CS-E4870
Research Project in Machine Learning and Data Science

Ananth Mahadevan

Department of Computer Science, Aalto University
ananth.mahadevan@aalto.fi

Abstract

1 Introduction

Wikipedia is the largest online encyclopedia containing over 5 million pages of content. It is one the most popular websites on the Internet. Wikipedia has a diverse collection of articles from many different topics and is constantly being updated. Although Wikipedia started out as an open platform where anyone could create and articles, this lead to many factual errors and biased articles. Wikipedia started to incorporate elements hierarchy gradually over time. In the English version of Wikipedia all editors need to have a registered account and pages that are controversial and of a sensitive nature are protected by administrators.

Administrators are editors who are given access to tools such as blocking and unblocking other users, deleting and undeleting pages, protecting and renaming pages etc. Any user can **Request for Adminship**(RfA) in which the Wikipedia community participates. The RfA spans over seven days, during which any editor can comment and discuss the candidate. Editors scrutinize the candidate's contributions and credentials as well their conduct in the online discussion and overall experience. They can then state either their support or opposition to the candidate along with comments. At the end of seven days a Bureaucrat (an editor higher up in the hierarchy) decides on the consensus of the election and declare the outcome. Consensus is not a direct majority voting scheme and the final call rests with the Bureaucrat.

The RfA is a very intense and selective process, there are only 1400 total administrators of which only 500 are currently active[1]. This is out of 38 million registered editors with only around 130 thousand are regular contributors. This small group of active administrators and editors are responsible for creating and maintaining all articles on Wikipedia.

Therefore the RfA process can give us valuable insight into the dynamics of social interactions and elections in an online platform. In this paper we will first discuss the existing work on studying the RfA elections and other such similar online processes. Next we provide an overview of the data collected and used from Wikipedia in this paper. We then present our main contribution, the use of a *Viscous Democracy* to model the RfA election process. We discuss the results and possible extensions of this framework to other online elections systems.

2 Literature review

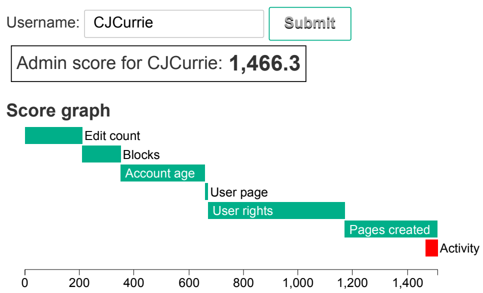
The Wikipedia RfA process has been widely studied in various domains from many different perspectives such as those of the candidate, the voters, the community etc. In this section we discuss the existing work in this field.

Administrator is a highly coveted status on Wikipedia and there are many features than can be used to determine the worthiness of a candidate. Wikipedia themselves provide tools and guides[2] to help potential candidate assess their own electability. Wikipedia's *admin score tool* as seen in Figure 1 uses features such as edit counts, pages created, age of account etc. Similarly, Burke et al. [2] utilized past RfAs to find features that correlate highly with success such as presence of edit summaries, politeness in user interactions and varied experience. Such tools and models are useful for finding potential nominees and understanding what the community values and respects. This however doesn't offer

[1] all data as of March 2020 for English version Wikipedia

[2] http://en.wikipedia.org/wiki/Wikipedia:GRFA

any insights into the dynamics that might play out in any particular election



Score components

Component	Data	Score
Edit count	62,616	+210.95
Blocks	never blocked	+140.00
Account age	5,808.3 days (15.91 years)	+310.04
User page	exists	+10.00
User rights	sysop	+500.00
Pages created	5,697 article-space pages created	+340.36
Activity	14.4 edits per month, on average (over the last year)	-45.00
CSD log	missing	+0.00

Figure 1: Admin score tool for user CJCurrie and it's breakdown

Leskovec et al. provide a thorough analysis of the election from the perspective of the voter. They show that the voters make decisions based on *relative assessment* of merit and degree of correspondence with the candidate. Elections do not follow a *herd mentality* and standard information cascades. We see an interesting result that voters have diverse personal response functions as well as admin and non-admin patterns of voting differ. [6] We get a detailed picture of the temporal dynamics in a RfA.

