depict interactions among users and groups across all history, using Special:Export or multistream data

[1] https://meta.wikimedia.org/wiki/Datasets#Tools_to_extract_data_from_Wikipedia:, https://meta.wikimedia.org/wiki/Data_dumps/Other_tools, https://meta.wikimedia.org/wiki/Data_dumps/Tools_for_importing, https://web.archive.org/web/20191218101830/http://wikipapers.referata.com/wiki/List_of_tools
[2] https://en.wikipedia.org/wiki/Special:Export

files are insufficient. Therefore, we focus on processing the published data dumps consisting of all Wikipedia edits[3].

We extract and analyze edits on talk pages from the published data dumps to depict user engagement on Wikipedia. We anticipate that there is a larger diversity in editor engagement with these pages compared with other pages, such as those in main space, as users are making requests and having discussions with other users [15]. We separate users into several distinct groups, such as special users, blocked users, IP users and bots. This provides a framing for visualization and classification.

More specifically, in this study, we aim to explore the diversity of editor engagement on talk pages through a discussion of the following research questions:

RQ 1. How can the Wikipedia dataset be made more accessible, in terms of reduction in size or computation time, for research using consumer hardware?
RQ 2. Do visualizations of the data and metadata extracted when addressing RQ1 provide insights into differentiating Wikipedia user groups?
RQ 3. Does the metadata extracted while addressing RQ1 alone provide enough context about user edits to classify blocked users?

In response to RQ1, we present an open-source tool to create a database of edits for any Wikipedia namespace (Section 3). It downloads and partitions each dump before extracting the diff and metadata (which we refer to as *features*) of each edit. Next, regarding RQ2, we present the first visualizations depicting how different user groups interact with Wikipedia talk pages (Section 4). We then investigate RQ3, by demonstrating a future area of research with this data (Section 5). We attempt to classify blocked users using a linear classifier on the calculated features, rather than the edits themselves; while providing a provoking depiction of the group, it was unable to adequately classify the individual users. In the last section, we discuss our findings and outline future research directions.

2 RELATED WORK

Multiple studies have emerged in an attempt to understand user engagement on Wikipedia. Recent work has focused on classifying actors, detecting vandalism, analyzing the content of talk pages, and extracting a social network of users. In terms of visualizing talk page data, focus is mostly on user interaction, specifically regarding edit wars [3][13] or deletion discussions [18].

Rawat et al. [14] attempted to accurately classify abusive actors. They acquired their data by scraping user contributions from Wikipedia and applying machine learning to this data set. Their model provided an 84% accuracy, however the data set they used was very small. Our research instead provides a vehicle for expanding this dataset to all Wikipedia edits.

In the field of vandalism detection, Javanmardi [8] provides a high performing and fast model. They used a data set of Wikipedia edits which were manually classified to be spam or not spam. They created a classifier with 66 features which had an accuracy of 95.5% Area Under Curve (AUC) on the test set. To create the high performing model, they used the Lasso technique which resulted in 27 features and 95% AUC.

Schneider et al. [15] discuss the articles' talk pages. They aimed to classify the diversity in these pages and created thirteen such categories. They explored how users could signal which category their edit belonged to for aggregation purposes. They found that the most controversial articles have relatively short talk pages due to repetitive arguments in which neither side convinces the other. In contrast, Martinez-Ortuno et al. [11] looked at users' talk pages and how this is related to user activity. Compared to article talk pages, user talk pages can be thought of as the user's profile where people can thank or ask questions of the user. This is therefore a good predictor of a user's standing in the community. The researchers did not find a direct correlation between negative messages and decreased edits, but did present a model that could be used to predict user edit activity.

From a user standpoint we can look at the social networks of Wikipedia. Massa [12] found that extracting a network-based dataset could be approached in three ways, each of them flawed. Manual extraction is the most reliable but very time consuming and would not find edits which were reverted. Scraping talk pages faced many challenges such as custom signatures. Finally, they used the Wikipedia XML dumps. This was the most accurate for finding user's edits but could not verify to whom users were replying.

3 DATA ACQUISITION

In order to process the large amount of data in the compressed XML data dumps, and address our first research question RQ1, we created a tool NSDB (Namespace Database) [9] to extract all edits from a user-provided namespace. Succinctly, our tool temporarily downloads, splits, and extracts edit diffs and user-interaction features to create a complete SQL database of edits for any Wikipedia namespace, which is specified on the command line. Based on user parameters, it limits the amount of temporary storage and number of cores used.

3.1 Namespace Database

To ensure the tool is of use to future researchers, NSDB was developed in Python, since it is a popular and stable language for Data Science. For data storage, the tool utilizes either a MySQL or MariaDB database, since they are both robust and open-source. The tool has been released as open-source on GitHub[4], including thorough documentation.

Downloading the XML dumps can take significant time due to the location of any given mirror, high load, throttling, traffic, etc. Consequently, a speed test is performed to each mirror before the dump is downloaded. As each dump is independent of the others, we can easily parallelize parsing. To maximize parallelization, and localize errors, we split each dump into several partitions. This functionality is performed by a helper tool, `splitwiki`, which splits each dump at page boundaries into N partitions—123 per file on average—preserving the header with each one, to create a valid

[3]https://dumps.wikimedia.org

[4]https://github.com/carlinmack/NamespaceDatabase

Table 1: Features extracted from edits

(1) comment contains "copyedit"	(7) len. of longest inserted word	(13) ratio of # pronouns to # words
(2) comment contains "Personal Life"	(8) len. of longest inserted char. sequence	(14) number of deleted words
(3) comment length	(9) ratio of inserted capitalization	(15) if the article was blanked
(4) ratio of special char's in comment	(10) ratio of inserted digits	(16) if they inserted vulgarity
(5) # inserted internal links	(11) ratio of inserted special char's	(17) if they are a special user
(6) # inserted external links	(12) ratio of inserted whitespace	

smaller dump. This means that the storage requirement of the partitioned dumps is slightly greater than the singular file, however they allow for more efficient parsing and fine-grained error handling.

