A Theoretical Analysis of Crowdsourced Content Curation

Georgios Askalidis
Northwestern University
Evanston, IL, USA
gask@u.northwestern.edu

Greg Stoddard
Northwestern University
Evanston, IL, USA
gregs@u.northwestern.edu

ABSTRACT

To deal with the huge amount of potentially interesting content on the web today, users seek the help of *curators* to recommend which content to consume. The two most common forms of curation are expert-based (the editor of a newspaper decides which articles to place on the front page), and algorithmic-based (a search algorithm determines the ranking of websites for a given query). In recent years, content aggregators which use explicit vote-based feedback to curate content for future users have grown exponentially in popularity. The goal of this paper is to provide a descriptive analysis of these *crowdsourced curation mechanisms.*

In particular, we study crowd-curation mechanisms that rank articles according to a score which is a function of user-feedback. We precisely quantify the dynamics of which articles become popular in such these systems. While crowd-curation can be relatively effective for cardinal objectives like discovering and promoting content of high quality, they do not perform well for ordinal objectives such as finding the best articles. Our analysis suggests that user preferences and behavior are a far greater determinant of curation quality than the actual details of the curation mechanism. Finally, we show that certain shifts in user voting behavior can have positive impacts on these systems, suggesting that active moderation of user behavior is important for high quality curation in crowd-sourced systems.

1. INTRODUCTION

Today, the internet is flooded with vast quantities of content, both professionally created and user-generated. While the amount of potentially relevant content constantly grows, the time that people are able to devote towards consuming content remains the same. *Curation* is the layer that sits in between the universe of content and the limited attention of users. Informally, the goal of curation is to collect content, assess the quality of each item, and promote a relatively small set of content that the user should focus his attention on. Traditionally, curation was performed by experts, such as a newspaper's front page editor, while today the most common form of curation is algorithmic, ranging from ranking relevant webpages for a search query to produce recommendation systems on large e-commerce sites. In recent years, a new form of curation - crowdsourced curation - has risen in popularity due to the success of content aggregators such as Reddit or Digg.

The motivation behind crowdsourced curation is quite simple: solicit the opinions of users who are already reading articles and use that feedback to help curate content for future users. One common method of crowdsourced curation mechanisms is to rank each article according to the number of upvotes (approvals) and downvotes (disapporovals) that each article has received. Such a simple approach can actually circumvent some drawbacks of expert-based and algorithmic curation. Experts provide high quality curation but it can be expensive to pay a full-time editor and their attention is generally not scalable to the rising volume of content. Algorithmic curation can provide high quality curation and scales well but implementing good algorithms requires some degree of technological sophistication, even just to implement a machine learning algorithm from a standard library. Furthermore, high quality algorithmic curation is generally not flexible enough to handle changes in content type, e.g. from articles to videos, or shifts in user preferences. By contrast, crowdsourced curation is cheap, can be implemented by most web developers, and is flexible but it is not clear if it can provide high quality curation. While previous work addressed specific aspects of crowdsourced curation, there is not a detailed understanding of the exact dynamics and properties of such crowd curation mechanisms. Our objective in this paper is to provide a theoretical, descriptive analysis of these systems.

In recent years, the popularity of link-aggregating websites which employ these crowd curation mechanisms have greatly risen in popularity. Today, the website for the New York Times receives 15 million page views per day, Google News receives 11 million page views per day and Reddit receives 8.5 million page views per day[1]. Before Reddit reached this sort of popularity, another crowd curator called Digg received over 238 million unique visitors per year. As the popularity of such curation websites starts to rival that of traditional curators such as the New York Times, we believe it is important to understand the dynamics in these mechanisms that determine what content becomes popular.

[1] According to Alexa estimates taken during October 2012

In this work, we study a class of crowdsourced curation mechanisms which closely model the design and dynamics of the crowd curation mechanisms employed by the most popular link-aggregating websites. These websites have two defining characteristics. First, articles are ranked according to a score based on the number of upvotes (approval votes) and downvotes (disapproval votes) received. Second, these sites use a very specific website design which we call the front page model. When a user visits the site, they are shown the top k articles in the ranking. To view the $k+1^{\text{st}}$ ranked article, the user must click on a link to visit the next page. In effect, this focuses almost all attention on the top k articles while the articles lower in the ranking receive almost zero attention (see Section 3 for more details on this observation). Member of the popular press as well as researchers have observed that such a design introduces a "rich get richer" effect because articles which are in the top k are likely to receive more positive feedback and thus more likely to appear on the front page in the future. Our analysis quantifies the extent to which this effect can harm the mechanism's ability to curate effectively.

We emphasize that our goal is a descriptive analysis, in the sense that we are attempting to answer the question "what is currently happening in crowdsourced curation mechanisms", as opposed to the prescriptive question of what "should" happen in these systems. We adopt this focus in an effort to bridge the gap between theory and the reality of existing systems. There are many scenarios where the theoretically optimal solution is not used in practice because that solution may violate practical design constraints or not take into account a few important practical objectives. In the realm of crowd curation systems, a prime example of the gap between theory and practical systems is the role of randomization. These systems present an obvious explore vs exploit problem, should we explore more articles to find out their quality or just exploit the best ones that we have at the moment, for which randomization is necessary for a good algorithm. Indeed, in 2005 Pandey et al, [10], demonstrated that a small amount of randomness can yield large gains in quality. Despite this theoretical understanding, popular user-generated content and link aggregating sites, such as Reddit, Hackernews, Slashdot and Stackoverflow, do not fundamentally incorporate randomness into their rankings algorithms. In light of this apparent gap, we feel asking the descriptive question is interesting and well-motivated.

