Viscous Democracy Model for Wikipedia Admin Elections

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 encyclopaedia 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. In their online discussions, 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 for 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 whom only 500 are currently active[1]. This is out of 38 million registered editors with only around 130 000 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 from Wikipedia and how it is used 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

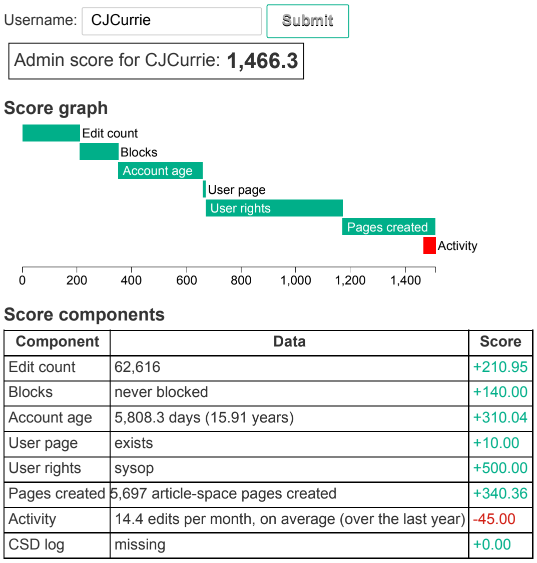
The Wikipedia RfA process has been widely studied in various domains from many different perspectives such as those of the candidate, the voters, the community etc. In this section we discuss the existing work in this field.

Administrator is a highly coveted status on Wikipedia, and there are many features that 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 any insights into the dynamics that might play out in any particular

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

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

election


Username: CJCurrie

Admin score for CJCurrie: **1,466.3**

Score graph

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. [7] 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 analysed in great detail. We see that the Wikipedia RfA network has more compliance with status theory compared to balance theory in Leskovec et al. [9]. When Leskovec et al. try and use these signed structural properties to predict edges in [8], 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 analyse the dataset in coming sections.

3 Dataset

We use 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 provides us with 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.) includes a corresponding *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 Wikipedia datasets from Stanford Network Analysis Project (SNAP) [10] 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, there are two existing Wikipedia RfA datasets in SNAP namely WIKI-ELECT a nd WIKI-RFA. They both contain the attributes of each vote in a RfA, such as the source, target, vote, result of RfA and the timestamp. The WIKI-RFA is a more recent version of the WIKI-ELECT dataset. It contains RfAs till May 2013 and also has the comment text of voters. There are 11 000 users and around 190 000 votes in total present in the dataset. Both datasets have been used in many previous works, mostly as signed networks. When we analyse these datasets as vote cast in an election, there are a couple of limitations. More than 5% of WIKI-RFA votes 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 are useful in understanding their 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. This measure provides a good indication of probability of supporting a candidate for an election [7]. 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 cannot 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 4 000 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 myriad of edge cases in processing we achieve 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 [11]. This offers a rich source of data on the activities of a user on Wikipedia as seen in Figure 12 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 collection, 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 will refer to this dataset USER-CONTRIB from this point.

We can see that edits in USER-CONTRIB have a *namespace* as seen in Figure 12b. 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 Figure 12d, 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 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 12c 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 analyse the datasets and present some general statistics and trends from the datasets described in the previous section.

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

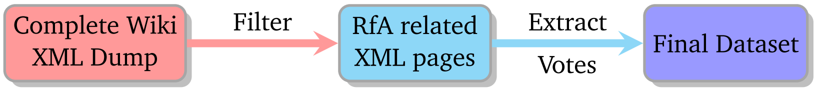


Figure 2: RfA Data Collection Process

4.1 RfA statistics

In Figure 13, we see statistics of elections in Wikipedia that show some interesting trends. First, in Figure 13a 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 13c, 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 13b, the interesting fact is that when we see the average number of words in comments in Figure 13d 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 4 548 elections and nearly 13 000 users so the data matrix is roughly $X \in \mathbb{R}^{4548\times13000}$ 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.

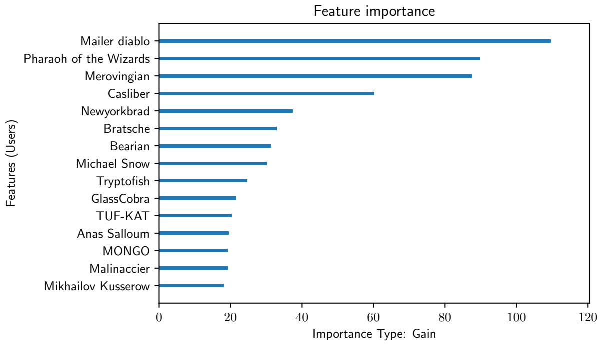


Figure 3: XGBoost feature importance

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 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.

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 who can be important in predicting the result of an election.

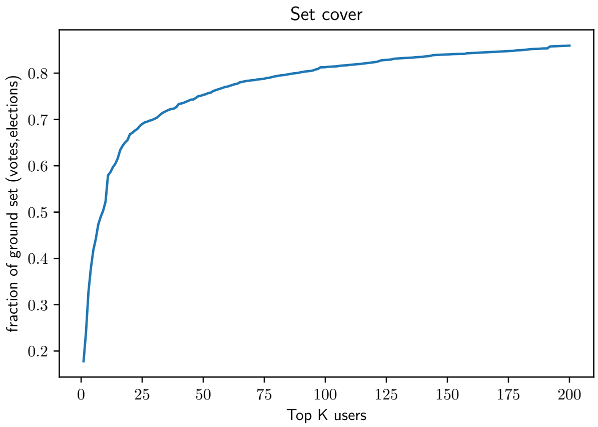


Figure 4: Election set cover

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

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

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

There are three main forms of democracy namely **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 fulfil 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 [6, 5]. 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 that is delegated as viscous and therefore its strength reduces with each additional delegation. 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.