Each partition is parsed by `parse`. The parser streams each page in the partition, extracts features and inserts them into the database. MediaWiki's mwxml tool [6] is used for efficiently streaming the XML, allowing it to be iterated in Python with low memory overhead. Since each edit in the dump is stored as the resulting full page, NSDB must extract the diff between the current and previous edit to determine exactly which text was modified. The wdiff [10] tool was used to provide a word-level difference between the edits, extracting both the added and/or deleted text. If the page is in the selected namespace, this difference is then stored in the database to provide a more granular user-based edit history. From these texts, measurements about the text–*features*–are calculated and stored alongside the edit. When deciding which features to extract, we followed the methodology of Javanmardi [8]. In total we extracted 17 out of 27 of their features, listed in Table 1.

Features without implementation details were not implemented.

If the page is not in the selected namespace, NSDB calculates basic statistics on the editor, including number of edits and reverted edits, which is stored with the user information. This means that for the target namespace we insert into the database for every revision, whereas in the non-target namespace we insert once for each user that makes an edit. Finally, we add information about the page to the database.

In the database, there are four main tables:

- **edit**, which stores each edit for the target namespace, including added and deleted text as well as calculated feature values;

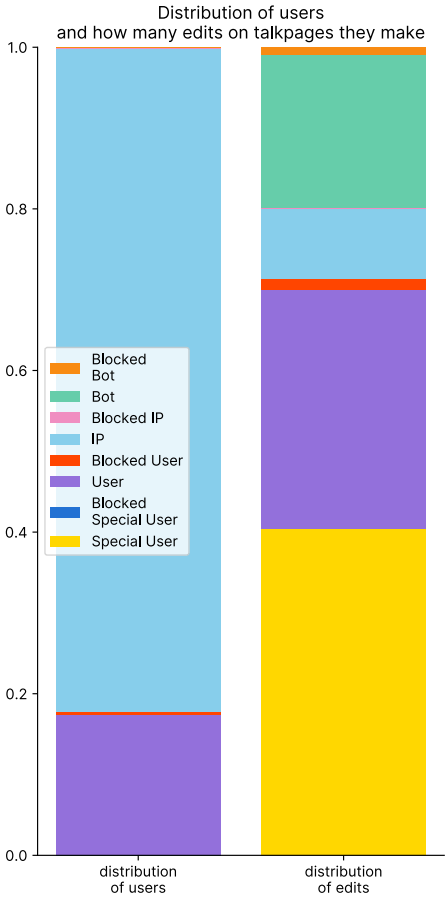

Figure 1: Distribution of users across groups and the total number of edits made by each group. While IP users constitute the bulk of talk page editors, they edit only a small fraction of pages.

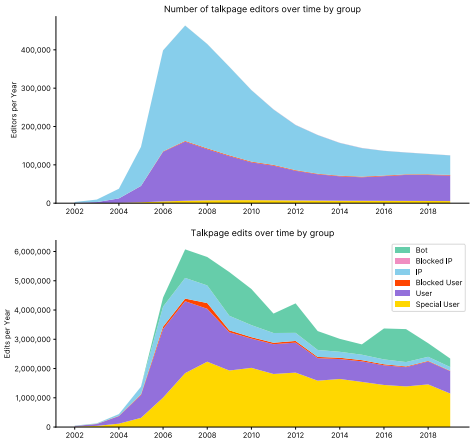

Figure 2: Number of active editors on talk pages across time by user group (above) compared with the total number of talk page edits per year by user group (below).

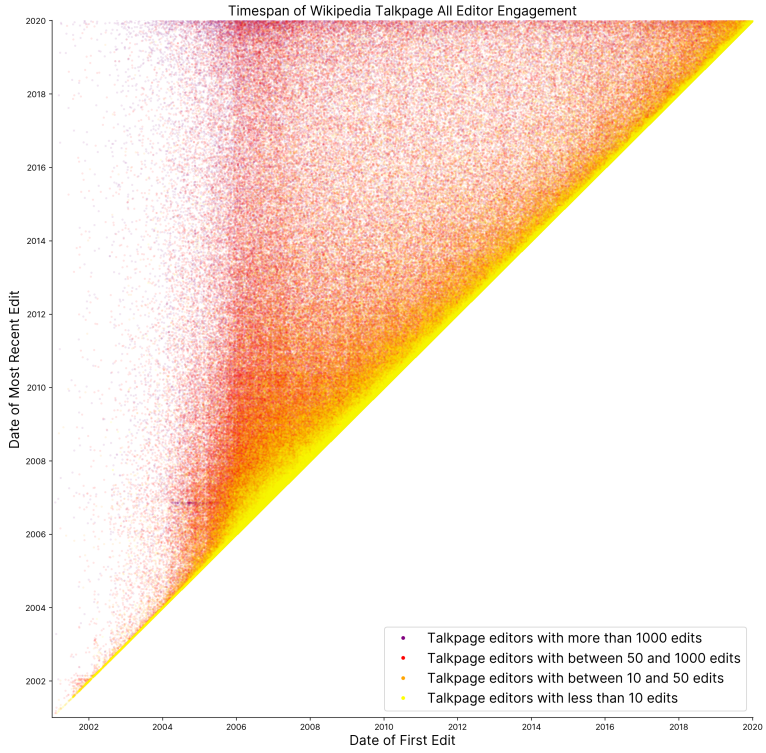

Figure 3: User engagement with talk pages plotted as their first and most recent edits. Users who do not engage long with talk pages are depicted along the diagonal. Each user's number of edits is shown by the hue of their point.

- **page**, which stores information about pages such as the title, namespace, and number of revisions;
- **user**, which stores user information and overall metrics for them, including number of edits and reverted edits across all namespaces;
- and **partition**, which provides bookkeeping to keep track of NSDB processing tasks.

Error handling was important as we expected high variation in content. Errors were logged to a file unless parsing was stopped, in which case they were logged to the database. Partitions which failed were manually restarted and the cause of failure was investigated and fixed.

Coordinating these processes is `nsdb`, the main application. It downloads a list of current XML dumps and starts the process of downloading, splitting, and processing in parallel. It allows the user to specify the location of the temporary data dump and partition files, the maximum space it should use, and the number of cores it should keep available for other processes on the system.

Depending on the timeline needed for NSDB's parsing, NSDB may be run independently or in combination with a Slurm Workload Manager [20] and Python multiprocessing to increase parallelization. Slurm may be used to distribute `nsdb` on several nodes, while `nsdb` itself uses multiprocessing to create several processes of `splitwiki` and `parse` on the same node.