As the votes in an RfA election can be positive or negative they can form a *signed network* which has been studied and analyzed in great detail. We see that the Wikipedia RfA network has more compliance with status theory compared to balance theory in Leskovec et al. [8]. When Leskovec et al. try and use these signed structural properties to predict edges in [7], they see that the predictive accuracy is poor for Wikipedia RfA network compared to the other networks used. However as signed edge prediction methods are designed to work with any generic signed network, they tend to discard information that RfAs are elections and play out in a timely manner. Also predicting a single edge i.e a vote does not increase the accuracy in predicting the result of an election.

The work of Desai et al. [3] is related closely with the contributions presented in this paper. They use linear models for regression and classification to identify a core of *influential voters* through feature selection. Using a set of 40 most influential voters they are able to predict the result of an election with a high accuracy. They also collect additional network features of the voters independent from the elections. Their results do not improve significantly in using the additional features in predicting election results. These results show that there are a group of influential voters that determine elections. This will be more evident when we analyze the dataset in coming sections.

- explain RfA data collection
  - existing SNAP data and limitations
  - XML parsing
  - regex and string matching
  - date parsing
- Social interactions
  - User contributions
  - wealth and diversity of info
  - creating underlying network

3 Dataset

We would like to have two different types of data to help build the election model in this paper. The first would be information of the votes cast in a RfA and the eventual result of the RfA. This gives us the users interactions in an online election process where they have to judge their peers. The second, is information on the interactions of users in other non-elections settings. In Wikipedia discussions occur in *Talk Pages*. Every type of Wikipedia page (articles, user pages, help pages etc) has a *Talk Page* where users can discuss the contents of that article or interact with the user or provide information to others. These are valuable data sources to gather more details on user activities.

In this section we will discuss the existing Wikipeida datasets from Stanford large network datasets (SNAP) [9] that satisfy our requirements and their inherent limitations. Next we will illustrate the process by which we collected newer data from Wikipedia.

3.1 Existing Datasets

For the first type of data we requore thre are two existing Wikipedia RfA datasets in SNAP namely *wiki-Elect* a nd *wiki-Rfa* . They both contain attributes of each vote in a RfA such as the source, target, vote , result of RfA, timestamp. The *wiki-Rfa* is a more recent version of the *wiki-Elect* dataset. It has RfAs till May 2013 and also has the comment text of voters. There are 11,000 users and around 190 thousand votes in total. Both of these datasets have been used in many previous works mostly as signed networks. There are a few limitations of these datasets when we would like to analyze them as vote cast in an election. There is more than 5% of *wiki-Rfa* votes that have no timestamp and almost 1% of votes that have no source. As most RfAs have fewer than 300 votes this is an issue when considering the sequence of votes as well as who has cast a vote.

The interactions between users outside of RfA elections is useful to understand behaviour and perceptions of others. Wikipedia users can directly interact with another user by writing on their *User Talk Page*. This can be a measure of how much correspondence exists between two users. We saw how this is a good indication of probability of supporting a candidate for an election [6]. The *wiki-Talk* dataset on SNAP contains a directed network where an edge from node u to v signifies that u has written in v's talk page. This dataset is a large network containing more than 2 million nodes and 5 million edges. The limitation of this network data is that nodes do not have user id mapping and also the edges are not weighted. Without a node to user id mapping the network canot be used with the election data. Having weighted edges tells us how many times a user has interacted with someone else which is more informative.

Due to the limitations of the existing datasets we set out to collect our own data to build our election model.

3.2 RfA Data Collection

We went through a 60GB XML dump of Wikipedia from Jan 2019 to extract the RfA data. We chose to scrape the data in a format similar to the SNAP *wiki-Rfa* dataset. The outline of the data extraction process is illustrated in Figure 2. In the first step we filter out all Wiki pages whose title doesn't contain the term `Requests for adminship`. This still leaves us with a lot of non-election Wiki pages, so we can further filter by checking for terms such as `Category:Unsuccessful requests` or `Category:Successful requests`.Now this reduces the space from over 5 million pages to the roughly 4000 pages related to RfA elections.