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

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

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* [1]. 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 neighbours. 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

The tally is the algorithm that decides the final outcome. In direct or indirect democracies the method of tally might be **first-past-the-post** an example of plurality based tallying or a proportional system where seats or power is awarded to the proportion of the votes received.

The tally proposed by Boldi et al. is a scored based tally where each node in the network receives a score that is calculated based on the delegation graph [1]. This score can then be used along with any requirement such as choosing a committee or one policy out of many alternatives. The method is called *transitive proxy voting with exponential damping*, where a dampening factor α controls the amount of reluctance in the transitive delegation of votes. They proposed using **Katz's Centrality index** on the delegation graph. Therefore the score for a node i is defined as

$$\text{Score}_i = \sum_{p \in \text{Path}(-,i)} \alpha^{|p|}$$

Here, $\text{Path}(-,i)$ is set of all delegation paths ending in node i and $|p|$ is the length of the path [1]. This step is shown in Figure 5, where the *Node Importance Graph* is the delegation graph where each node's size is proportional to the score that is obtained from Katz's Centrality with $\alpha = 0.5$. Here we see that the nodes where the delegation chain ends have a larger node score as the votes are transitively transferred.

The parameter $\alpha \in (0,1)$ acts as the *delegation factor* and $1-\alpha$ is the *viscosity*. As α approaches 0 the strength of votes delegated tend to 0 hence direct votes have more weight. Therefore the more viscous the system is the more analogous it is to direct democracy or majority voting. As α approaches 1 almost all the weight of the vote is transferred to the delegate and the node scores correspond to the size of the subtree that they are a part of and the vote and the system is more analogous to liquid democracy [1].

5.4 Model

Now utilizing all the concepts behind *viscous democracy* we propose a model to predict the Wikipedia RfA elections. This election model also consists of the two parts, namely ballot and tally.

We can utilize the interactions between users on Wikipedia to create the underlying **social graph** that is required. Once we have a social graph we can use the many features from the user's contributions to implement a **delegation rule**. Applying the delegation rule upon the social graph we obtain the delegation graph. Then choosing an appropriate value of α and from Katz's centrality on the delegation graph we obtain scores for each node which corresponds to a user. We can then use the scores to tally votes in an election to predict the result using two approaches.

As we saw in Section 4.2 a small core of voters are important in the result of an election. We can determine these influential voters either on a local scale or a global scale. On a local scale we can choose k most influential voters in an election and then tally their votes to predict the election. On a global scale we can choose the top k influential voters overall and then only tally their votes in a particular election to find the result. Both these approaches aim to use a smaller set of voters to effectively predict the result of an election.

6 Implementation

In this section we discuss the implementation of various parts the viscous democracy voting model. First we explain the construction and details of the social graph of Wikipedia editors. Next we describe the different approaches for the delegation rule. Lastly we see the details of the local and global tally methods.

6.1 Social Graph

As we described in Section 3.3 the USER-CONTRIB dataset has comprehensive edit history for each user. We use the *User Talk Page* edits to create an edge in the social graph. The nodes are limited to the users from the

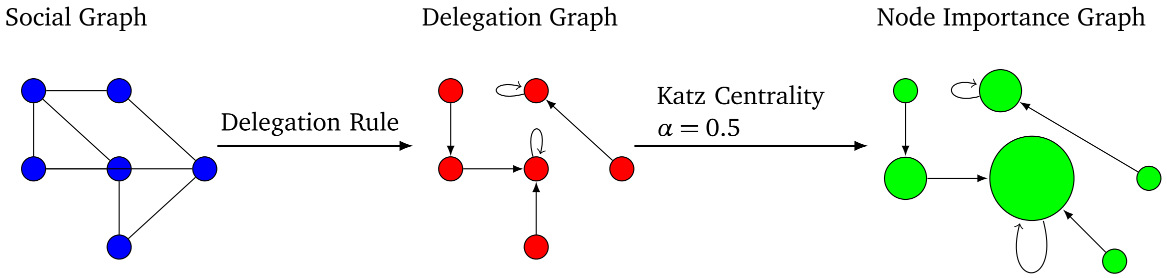


Figure 5: Viscous Democracy Model

Property	**Value**
Number of Node	12,529
Number of Edges	1,149,415
Density	0.00732
Largest connected component size	10,565

Table 2: Social Graph properties

RfA dataset so that the size of the graph is not unnecessarily large. An edge exists between user U and user V if U has edited V's talk page. This provides us an underlying directed social interaction graph. The network properties are show in Figure 2.

We see that the size of the social graph is smaller than the total number of unique users. This is due to many user changing names or accounts being inactive. The graph is also fairly well connected with a large connected components and all others singleton components indicating temporary or one time users. The graph is inherently directed in nature and the successors and predecessors of a node can provide information of who the has contacted or who have contacted the node respectively. if we convert the graph into an undirected network the neighbours of a node U is the union of the successors and predecessors. We will use this social graph as the basis of neighbours or contacts for voters in the viscous democracy model.