3.2 Performance

All processing was performed on namespace 1, the article talk space. This is a namespace that only 10% of users edit and receives far less editing than main space. We parsed the publicly available April 1, 2020 dump release, which consisted of 660 individual XML files and resulted in 75,355 partitions. The resulting database was 83GB in total, with the edit table being the bulk of the database, at 68GB.

3.2.1 Consumer Hardware. NSDB was developed using a laptop with 8GB RAM, 8 cores, and limited hard drive space. Due to space constraints, only one dump could be processed at a time. However, we found that NSDB was able to process approximately 2–3 partitions per minute in this environment.

3.2.2 Ingest at Scale. In order to speed up processing, we used external computing resources in addition to some occasional processing locally. We used 6 compute nodes, with 14 cores on each. One thread was reserved for `nsdb`, 3 for `splitwiki` and 10 for `parse`. Using our final diagnostics, we averaged 7 partitions per minute for 132 hours. On average, we processed 5 dumps per hour (660 dumps total). We saw at least a 100 times reduction in database size compared with dump size.

3.2.3 Consumer Trade-offs. Based on our initial benchmarks, we predict that the database could be created using a single personal

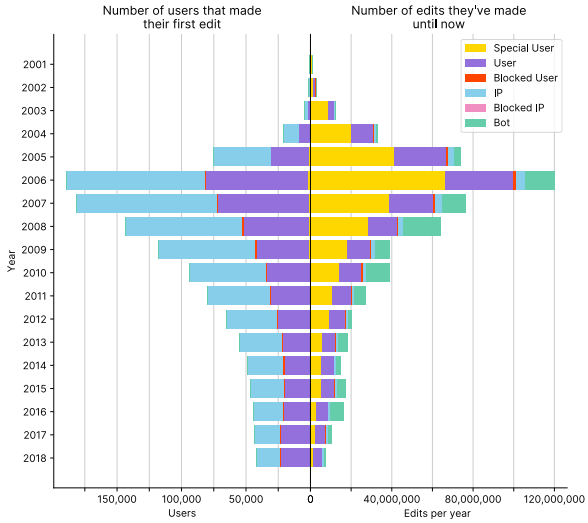

Figure 4: Comparing the number of users who made their first talk page edit in a given year with how many edits those same users make in the future. The users who began in 2006 have made approximately 120M edits since their first interaction; a majority of those edits were made by special users even though they constitute only a small fraction (1.2%) of users who began that year.

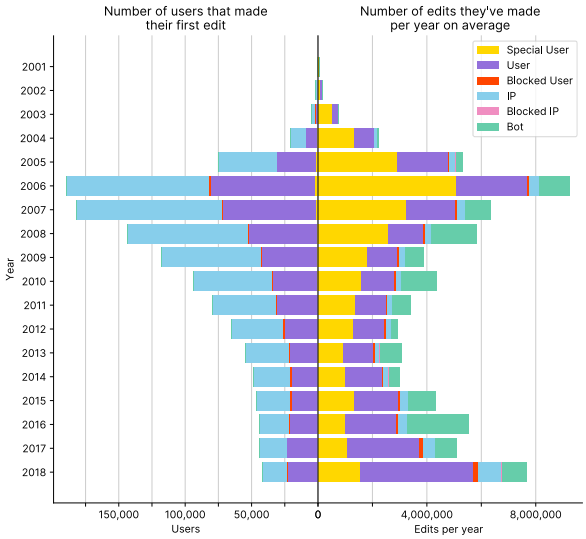

Figure 5: Comparing the number of users who made their first talk page edit in a given year with how many edits they make per year on average. The users who began in 2018 and 2006 appear to be among the most active groups.

computer with several weeks of constant processing. Disk space is the main bottleneck on consumer hardware, as there needs to be 100GB of space free per dump due to the brief doubling of space required during partitioning. Utilizing external disks or focusing either only on a sample of the dumps or a category of pages may significantly increase compute time and reduce the time needed to complete the parsing.

3.3 Extensions and Research Accessibility

NSDB provides a turn-key approach to extracting, simplifying, and reducing the size of the Wikipedia XML data dumps into a manageable MySQL database, confirming our first research question, RQ1. By exploiting the inherent parallelism of the download, extraction, and feature computation tasks, we have created a solution that scales from space-limited personal computers to server-level hardware. However, further benchmarking is still required on mid-level hardware, such as current consumer desktops, to determine the processing time on those systems. Additionally, we did not investigate the size and time taken to parse the main article namespace. However, an estimate from the diagnostics we performed would suggest it would be at least 10 times slower and produce a database that is at least 10 times larger.

4 VISUALIZING TALK PAGE DATA

After creating the database of edits, we investigated user groups and user engagement over the history of Wikipedia. Our second research question, RQ2, asks if visualizations of this data can provide insights into differentiating user groups and their behaviors over time. To begin addressing this question, we chose six groups of interest to focus on in our visualizations and analysis. From largest to smallest, they are:

- IP users, which are not logged in;
- Users, which are logged in;
- Blocked users, which are users that have been blocked;
- Blocked IP users, which are users that are not logged in but the IP address they are using has been blocked;
- Special users, which are users that have been given privileges (such as being able to protect pages or rollback many edits at one time); and
- Bots, which are accounts which are operated automatically.

Figure 1 shows the breakdown of these user groups in the database. We can see that there exists a reverse correlation between group size and number of talk page edits made. The most substantial group of editors is the IP users with 70% of the total, however they make only 9% of the edits. One of the smallest groups, special users, creates the most edits. Even though blocked users are a small subsection of the overall user population, they make a surprisingly large number of edits.