The next step is to process the body of the election pages individually and extract votes from the *wiki-Talk* , Wikipedia's own markup syntax. After locating the `Support`, `Oppose` and `Neutral` sections we can extract the individual votes. This step is particularly hard as *wiki-Talk* syntax changes through the years and there is no fixed page structure. The user's comment can also nested discussion threads which we chose to not extract. As user vote are ended with a signature, i.e their user id and timestamp. The timestamps also have varied syntaxes adding to the overall complexity of this extraction phase. Using more robust regular expressions to capture multiple timestamp formats and also handling a myraid of edgge cases in processing we acheive a much higher coverage of election votes.

We collected $226,781$ votes from $4,557$ elections with over $13,000$ unique user ids. Only 1.6% of votes have missing timestamps and 0.4% have a missing source. We also added unique id (UID) field to differentiate candidates who stood for elections multiple times.

3.3 User Interaction Data Collection

Wikipedia has an API to request all the contributions made by a particular user [10]. This offers a rich source of data on the activities of a user on Wikipedia as seen in Figure 6 from the online *editsummary tool*[3] for Wikipedia users. We proceeded to collect the contributions for every unique user in the RfA data querying the API. There are some issues with the user ids that are present in the RfA data. As a single user can have multiple aliases and/or change their user id at any point, some users might not have any contributions under an alias that has been discontinued. To simplify our data colection, we assume that each user id is a unique user and will fetch contributions under that user id if present. This resulted in 100GB of data for nearly $11,000$ out of $13,000$ user ids. We can see that edits have a *namespace* as seen in 6b. These are categories for each Wiki page like `Main` is for all the articles on Wikipedia and `User Talk` is available for each user. Each category also has a the corresponding *Talk Page* for discussions. Therefore, we can get user interactions by looking at the `User Talk` namespace. As an example in 6d, the top edited user talk page is of `Dianna` (The actual user is `Justlettersandnumbers`, hence the top results are edits on their own page). This allows

[3] `https://xtools.wmflabs.org/editsummary`

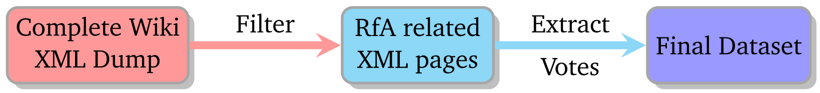


Figure 2: RfA Data Collection Process

us to create a dataset similar to the *wiki-Talk* dataset, with user id mappings as well as count of number of interactions. The data on top edited pages as seen in Figure 6c can also be used to create a profile of a user's diversity or speciality of topics and much more.

4 Analysis

In this section we analyze the datasets and present some general statistics and trends from the datasets described in the previous section.

4.1 RfA statistics

In Figure 7 we can statistics of elections in Wikipedia that show some interesting trends. First, in Figure 7a we see that the average number of votes in elections is increasing with time. This is as expected as initial RfA were just confirmation processes for candidates who were qualified. As the years went by the process starts to get more involves. This is seen in Figure 7c, where there is a peak in the number of successful and unsuccessful elections around 2008 and since then there are fewer election and in total fewer successful ones. A pattern that is good to note is that in the distribution of votes we see that there is a clear majority of support votes in Figure 7b, the interesting fact is that when we see the average number of words in comments in Figure 7d we see that support votes have much fewer words compared to oppose or neutral votes. This indicated that people who are casting support votes might have small positive comments, while people casting negative or neutral votes tend to write larger comments to convince others of the issues that they find in the candidacy.

4.2 Influential Voters

To find if there is a set of voters in elections who are influential we utilized two approaches. The first is performing feature selection using a gradient boosting model on the whole dataset as done by Desai et al. in [3]. The second approach is similar to finding a set cover.

For the first approach we created a dataset where each column corresponds to an election and each columns is one user. Therefore as we have 4548 elections and nearly 13,000 users so the data matrix is roughly $X \in \mathbb{R}^{4548\times 13000}$ and the target is the result of the election, therefore $y_i \in \{1, -1\}$ and $y \in \mathbb{R}^{4548}$. As most users don't vote in all elections the data matrix is very sparse. Unlike Desai et al. [3], we did not fill the unknown votes with 0, we left them as missing values. This is because the XGBoost model is able to handle missing values. After fitting the model with the data we extracted the top 15 features based the *gain* they bring to the model in Figure 3. These top 15 users can be thought of as the most influential voters for prediciting an election.