6.2 Delegation Rule

The most important part of the model is the delegation rule. In their work, Boldi et al. mention that delegation usually happens within the community and that once delegation occurs you can use the viscous democracy to evaluate the scores of each node in the graph. The difficulty is that when using this model to simulate an election we require a heuristic by which people delegate within their neighbourhood. Boldi et al. in their simulation of voting in a co-authorship network used the criteria that a voter would delegate to the person in their neighbourhood who has published more papers, if none exists then they would vote for themselves, choosing not to delegate themselves [1].

Therefore in our simulation of Wikipedia RfA election we can use attributes of each voter to decide how the delegation would be carried out. We recorded the following information for each node[6] in the social graph

1. Start date of account
2. Total number of edits
3. Ranking

Each of these will result in a different delegation rule which are **seniority**, **edit count** and **rank**. Given a node u we will define the neighbourhood of the node as $\mathscr{N}_u$. As mentioned previously the nodes in the neighbourhood depends on whether the graph is directed or undirected. The augmented neighbourhood is defined as appending the node u to the neighbourhood, $\mathscr{N}_u' = \mathscr{N}_u \cup u$. We will explain each rule and how it will be applied to the social graph.

6.2.1 Seniority Rule

This rule will be based on the start date of each node. The delegation will be done to the node that has the most seniority or has the lowest starting date. If the function StartDate(v) would return the start date of node v then the function can be written as follows where $\text{delegate}_\text{S}(\cdot)$ is the delegation rule based on seniority.

$$\text{delegate}_\text{S}(u) = \underset{v \in \mathscr{N}_u'}{\arg\min}\,(\text{StartDate}(v))$$

[6] the terms node, voter and users will be used interchangeably

This delegation rule is based on the heuristic that people who have been in Wikipedia longer are better placed to make a decision on the administrative qualities of a candidate.

6.2.2 Edit Count Rule

The heuristic is based on the fact that the users on Wikipedia who have more edits are usually more active in the community and are also the ones whose votes will be most influential in swaying other voters to support a candidate. Hence given a node u we take the augmented neighbourhood $\mathcal{N}_u'$ and then choose the node which has the highest number of total edits. In this way, if there is no neighbour who has more edits than the node u then, the user votes for herself. If we define the function $\mathrm{EditCount}(v)$ to return the total number of edits for user v and $\mathrm{delegate}_\mathrm{E}$ as the delegation function based on edit count, we have

$$\mathrm{delegate}_\mathrm{E}(u) = \operatorname*{arg\,max}_{v \in \mathcal{N}_u'} (\mathrm{EditCount}(v))$$

6.2.3 Rank Rule

If there exists a hierarchy in the social network then we can use that as a heuristic upon which to delegate votes. It would work the same way that the other delegation rules have been defined. Given a node we delegate to the neighbour who has the highest rank, if not then the node votes for herself. This style of voting is particularly useful in societies where is there is some organizational structure. The issue is that there is no such defined structural hierarchy amongst Wikipedia editors. Even the administrators themselves are no different than editors with access to tools to help with their roles. Therefore the first requirement is to find a hierarchy in the directed social graph.

There is already existing work in finding hierarchies in directed networks using concepts such as the concept of *agony* [12, 4]. If each node in a directed graph $G = (V, E)$ is given a ranking, formally defined as $r : V \to \mathbb{N}$ then any edge $u \to v$ causes agony if $r(u) \geq r(v)$ and is equal to difference plus one, i.e., $r(u) - r(v) + 1$. The agony of the whole network G with respect to the ranking r is the summation of the agony caused by every edge $e \in E$. The concept of agony corresponds to people higher up in a social hierarchy do not prefer to interact with people lower down in the hierarchy. This is the reason why edges $u \to v$ where $r(u) < r(v)$ causes no agony. Gupte then goes on to show that the agony of a network corresponds to how close to a Directed Acyclic Graph (DAG) a network is. As DAGs have perfect hierarchies this means that they also correspondingly have an agony of 0 [4]. Therefore, the ranking r is the proposed hierarchy of the network. Gupte provides a method of uncovering this hierarchy in a network by solving the problem of finding the ranking r that minimizes the agony. Tatti provides an algorithm that can solve this problem in polynomial time by solving the dual problem of finding the maximum Eulerian subgraph [12]. We use the algorithm and code of Tatti[7] to find the ranking for the social graph of Wikipedia editors.

This method of finding the hierarchy penalizes edges from high rank to low rank. This means we must also understand the context in the case of the social graph. In our graph an edge $u \to v$ indicates that user u has written in the talk page of v. Therefore, we must ask ourselves if agony is lower ranked users writing on higher ranked user pages or if it is the otherwise around. To not restrict ourselves to one approach we computed the rank for both possibilities. In the regular case this means that agony is when high rank users write on low rank user talk pages. The other case is the reversed social graph where an edge $u \to v$ means that v has written in talk page of u, and here agony is when a lower ranked member writes in a higher ranked user's talk page. We will call the ranks from this reversed social graph as the *reversed rank*.

We see the distribution of the ranks in Figure 6 & 7. The number of levels are different with regular ranking having 8 and reversed rank having 9 levels of hierarchy. In both distributions we see that the highest level of hierarchy has only around a 100 users. This again points to the fact that there exists a small core of influential users in the network based only on their interactions. We can now use the rank that each node had to create 2 distinct delegation rules.