As we begin to incorporate time into the visualizations, Figure 2 temporally extends the bar charts of both user composition and number of edits from Figure 1. These area graphs depict the number

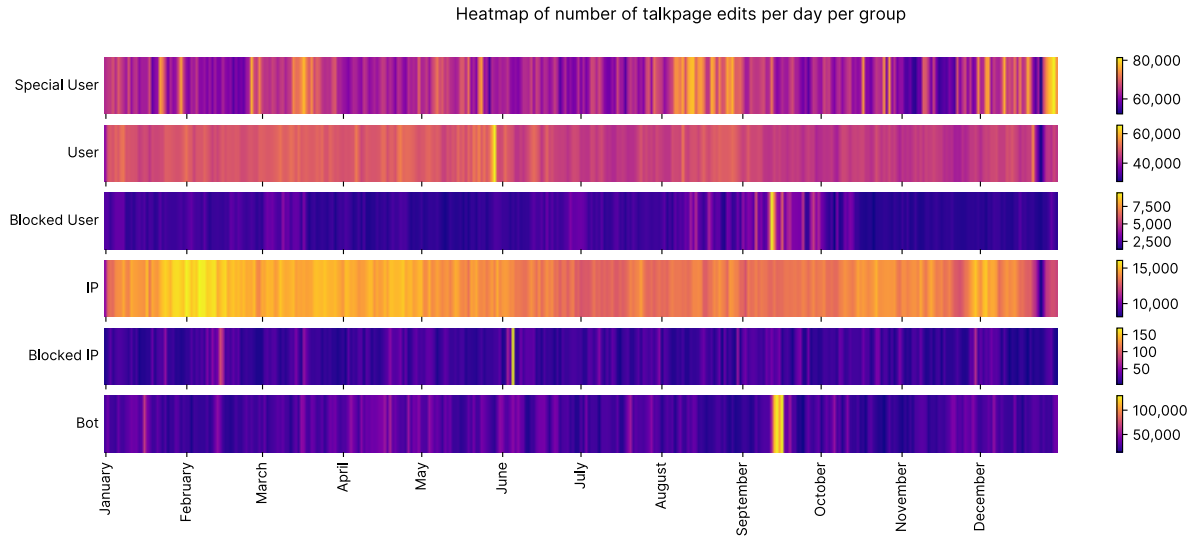

Figure 6: Heatmap showing the activity of user groups per day of the year.

of editors active each year along with their group classification and the number of edits made overall each year. This demonstrates that talk page activity peaked in 2007 and the number of engaged users has steadily plateaued. The number of edits made has decreased in all groups, besides bots, but the activity of special users appears to be decreasing at a slower rate which shows that they are remaining engaged.

4.1 Capturing Time and History

Figure 3, inspired by the work of Bégin et al. [2], was created to investigate the duration that users are active on Wikipedia's talk pages. The first and last edits of each user are plotted, with the total number of edits they have made encoded by the hue of the point. In this visualization, we see many users with relatively few edits along the diagonal, denoting that they made an edit but did not stay engaged with the site. A dense group of long-time editors, with large numbers of edits through 2020, created their accounts in 2006–2008 and have continued editing since. However, it appears that users who began editing after 2008 do not engage with the site as long. Some posit that the decline in the following years was due to the automated edit filters that were set up, which turned people away from Wikipedia [7][17].

To continue investigating the vertical bands of extended engagement from users who started editing around 2006, we created a second novel visualization of similar data in Figure 4. It presents a portion of the data from the previous visualization to connect the number of users who started editing in each year with the number of edits they have created since.

Specifically, the left-hand side of the plot shows how many users made their first edit per year. Directly connected to that bar on the right-hand side, we show how many edits their accounts have made until now. For example, with uniform data, such as if 100 accounts were created each year and each account made 100 edits per year, the left-hand side would show a constant 100 for each year. However, the right-hand side would decrease since the accounts in each successive year would have less time to edit. We do not observe this trend in the actual data. The editors who began in 2006 have made the most edits, surpassing those who came before them, and appearing to have an out-sized role compared with future editors as well. We surmise that this highlights the very active group of core users, which is also evident in Figure 3. The height on the right in this figure is decreasing, but not uniformly as expected; there is a plateau from 2013 to 2016, which demonstrates an increase in editing by some new users.

In order to understand these trends, our visualization includes the breakdown of user groups. Even though there are a few special users who started editing in 2006, they have contributed the most edits of all user groups. Specifically, these 2,344 users have made 66.2M of the 120.3M edits of all users who started in 2006. Conversely, in the plateau of 2013 to 2016, while the number of user and special user edits decreased, the amount of bot edits increased.

Another interesting trend noted in this visualization is the composition of user groups compared with their edit behaviors. A bulk of the first edits, shown on the left-hand side, are edits by IP addresses–specifically, users who are not logged in. They consisted of a majority of first edits from 2005 through 2015. Since 2016, a positive trend has emerged showing that more editors are creating accounts to make their first edits rather than editing via IP. In all cases, the IP users do not have longevity on the site, since even though they are a majority of first edits each year, they are a small fraction of the continued future edits of that year.