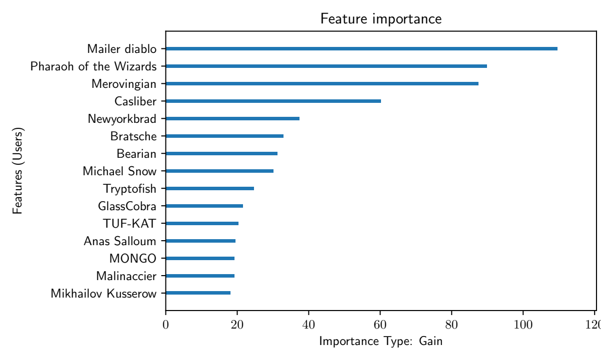


Figure 3: XGBoost feature importance

The second approach was formulating a set cover problem, every element of the ground set is a tuple of (voter, election) and then we create a subset for each unique user. For every user we take each election they participated in and add the people who follow that user. This means that they voted after the user and also voted the same as the user. Therefore the set S_u for every user is defined as

$$S_u = \{(v, e) \mid \text{where } v \text{ voted the same after } u \text{ in election } e\}$$

Then we order the subsets S_u by their size and then try to find how much of the ground set we can cover by taking the top k users' subsets. The ground set has 221,766 elements, which is fewer than the total number of votes as there are certain elections where votes were duplicated or people voted twice. We see how the set cover increases as we increase the value of k as seen in Figure 4 We see that with only 200 top users we can cover

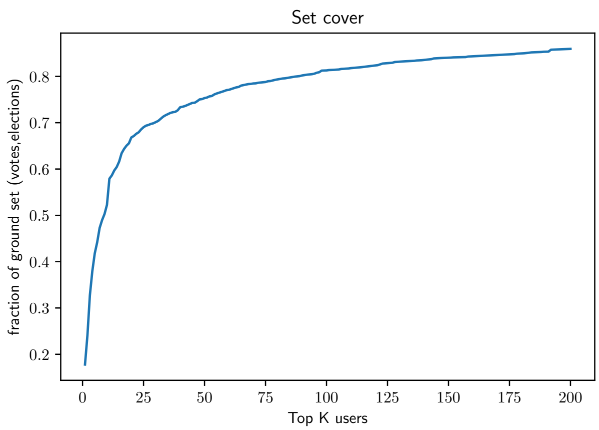


Figure 4: Election set cover

nearly 85% of the whole ground set. More interestingly we see that there is a knee around 25 users indicating that there is a small core of influential users. With the top 15 users we have 60% coverage.

Ranking	XGBoost	Set Cover
1	Mailer diablo	Siva1979
2	Pharaoh of the Wizards	Mailer diablo
3	Merovingian	Newyorkbrad
4	Casliber	Wizardman
5	Newyorkbrad	Pedro
6	Bratsche	Dlohcierekim
7	Bearian	Juliancolton
8	Michael Snow	Casliber
9	Tryptofish	Acalamari
10	GlassCobra	Fastily

Table 1: Top 10 influential users from XGBoost and the Set Cover models

In Table 1 we see many common users among the top 10 influential users from both approaches. This confirms the fact that in Wikipedia RfA elections there is a core set of voters that can be important in predicting the result of an election.

5 Viscous Democracy Model

In this section we explain the concept of viscous democracy [1] and it's relevance in decision making on online platforms. We then go on to explain how we can use the concept of viscous democracy model to create a model to predict the results of Wikipedia RfA elections

5.1 Concept

The difference between **direct democracy**, **representative democracy** and **liquid democracy**.

Direct democracy is when all the people are involved directly in deciding on any policy. it is the purest form of democracy and can only function well in relatively small social groups, as when the number of people cross a certain limit it is logistically infeasible to include everyone's opinion. Representative democracy (or indirect democracy) when people choose a representative who will carry out policy decisions in their favour. This system is widely used by most major countries as well social groups. The main drawback of this system of democracy is that the representatives are not obligated to fulfill promises to the society and once in power can further their self interest. The representatives also stay in power for a period of time within which the public's stance or view on policies might change.