Given a node u we take the augmented neighbourhood $\mathcal{N}_u$ and then choose the node which has the largest rank or reversed rank from the set. If the functions $\mathrm{Rank}(v)$ and $\mathrm{ReversedRank}(v)$ provide the rank and reversed rank of the node v respectively then the delegation functions $\mathrm{delegate}_\mathrm{R}(u)$ and $\mathrm{delegate}_\mathrm{RR}(u)$ can be defined as

$$\mathrm{delegate}_\mathrm{R}(u) = \operatorname*{arg\,max}_{v \in \mathcal{N}_u'} (\mathrm{Rank}(v))$$

$$\mathrm{delegate}_\mathrm{RR}(u) = \operatorname*{arg\,max}_{v \in \mathcal{N}_u'} (\mathrm{ReversedRank}(v))$$

[7] http://users.ics.aalto.fi/ntatti/agony.zip

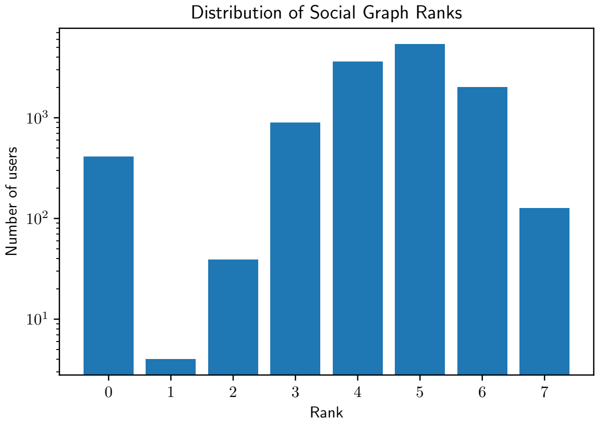


Figure 6: Rank Distribution in Social graph

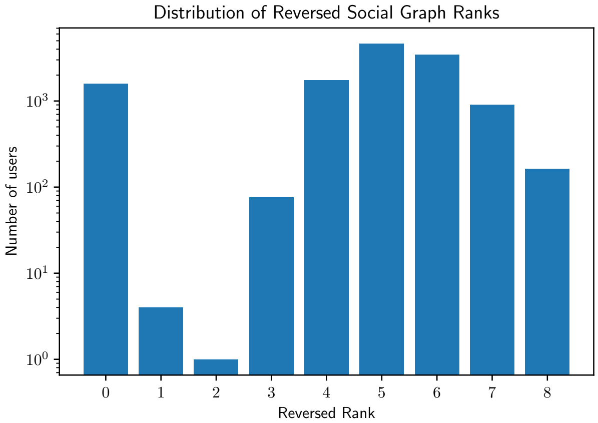


Figure 7: Rank Distribution in Reversed Social graph

6.3 Tally Schemes

Now after implementing the social graph of Wikipedia users and simulating the delegation using the delegation rule we have a delegation graph. From the delegation we can get the node scores by computing Katz's centrality measure with a given value of α. These node scores defined as a function $s : V \to \mathbb{R}$ can be used to predict elections as discussed in Section 5.3. In both approaches we want to select k users and then predict an election by tallying their votes. This can be either in a global context or a local context. We now describe these two approaches.

6.3.1 Global Tally

In the global context we take all the nodes of the social graph and then choose k users with the largest scores obtained from the previous steps. We then go through each particular RfA and then only take the votes corresponding to the chosen k users and then tally their votes. We then predict the election is successful is there are more support votes, unsuccessful if more oppose than support and neutral outcome otherwise. This method aims to find a constant core set of influential users obtained from the viscous democracy model. As most users don't vote in every election, we would need a large set of users to have a good predictive accuracy across all the elections in the dataset. As $k \to N$, where $N = |V|$ of the social graph, we take nearly all users and their votes in each election to predict the results.

6.3.2 Local Tally

The local context of tallying is when we take an RfA election and then of all the users who voted we choose the k with the largest scores and only tally their votes. We consider a support vote as $+1$, oppose vote as -1 and a neutral vote as 0. Then the tally of k votes would be the sum of all the k votes, we predict a successful elections if the tally is positive, unsuccessful if the tally is negative and neutral if the tally is exactly 0. It is evident that in this method that the top k users are not unique and change based on the RfA being predicted. This also means that the size of k could be smaller and might lead to similar and possibly better predictive power compared to the global context.

6.4 Hyperparameters

The viscous democracy model has many parameters and hyperparameters that can be varied. These occur in different stages of the model and affect the final predictive accuracy of the model. In this subsection we will describe all these hyperparameters and their possible values.

- **α** : the delegation factor, $\alpha \in]0, 1[$
- **k** : number of users to consider using node scores, $k \in \mathbb{N}$
- **Delegation Rule**: Seniority, Edit Count, Rank or Reversed Rank
- **Tally Scheme**: Local or Global

We see that there are many possible combinations of parameters and hyperparameters to arrive at a model. In the following section we will present the results of the model.