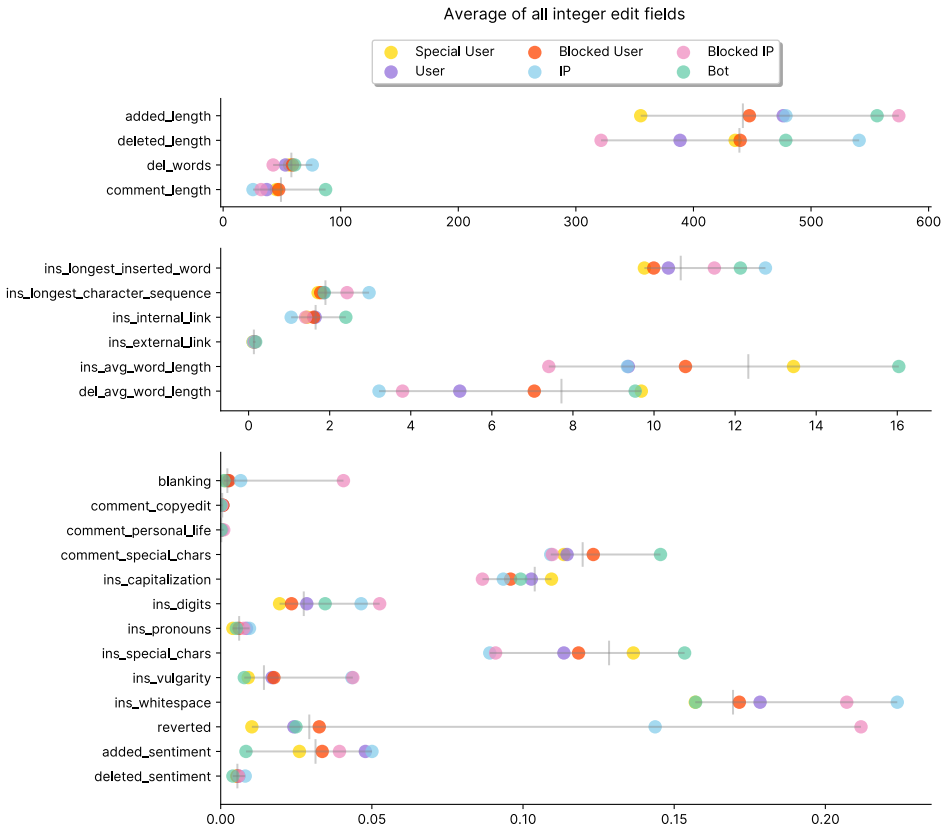

Figure 7: Average value of each feature taken from talk page edits, displayed per group.

Figure 5 provides a slightly different view, replacing the total number of future edits for each year on the right-hand side with the average edits per year. This visualization removes the confounding problem of decreasing future time inherent in the previous figure. For our simple example of 100 users making 100 edits per year, this view would make both sides constant: 100 users starting each year with each yearly set of users making 10,000 edits per year on average. Through this diagram, we see the out-sized role of the special users in 2006; they make more edits per year on average than each group of users make per year from 2009 through 2016. In fact, it highlights the special user engagement across all the years, since they make a substantial number of edits even though they are only a small fraction of the new users in any given year. In 2013, users began to overtake special users by average number of edits per year. Users who started editing in 2017 and 2018 make the most edits compared with other types of users that started the same year. This may denote a shift towards more engagement and edits by users rather than their special user counterparts, or perhaps these users are a part of the next cohort to be given permissions.

Conversely, the IP users' edits average out as we look backwards in time through the figure. Indeed, Figure 4 shows that IP users do not make a large number of edits overall relative to other users each year; that is, they appear to make only a handful of edits per IP address. Therefore, this phenomenon is expected, as the IP users will account for a proportional number of edits in the year they started, but very few in subsequent years.

Next, we take a closer look at the time that users in each group edit and engage with talk pages. Specifically, the days of the year that different groups of users edit provides another important metric of each group's engagement. Figure 6 shows a heatmap across time of how many edits on average each group makes per day of the year. Each group has a drastically different editing pattern, however they share some similarities as well. We see that there is a consistent drop in editing across all the groups over the holidays in late December, with the largest drop being in IP users. There is a peak in blocked activity in September, which coincides with the start of the new academic year. Plots which are mostly blue or yellow have high variance between high and low days of editing respectfully. We see that special users and users have low variance implying that they frequently edit regardless of the day of the year.

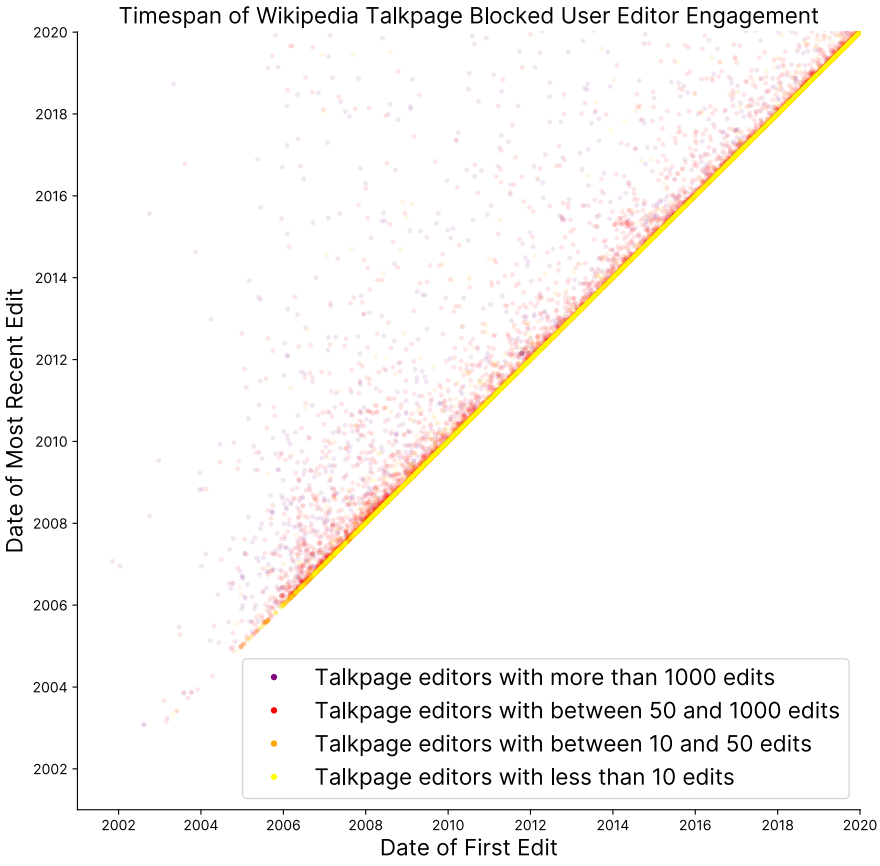

Figure 8: Blocked users tend to engage only for short periods of time, as depicted by the heavy-weight diagonal.

4.2 Capturing Group Edit Features

Moving away from temporal visualizations, we averaged the calculated features for each group's talk page edits. Figure 7 is a connected dot plot which shows each group's average value for every feature. The range is shown with a horizontal grey bar for clarity and the average value is depicted by a vertical line. This plot depicts how each user type engages with talk pages, evoking the possibility of a rubric to measure users against these group trends. For example, the plot shows that on average blocked IP users and bots added the most content per edit, whereas special users added the least. In contrast, blocked IP users delete the least per edit on average while non-blocked IP users and bots delete more per edit. Some distributions are more difficult to interpret, like inserted capitalization, than others, like blanking—where blocked IPs and IPs tend to remove the entire contents of talk pages the most.