Liquid democracy is in-between direct and representative democracy. It can be called **delegative democracy** as people have the chance to delegate their vote to a *proxy* or choose to vote directly. This way the proxy's vote is augmented with all the delegations one receives and can also be transitive in that the proxy can again delegate to another individual. This style of proxy voting works due to **strong transitive delegation**. This means that I am more likely to trust my friend's friend to have similar interest as my own. There is a lot of theoretical and practical research ongoing in this field [5, 4]. The liquid democracy model is not directly suitable for online settings. This is because the social ties online are weak and therefore the transitivity of delegation is weak. It just means that you are less likely to trust your Facebook friend's other Facebook friend.

There is a *reluctance* to delegate in online communities. Hence the weight of the vote attenuates as it is delegated further down a chain. This can be visualized as the vote being delegated to be viscous and therefore the weight of your vote when delegated again is smaller than it was before. This is the main concept behind *viscous democracy*. Every voting model requires a *ballot* and a *tally* which we will now describe.

5.2 Ballot

A ballot is how a voter expresses their preferences. In the examples of direct and indirect democracy the ballot is usually cast in the form of a vote, either directly for a policy or indirectly for a candidate. The form of voting can be a **one person, one vote** system where the voter indicates one option. The other is **ranking based voting** such as Instant-runoff voting (IRV)[4] or Single Transferable Vote (STV)[5] where the voter ranks the options in the order of preference. There can also be ballot systems where you rate the options with a score. In the

case of delegative voting systems such as liquid or viscous democracy the choices are to either vote directly or choose to delegate to another person. If we restrict ourselves to only a single vote model then we can consider the ballot as making a *delegation graph*. This concept is illustrated in Figure 5. We assume that we have an underlying undirected social graph. Every node is a person and each edge indicates a connection. This can be like the friends network in Facebook or contacts networks in LinkedIn etc. Then the delegation graph is built upon the same nodes of the social graph where each node can either vote directly, leading to self loops or choose to delegate to one of their neighbors. Therefore if we assume there is some *delegation rule* that each node follows then the delegation graph is induced from the social graph by applying that rule.

5.3 Tally

6 Implementation

directed graph concepts and delegation function considerations. Agony and hierarchy. local and global top k delegates.

7 Results

The quality of predictions using local or global important editors.

8 Conclusions

How we can instead try and model individual voter behaviour. Find a more robust ML framework to learn an optimal delegation function.

[4] `https://en.wikipedia.org/wiki/Instant-runoff_voting`

[5] `https://en.wikipedia.org/wiki/Single_transferable_vote`

References

[1] Paolo Boldi, Francesco Bonchi, Carlos Castillo, and Sebastiano Vigna. Viscous democracy for social networks. *Commun. ACM*, 54(6):129–137, June 2011.

[2] Moira Burke and Robert Kraut. Mopping up: Modeling wikipedia promotion decisions. In *Proceedings of the 2008 ACM Conference on Computer Supported Cooperative Work*, CSCW '08, page 27–36, New York, NY, USA, 2008. Association for Computing Machinery.

[3] Nikhil Desai, Raymond Liu, and Catherine Mullings. Result prediction of wikipedia administrator elections based on network features. 2014.

[4] Steve Hardt and Lia CR Lopes. Google votes: A liquid democracy experiment on a corporate social network. 2015.

[5] Anson Kahng, Simon Mackenzie, and Ariel D Procaccia. Liquid democracy: An algorithmic perspective. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[6] Jure Leskovec, Daniel Huttenlocher, and Jon Kleinberg. Governance in social media: A case study of the wikipedia promotion process. In *Fourth International AAAI Conference on Weblogs and Social Media*, 2010.

[7] Jure Leskovec, Daniel Huttenlocher, and Jon Kleinberg. Predicting positive and negative links in online social networks. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, page 641–650, New York, NY, USA, 2010. Association for Computing Machinery.