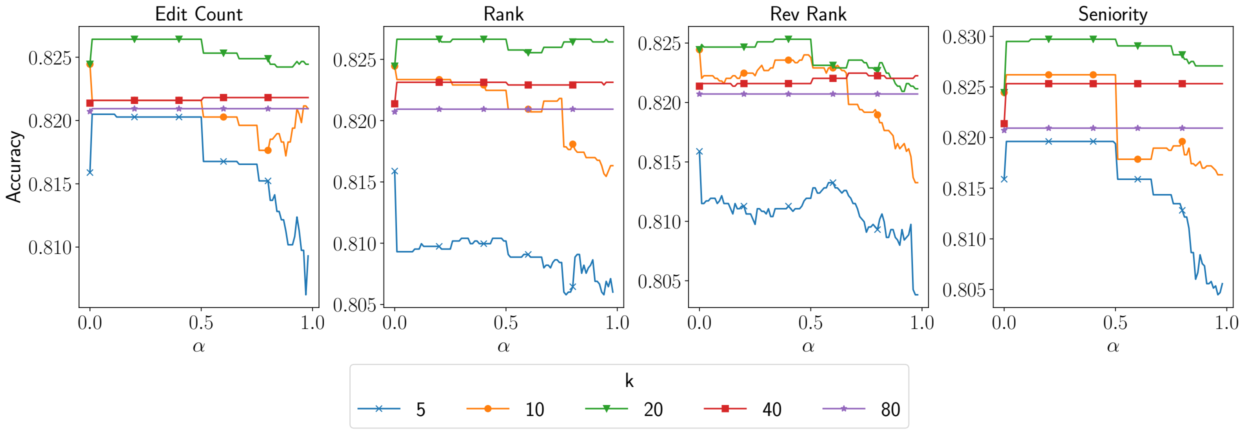


Figure 8: Effect of α for the LOCAL VISCOUS model with different delegation rules and values of k

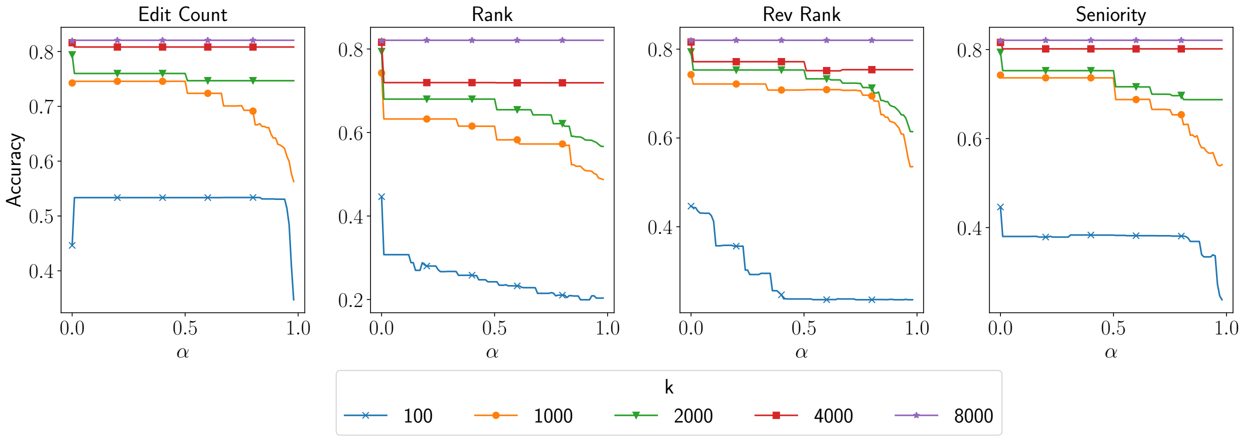


Figure 9: Effect of α for the GLOBAL VISCOUS model with different delegation rules and values of k

7 Results and Discussion

We presents the results of the viscous democracy model and analyze the effects of the various parameters on the accuracy of the model.

The main metric we will be using to compare the quality of the Viscous Model is **accuracy**. The baseline that we will be using to measure the model is a simple tally of all the votes in an RfA election. This compares the most directly with our models as we increase the value of k. As we discussed before generally Wikipedia RfA results are positive if there is at least 65% supporting votes. Therefore we filter out elections that have fewer than 10 votes to get a better view of the predictive quality of the model.The baseline model of just tallying the votes gives an accuracy of **82**%.

There are many parameters that have an effect on the predictive accuracy of the model. This means that the effective size of search space is quite large especially as parameters like α have a continuos domain. To narrow the search space as well gather a better understanding of the effects each parameter has on the model we fix all others and take a closer look in the following subsections. Finally we discuss the results of the grid search over the pruned parameter space to find the best performing model.

7.1 Effect of α

To study the effects of the delegation parameter α we need to fix k, the delegation rule and the tally method. As the range of k depends on the tally method, we split these results firstly as either using the GLOBAL VISCOUS or the LOCAL VISCOUS model.

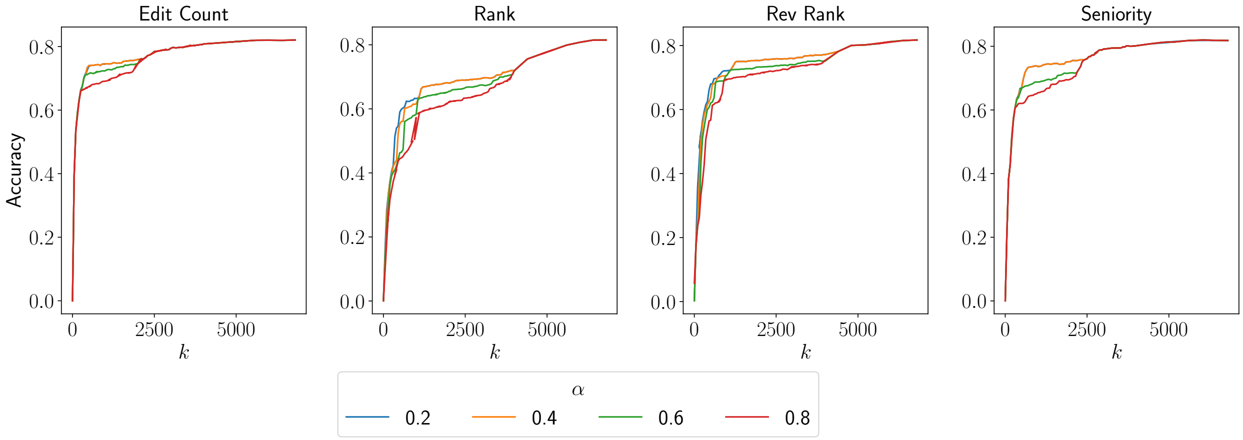


Figure 10: Effect of k for the GLOBAL VISCOUS model with different delegation rules and values of α