From this chart we can tell that edits that we think of as spam, such as high vulgarity and high reversions, are mainly made by IP and blocked IP users. Blocked users are sometimes higher in these regards, but in general are very similar to users. Also, special users, or users with privileges, aren't regularly found on the extremes like we might expect, or when they are, it is hard to justify why that would be the case. The only feature that they are on the extreme of, which follows intuition, is they are reverted the least. As they edit the most, they generally pull the all-user average towards them. Finally, it seems that on average all user groups add content with positive sentiment. This suggests that people are generally positive and nice when talking to each other on talk pages.

4.3 Discussion

Across our visualizations of user edit history and edit features, trends emerge among the different groups of users. Special users and bots tend to make the bulk of the edits compared with general users and IP users. Blocked users and bots tend to edit more in mid-September, while special users, users, and IP users tend to have a constant stream of edits all year. Likewise, blocked IP users tend to be reverted more, insert shorter words, and make longer edits overall. We can affirm that these visualizations help differentiate these groups and highlight their differences, confirming RQ2.

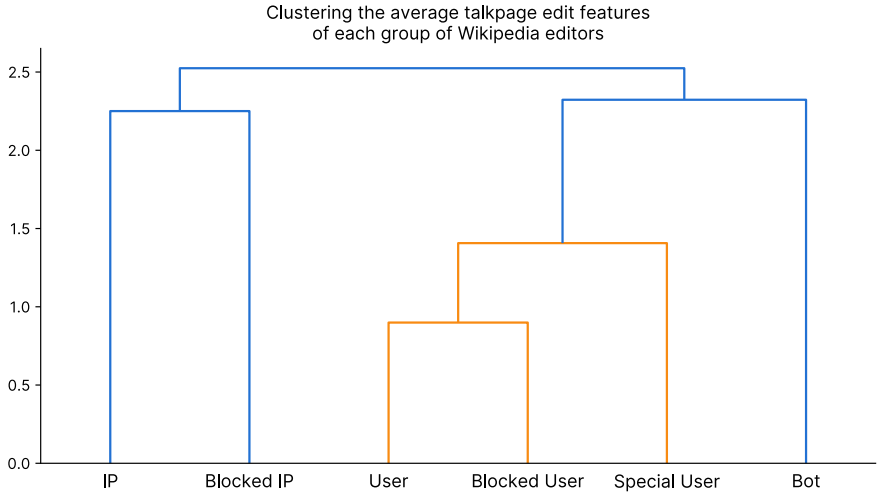

Figure 9: A dendrogram, using the centroid distance function, showing the similarity of talk page editing features of different groups. Users and blocked users are shown to be the most similar groups.

5 CASE STUDY

Building on the trends apparent in our visualizations, we introduce a case study to address RQ3, in which we investigate whether blocked users can be classified based upon features calculated from their edits alone.

First, to get a signature of their edits over time, we isolated blocked users from Figure 3 to create Figure 8. We see that blocked users tend to edit only for a short time and make few edits overall. This may be due to being blocked from editing quickly, or it might be indicative of their edit habits. To further investigate, we attempted to linearly classify blocked users using the average feature values for the group as shown in Figure 7. By combining the average feature values of blocked users compared to other groups, and the timelines typical of these users, we anticipated that a signature would emerge that would identify users that should be blocked. Users that have been blocked on average add less than their unblocked counterparts, while deleting more text, use smaller longest words, include more special characters with less white space, and use text with a lower computed sentiment. Unfortunately, we found that the average values for each of these features were not indicative of the individual members of the group—there was a high variance among the members that confounded the ability of the classifier to correctly identify blocked users.

We verified the difficulty in classification using clustering as depicted in the dendrogram shown in Figure 9. A dendrogram [5] is a visual representation of a clustering algorithm, where at each step the closest points in N-dimensional space are clustered together and depicted as siblings on the visualization. As a side-effect, since nodes which are clustered earlier are more similar, their connections appear lower in the visualization. The normalized average features for each user group were used as the 23-dimensional input and then clustered. We note that users and blocked users are shown to be the most similar, on average, and therefore harder to separate and classify. This is also the case for IP and blocked IP users, however the distance between them is greater.

We then hypothesized that, as users would not have edited after they were blocked, their most recent edits would potentially be the cause of their blocking. Thus, we instead focused on the feature values of the users' last five edits and how that differs from their averages. In order to ensure data quality, we only considered users that had made 10 or more talk page edits; this resulted in a total user population of 223, 722, consisting of 4, 314 blocked users. Figure 10 depicts the averages of features for the blocked users of this smaller population against their overall feature averages. We see a higher amount of adding and deleting of text in blocked users' last edits, along with shorter average word lengths, more vulgarity and white space, and a significant spike in reversions. However, even though these trends are apparent when comparing the change in blocked user behavior shortly before their last edit, we see similar behaviors among other user groups as well. Figure 11 plots the difference between the average feature value and the average value of the last five edits for all user groups. Even though the blocked user's edit patterns change, they do not appear to evidence drastically different behavior compared with other groups. Therefore, this difference is not indicative enough to use as a signature to classify a user as one who should be blocked, and we can answer RQ3 in the negative.

6 FINDINGS

It is clear how important 2006 and 2007 were for gaining high impact and dedicated users that continue editing to this day. Overall, the number of talk page editors has plateaued over the last decade, but unfortunately the total number of edits per year has declined since. However, we can see promising trends of new, productive editors joining.

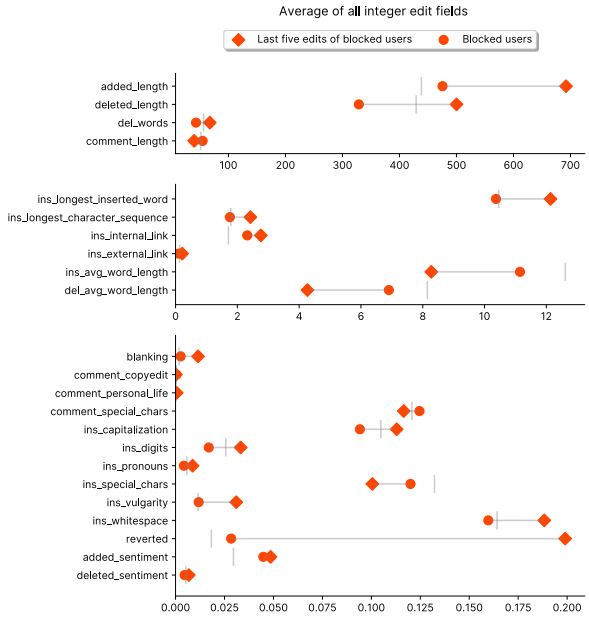

Figure 10: Comparing the features of blocked user's last five edits versus their average. Blocked user's edits are reverted more before they are blocked.