[8] Jure Leskovec, Daniel Huttenlocher, and Jon Kleinberg. Signed networks in social media. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, page 1361–1370, New York, NY, USA, 2010. Association for Computing Machinery.

[9] Jure Leskovec and Andrej Krevl. SNAP Datasets: Stanford large network dataset collection. `http://snap.stanford.edu/data`, June 2014.

[10] MediaWiki. Api:usercontribs — mediawiki, the free wiki engine, 2019. [Online; accessed 22-March-2020].

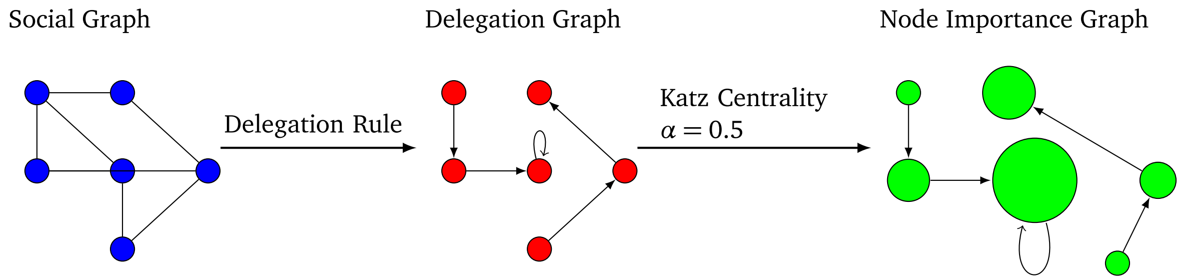


Figure 5: Viscous Democracy Model

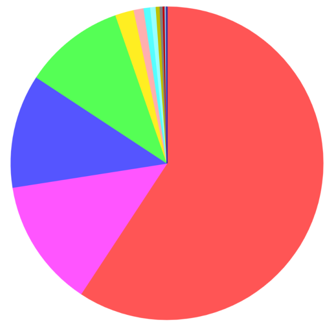
9 Appendix

Edits in the past 24 hours:	29
Edits in the past 7 days:	158
Edits in the past 30 days:	717
Edits in the past 365 days:	9,040
Average edits per day:	21.6 (3,275 days)
Average edit size*:	231.7 bytes
Minor edits:	5,397 · (8.3%)
Small edits (<20 bytes)*:	2,192 · (43.8%)
Large edits (>1000 bytes)*:	1,087 · (21.7%)

(a) edit statistics

Namespace Totals [hide]

Namespace	Count
Main	38,710 (59.3%)
User talk	8,651 (13.2%)
Wikipedia	7,658 (11.7%)
Talk	6,804 (10.4%)
User	1,257 (1.9%)
Template	672 (1%)
Wikipedia talk	445 (0.7%)
Draft	372 (0.6%)
Category	259 (0.4%)
Template talk	203 (0.3%)
File	129 (0.2%)
Draft talk	84 (0.1%)
Help talk	16 (0%)
Help	13 (0%)
Category talk	12 (0%)
MediaWiki talk	8 (0%)
Portal	7 (0%)
File talk	1 (0%)
18 namespaces	65,301



(b) Edit namespace distribution

Top edited pages [hide]

Main [hide]

Edits	Page title	Assessment	Links
274	Central Saint Martins	Start	Log · Page History · Top Edits
89	Donkey	B	Log · Page History · Top Edits
74	University of the Arts London	C	Log · Page History · Top Edits
71	American Pekin	Start	Log · Page History · Top Edits
70	Bobby Lockwood	Stub	Log · Page History · Top Edits
64	Academy of Art University	C	Log · Page History · Top Edits
63	Maremma Sheepdog	Start	Log · Page History · Top Edits
62	Giorgi family	Start	Log · Page History · Top Edits
61	Louise Blouin	Start	Log · Page History · Top Edits