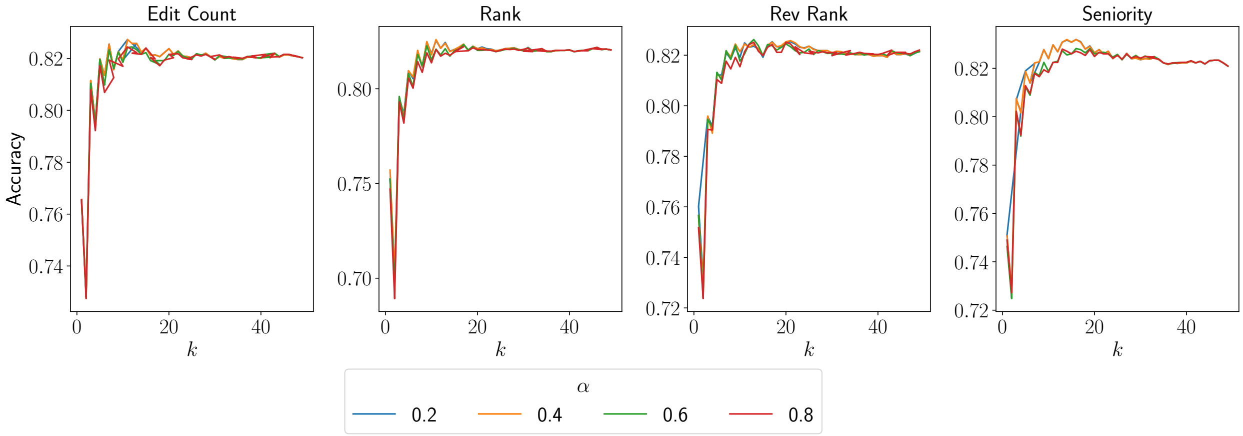


Figure 11: Effect of k for the LOCAL VISCOUS model with different delegation rules and values of α

In the GLOBAL VISCOUS model, $k \in [1, 13000]$ and therefore we pick 5 values of k and plot the accuracy of the model independently using each delegation rule. We see the results in Figure 9. There are some general trends that we can see across all the delegation rules. For small valued of k the quality of predictions are low and as a result the overall accuracy is poor and usually below 50%. This is as expected because in the global tally scheme the top 100 important users might not vote in all elections hence the model has very few votes to tally in each election. For large values of k such as 8000, we have close to two-thirds of the unique voters and then the value of α is irrelevant as the ranking of the top 8000 users changes very little. Almost all votes in an election are used to calculate the tally hence the accuracy is constant close to the baseline 80%. For values of k in the middle, we see that as α increases there is a gradual drop in the accuracy. This is clearly seen in the step like decrease for all delegation rules in Figure 9. This indicates that the GLOBAL VISCOUS model is more viscous in nature and works better for smaller values of α.

When we consider the LOCAL VISCOUS model we see that the value of k is bounded by the number of votes in a particular election. Theoretically we can have a k more than the number of votes in an election, in which case we just choose all the votes to get the tally. The average number of votes in a RfA election is 49 and we chose the following 5 values $k \in \{5, 10, 20, 40, 80\}$ to analyze. We again see that for $k = 5$ is accuracy is low and for $k = 80$ the accuracy is close to the baseline and stable. The reasoning is similar to the GLOBAL VISCOUS model, for small k not enough votes and for large k the α does not change ranking enough to affect the top k

users. The interesting trend is that unlike the GLOBAL VISCOUS model when we increase k to 20 we see that the LOCAL VISCOUS model actually performs much better than larger values of k as well as the baseline. This indicates that around the region $k = 20$ there is additional gain performance in choosing only the important votes to tally. We also see the trend that higher values of α tend to decrease the accuracy, especially for the seniority rule as seen in Figure 8. A point to not is that for most values of k and delegation rules the performance drops significantly at the point $\alpha = 0.5$, this indicates there might be a significant change in the ranking of voters in the LOCAL VISCOUS model. Therefore the local model is also **more viscous than liquid**.

7.2 Effect of k

Similar to how we studied the effect of α above, now we take the two tally methods, delegation rules and fixed values of $\alpha \in \{0.2, 0.4, 0.6, 0.8\}$ to understand the effect of the value of k. Though we have in a way analyzed this in the previous subsection, now we provide a more detailed look of the trends as well as the ranges to choose for k when later performing the grid search. We will again analyze the GLOBAL VISCOUS and LOCAL VISCOUS model separately.

In Figure 11 we see that the GLOBAL VISCOUS model has a knee around $k = 2000$, this is where we see the performance of the model approaching the baseline. It also around this region that the effect of the delegation parameter α is most prominent. In line with the previous analysis, we see that the smaller values of α perform much better than the larger values of α in this region. We can also verify that for both small and large k the change in α has no effect.

We see a similar knee for the LOCAL VISCOUS model in Figure 11, this time just before $k = 20$. In this knee region there are some differences compared to the GLOBAL VISCOUS plots. We see the effect of α is not as pronounced in this region. We also see that there are distinctive spikes that lead to a bump before plateauing back to the baseline as k increases. These spikes confirm the behaviour that we discussed previously in Figure 8, around this region the LOCAL VISCOUS model has slightly better performance than the baseline. This is again clearly evident in the $\alpha = 0.4$ line for the seniority delegation rule in Figure 11.