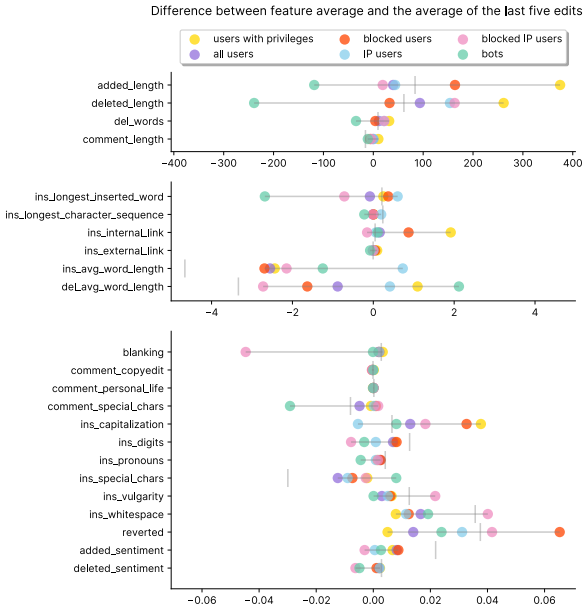

Figure 11: The difference in feature values between users' last five edits for each group and their average for all edits. The blocked users last five edits do not differ from their averages enough to make them an outlier.

We see a decrease in IP editing with a corresponding increase in user editing, suggesting that more editors are choosing to edit using an account. This is hopeful, as the quality of user edits are likely to be higher quality than those of IP editors.

Blocked users seem to edit similarly to users, which suggests they are not often blocked for "spammy" editing. This means that we cannot classify these editors easily based on their editing trends (RQ3); more advanced techniques or detailed data would be required to accurately classify them. Furthermore, a large majority of blocked users do not tend to edit for long, which is positive as it would be concerning to see many long standing and productive users getting blocked.

7 FUTURE WORK

We envision three specific avenues for future exploration of Wikipedia data using our techniques: improved classification methods, incorporating additional namespaces, and including the social domain of WikiProjects.

We will investigate more complex classification techniques, such as utilizing a Multi-layer Neural Network and computing additional features over the edit history, to effectively classify users—especially blocked users. It would also be beneficial to include features which consider additional semantic content and keywords of the edit. Secondly, we will expand benchmarking of our processes to include additional namespaces, specifically main space. By using this larger dataset, we would incorporate more user edit details, but it comes at a storage cost trade-off if inserted and deleted text for each edit is included. Expanding to main space would additionally require careful consideration of calculated features before the extraction is performed due to the computation time needed. Finally, we will investigate how WikiProjects change over time and whether there are positive or negative trends in the data. The guidance that results from these projects would help create a better encyclopedia for us all.

8 CONCLUSION

We created evocative and informative visualizations of user engagement on Wikipedia talk pages through the lens of user groups. These visualizations allowed us to see the role that special users have played throughout Wikipedia history, as well as the strong and continually engaged communities that began editing in 2006. Additionally, we show that while talk page engagement peaked in 2007 and has declined since, we can see a promising trend of rising engagement from new users in the past 4 years.

As part of this process, we have detailed a scalable method of performing research, utilizing the public XML data dumps containing the full Wikipedia edit history. We created the Namespace Database tool to create a MySQL database of Wikipedia edits for any Wikipedia namespace, category, or set of pages. This tool is fully documented and published under an open license, as so to be extensible and fork-able for future development.

Finally, we attempted to use our extracted features of user edits to classify blocked users. However, while these features provided

insight into how different groups of users engage with Wikipedia talk pages, they were insufficient to isolate the editing habits of blocked users. We surmise that a more complex model or more detailed dataset would be needed to successfully classify these users.

ACKNOWLEDGMENTS

Aaron Halfaker and Lane Rasberry were key to this project's success, and we are grateful for the support that they both provided.

REFERENCES

[1] Alex Woodson. 2007. Wikipedia remains go-to site for online news. https://www.reuters.com/article/us-media-wikipedia/wikipedia-remains-go-to-site-for-online-news-idUSN0819429120070708. [Accessed 5-May-2020].

[2] Daniel Bégin, Rodolphe Devillers, and Stéphane Roche. 2018. The life cycle of contributors in collaborative online communities—the case of OpenStreetMap. *International Journal of Geographical Information Science* 32, 8 (2018), 1611–1630. https://doi.org/10.1080/13658816.2018.1458312 arXiv:https://doi.org/10.1080/13658816.2018.1458312

[3] Anamika Chhabra, Rishemjit Kaur, and S. R.S. Iyengar. 2020. Dynamics of Edit War Sequences in Wikipedia. In *Proceedings of the 16th International Symposium on Open Collaboration* (Virtual conference, Spain) *(OpenSym 2020)*. Association for Computing Machinery, New York, NY, USA, Article 8, 10 pages. https://doi.org/10.1145/3412569.3412585

[4] Christine de Kock and Andreas Vlachos. 2021. I Beg to Differ: A study of constructive disagreement in online conversations. arXiv:2101.10917 [cs.CL]

[5] Brian Everitt. 1998. *The Cambridge Dictionary of Statistics*. Cambridge University Press, Cambridge, UK ; New York.

[6] Aaron Halfaker. 2017. Mediawiki-utilities/mwxml - MediaWiki. https://www.mediawiki.org/wiki/Mediawiki-utilities/mwxml [Online; accessed 8-May-2020].

[7] Aaron Halfaker, R. Stuart Geiger, Jonathan Morgan, and John Riedl. 2013. The Rise and Decline of an Open Collaboration System: How Wikipedia's reaction to sudden popularity is causing its decline. *American Behavioral Scientist* 57, 5 (May 2013), 664–688. https://doi.org/10.1177/0002764212469365

[8] Sara Javanmardi, David W McDonald, and Cristina V Lopes. 2011. Vandalism detection in Wikipedia: a high-performing, feature-rich model and its reduction through Lasso. In *Proceedings of the 7th International Symposium on Wikis and Open Collaboration*. ACM, 82–90.