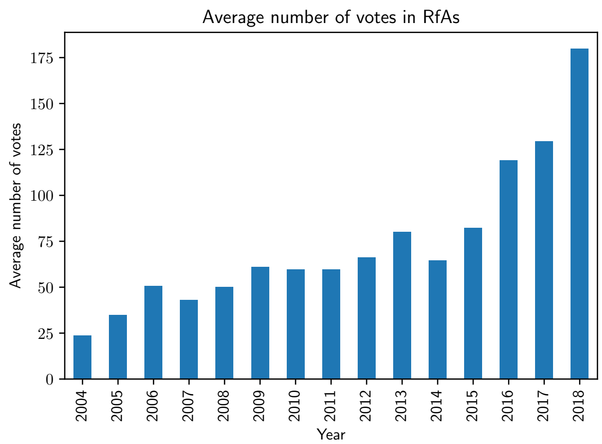
(c) Top edited pages

User talk [hide]

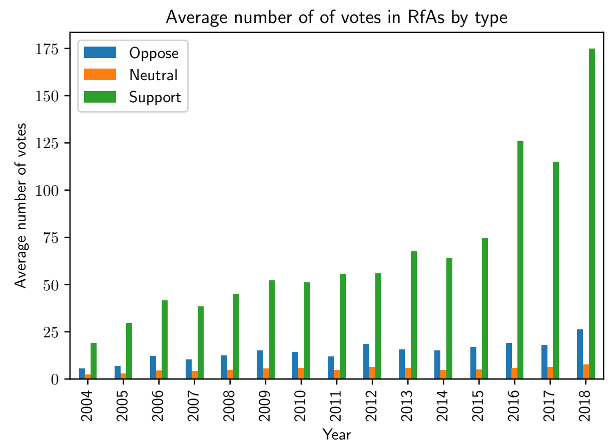
Edits	Page title	Links
520	User talk:Justlettersandnumbers/old2	Log · Page History · Top Edits
273	User talk:Justlettersandnumbers/old3	Log · Page History · Top Edits
204	User talk:Justlettersandnumbers	Log · Page History · Top Edits
188	User talk:Justlettersandnumbers/old4	Log · Page History · Top Edits
93	User talk:Diannaa	Log · Page History · Top Edits
63	User talk:Moonriddengirl	Log · Page History · Top Edits
44	User talk:Montanabw	Log · Page History · Top Edits
35	User talk:Alec Smithson	Log · Page History · Top Edits
21	User talk:Sphilbrick	Log · Page History · Top Edits

(d) Top user talk pages edits

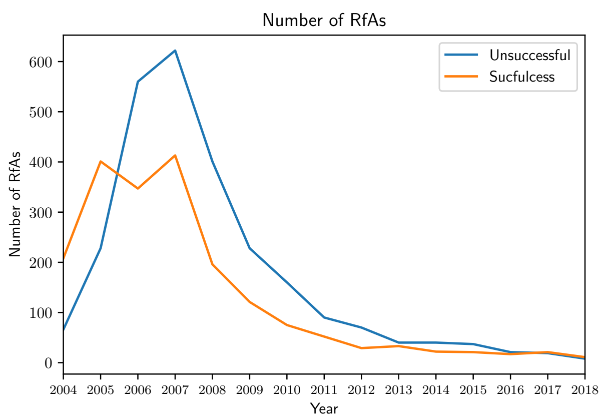
Figure 6: Edit summary of a Wikipedia user



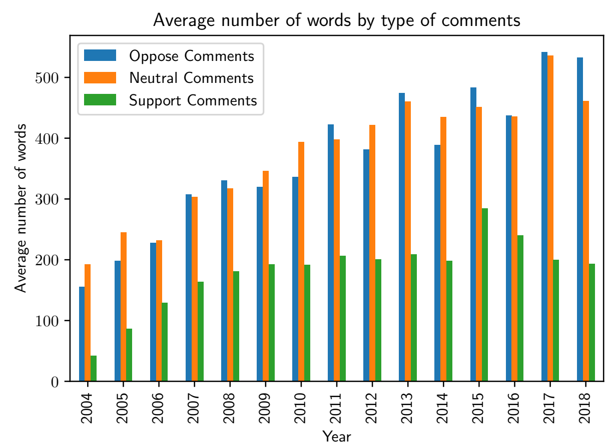
(a) Vote distribution



(b) Vote distribution by tye



(c) Number of RfA by year



(d) Comments distribution

Figure 7: Election Statistics