7.3 Grid Search

Gathering all the information from the previous sections on the effects of the parameter α and k we can narrow down the search space to find the best combination of parameters. As both the LOCAL VISCOUS and GLOBAL VISCOUS models show a tendency to perform better with more viscous votes, we can restrict the space to $\alpha \in [0.2, 0.4]$. The analysis of the GLOBAL VISCOUS model shows that we need nearly $k = 2500$ to get performance close to the baseline. Correspondingly for the LOCAL VISCOUS model we see for around $k = 20$ we start to get much better accuracy compared to the baseline. Using these as starting points for the grid and random search of the parameter space we obtain the following best combination of model parameters as show in Table 3.

Table 3: Results of the grid search of the parameters and the model accuracy

Delegation Rule	Tally	k	α	Accuracy
Seniority	Global	≈ 2400	≈ 0.4	
	Local	15	0.303	0.832
Edit Count	Global	≈ 2300	≈ 0.3	
	Local	15	0.346	0.826
Rank	Global	≈ 2000	≈ 0.43	
	Local	17	0.289	0.829
Reveresed Rank	Global	≈ 1900	≈ 0.35	
	Local	16	0.298	0.826

We see that all the LOCAL VISCOUS models have performance greater than the baseline of 82% with the value of k close to 15 most influential users. The value of α indicates that the votes in the network are sufficiently viscous in nature‘.

8 Conclusions

In this paper we explore the election process for administrator in Wikipedia. We consider the problem of election result prediction using a subset of votes cast. We proposed a novel approach to using the theory of viscous democracy to obtain a core set of influential users and predict elections by tallying their votes. This model allows us to understand the nature of election in Wikipedia and the degree of transitive trust present withing the network of users. The experimental results show that our model is able to match and also outperform the baseline while only using a subset of the voters.

The small values of the delegation parameter show that the transitive trust in delegating votes is weak and that models with viscous votes perform better. We also see that delegation on the basis of seniority achieves the best performance and indicates that older Wikipedia members wield some influence in administrator elections.

In the future, we would like to create a method to infer the best delegation network and in turn the most optimal delegation rule. Another interesting approach would be to find a model that can explain the behaviour of a voter in a given election that provides a micro viewpoint of the election. This, when combined with the macro viewpoint provided by the viscous democracy model could provide a comprehensive understanding of election dynamics in Wikipedia.

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] Mangesh Gupte, Pravin Shankar, Jing Li, Shanmugauelayut Muthukrishnan, and Liviu Iftode. Finding hierarchy in directed online social networks. In *Proceedings of the 20th international conference on World wide web*, pages 557–566, 2011.

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

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

[7] 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.

[8] 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.

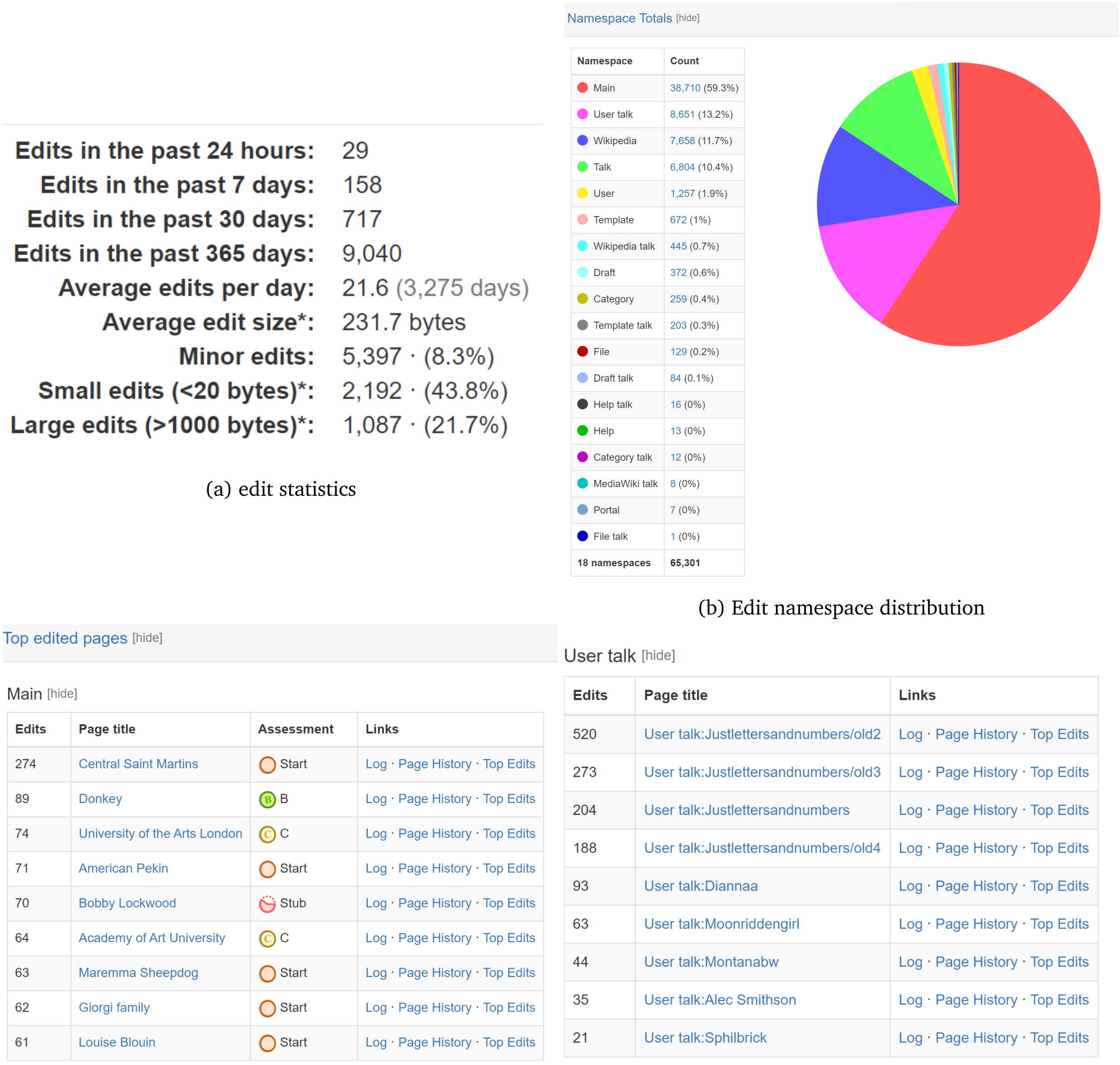
[9] Jure Leskovec, Daniel Huttenlocher, and Jon Kleinberg. Signed networks in social media. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, page 1361–1370, New York, NY, USA, 2010. Association for Computing Machinery.