[9] Carlin MacKenzie. 2020. Namespace Database - A tool to create a database of Wikipedia edits. https://www.github.com/carlinmack/NamespaceDatabase/. https://doi.org/10.5281/zenodo.3817987

[10] Martin von Gagern. 2014. GNU Wdiff. https://www.gnu.org/software/wdiff/. [Accessed 8-May-2020].

[11] Sergio Martinez-Ortuno, Deepak Menghani, and Lars Roemheld. 2014. Sentiment as a Predictor of Wikipedia Editor Activity. (2014).

[12] Paolo Massa. 2011. Social networks of wikipedia. In *Proceedings of the 22nd ACM conference on Hypertext and hypermedia*. 221–230. https://www.gnuband.org/papers/social_networks_of_wikipedia/

[13] David McCandless. 2020. Wikipedia's lamest edit wars. https://informationisbeautiful.net/visualizations/wikipedia-lamest-edit-wars/

[14] Charu Rawat, Arnab Sarkar, Sameer Singh, Rafael Alvarado, and Lane Rasberry. 2019. Automatic Detection of Online Abuse and Analysis of Problematic Users in Wikipedia. In *2019 Systems and Information Engineering Design Symposium (SIEDS)*. IEEE. https://doi.org/10.1109/sieds.2019.8735592

[15] Jodi Schneider, John G Breslin, and Alexandre Passant. 2010. A content analysis: How Wikipedia talk pages are used. (2010).

[16] Julian Seward. 1996. bzip2 and libbzip2. http://sourceware.org/bzip2/ [Online; accessed 20-June-2021].

[17] Bongwon Suh, Gregorio Convertino, Ed H. Chi, and Peter Pirolli. 2009. The Singularity is Not near: Slowing Growth of Wikipedia. In *Proceedings of the 5th International Symposium on Wikis and Open Collaboration* (Orlando, Florida) *(WikiSym '09)*. Association for Computing Machinery, New York, NY, USA, Article 8, 10 pages. https://doi.org/10.1145/1641309.1641322

[18] Dario Taraborelli and Giovanni Luca Ciampaglia. 2010. Beyond Notability. Collective Deliberation on Content Inclusion in Wikipedia. In *2010 Fourth IEEE International Conference on Self-Adaptive and Self-Organizing Systems Workshop*. 122–125. https://doi.org/10.1109/SASOW.2010.26

[19] Wikipedia contributors. 2021. Wikipedia:Database download. https://en.wikipedia.org/wiki/Wikipedia:Database_download [Online; accessed 13-June-2021].

[20] Andy B. Yoo, Morris A. Jette, and Mark Grondona. 2003. SLURM: Simple Linux Utility for Resource Management. In *Job Scheduling Strategies for Parallel Processing*, Dror Feitelson, Larry Rudolph, and Uwe Schwiegelshohn (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 44–60.

[21] Justine Zhang, Jonathan Chang, Cristian Danescu-Niculescu-Mizil, Lucas Dixon, Yiqing Hua, Dario Taraborelli, and Nithum Thain. 2018. Conversations Gone Awry: Detecting Early Signs of Conversational Failure. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Melbourne, Australia, 1350–1361. https://doi.org/10.18653/v1/P18-1125

A APPENDIX

A.1 Composition of special users

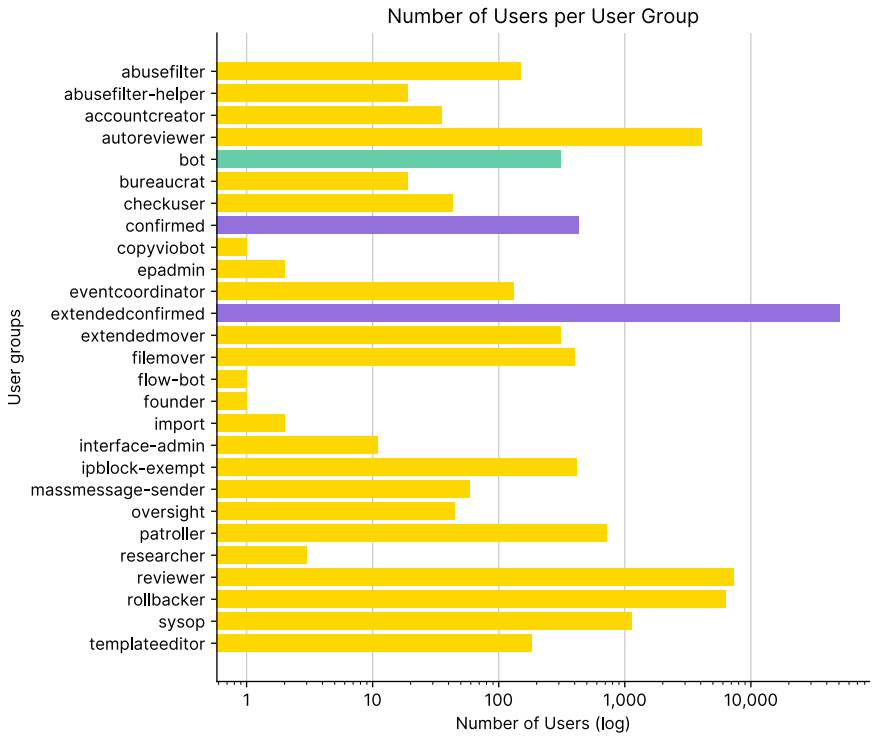

Figure 12: A plot of the number of users which have each privilege. Note the log scale when interpreting this graph. The color of the bar represents which of our groups that user belongs to.

Figure 12 illustrates the number of users with certain privileges on Wikipedia. This is the only plot which does not source from our database, but instead from the user group assignments dump[5]. All groups are positive and denote a privilege that the user can subsequently do. Details of the groups can be found on the "User access levels" page[6].

[5]https://dumps.wikimedia.org/
[6]https://en.wikipedia.org/wiki/Wikipedia:User_access_levels