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

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

[12] Nikolaj Tatti. Tiers for peers: a practical algorithm for discovering hierarchy in weighted networks. *Data Mining and Knowledge Discovery*, 31(3):702–738, December 2016.

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

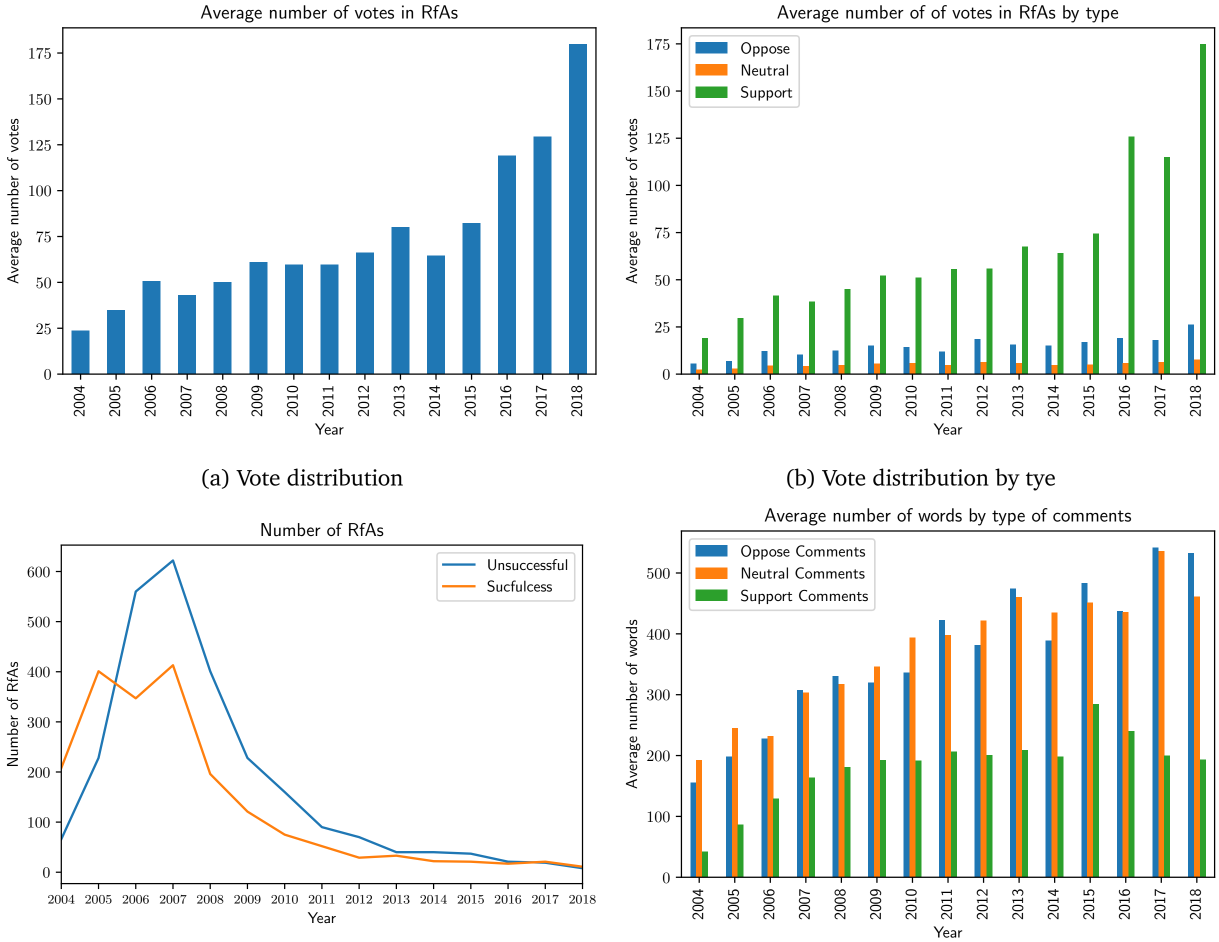
(c) Top edited pages

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 12: Edit summary of a Wikipedia user



(a) Vote distribution

(b) Vote distribution by tye

(c) Number of RfA by year

(d) Comments distribution

Figure 13: Election Statistics