Our Contributions We present a model which captures fundamental features of the most popular crowd-curation websites. Our model has three essential components: article quality, user voting behavior, and user attention. Our models for article quality and user behavior are general and based on similar models from previous work in this area, [7]. Our model for user attention is specifically tailored to model the allocation of attention induced by the popular Front Page design found in most content aggregators. This specific front page model, which comes at the loss of some generality, allows us to make more relevant theoretical predictions about the real-world curation mechanisms that we care about.

We analyze the evolution of the article ranking through a random walk and use a combination of markov chain techniques and Bayesian statistics to precisely compute distributions over articles which appear in the top k. Using this distribution, we compute the performance of a curation mechanism with respect to two metrics, *curation quality* which is the average fraction of the population that an article in the top k satisfies, and *discovery efficiency* - the probability that the best article is in the top k. For example, when article qualities are drawn uniformly between $[0, 1]$, a random article in the top k will satisfy 81% percent of users in expectation but there's a relatively small chance of finding the best article; the top k is only 3 times more likely to contain the best article than a random set of k articles is. We use this distributional characterization to compare performance of curation mechanisms on two different distributions F and G when they satisfy a stochastic dominance relation. We then provide theoretical evidence in support of active moderation of user behavior. We show that certain changes in user behavior, such as more-aggressive downvoting or submitting more diverse content, induce dominance shifts which cause an improvement in curation quality and discovery efficiency. Finally, we find that, under the top k site design, all curation mechanisms produce the same distribution of articles in the top k, suggesting that the most important design choice for any curation mechanism is how to allocate user attention.

2. RELATED WORK

In the realm of user-generated content and ranking mechanisms, there are two main bodies of work. The larger body of work is concerned with analyzing and designing ranking mechanisms that improve upon the basic pitfalls that simple voting mechanisms encounter, [14], [17], [16], [1]. Using a simulation-based analysis, Cho et al, [2], demonstrate that the top-k ranking principle leads to the "rich-get-richer" problem. One simple proposal to fight this problem is the use of randomization, [15], [10], but as we argued in the introduction, most curation mechanisms do not use randomization. Das Sarma et al [3], show impossibility results for these curation mechanisms but with respect to a very strong ordinal metric that might be appropriate for smaller environments but are not particularly relevant to large link aggregators. Hogg and Lerman, [8], use Markov chains to model the submission, rating, and sharing of content on social curation systems.

The more recent body of work examines the incentives in user-generated content systems. There has been a large work in modeling the ranking mechanisms in a game theoretic manner when users are rewarded by the attention their contribution receives but incur increasing costs to produce higher quality content. In a series of papers, [7], [4], [5], Ghosh et al showed that ranking mechanisms need to be designed carefully in how they distribute attention to each submission in order to incentivize high quality submissions in equilibrium. In a more recent paper, Ghosh and Hummel, [6], examined similar incentive issues in UGC systems that use machine learning approaches to discover good content. Our work does not model any incentive issues and is thus more similar in spirit to the first body of work.

3. MODEL

Before we describe each component of the model formally, we give a brief overview of the process.

1. Content is submitted to the curation website at the beginning of the day.

2. A user visits the site and is shown the k articles with the highest score, known as the top k. Scores are based on the total number of upvotes and downvotes that each article has accumulated before this user's visit.

3. After reading each article in the top k, the user upvotes or downvotes that article.

4. The website recomputes scores for each article based on the new totals of upvotes and downvotes.

Our model is characterized by three important entities: the articles, the users, and the ranking rule used by the mechanism.

The Articles At time $t = 0$, before any users visit the site, n articles are submitted to the site. Each article i has an unknown quality q_i, drawn identically and independently from a distribution F over $[0, 1]$; we will refer to F as the *quality distribution.* Similar to [7] and others, we define an article's quality as the fraction of the user population that would upvote this article if they read it. By this definition, the article distribution is endogenously determined by both the set of articles submitted to the curation mechanism and the preferences of the user community that inhabit the site. This is a practically relevant point for later results in the paper which show that changing the quality distribution in particular ways can positively impact the ability of the mechanism to curate. These particular changes to the quality distribution can be implemented by inducing changes in user preferences or behavior or shifting the sort of content which is submitted to the site.

User Attention At each time step $t = 1, 2...T$, a single user visits the site. All n articles are displayed according to the current rankings (determined by article score) but the user only views the "front page", the top k articles according to the ranking. That is, we assume that users view and vote on each article in the top k but then leave the site after that point. This modeling choice is motivated by the prevalence of the top k design of many webpages, particularly amongst the sort of crowd curation sites that we seek to understand. It is quite clear that such a heavy concentration of user attention on the top articles creates a "rich get richer" effect for the articles which manage to reach the front page. Our goal is to further this observation and exactly quantify the effect on curation quality.

The mere use of a front page design does not necessarily imply that there's a large concentration of user attention on the top articles. However, there's a myriad of empirical and anecdotal evidence, collected from popular websites, that suggest that user attention tends to be heavily focused on the front page. Figures 1a and 1b show the result of a statistical analysis of the set of links submitted to Reddit over a three day period, [9]. Figure 1a is a histogram of the distribution of votes across links. This histogram shows that not all attention is focused on the most popular but that an exponentially large fraction of the attention is devoted to the most popular articles (the buckets on the x-axis are exponentially sized). Similarly, figure 1b shows the evolution of score of some random articles. Most articles are created and have a very low score through the duration of the life of the link, except for the article which reaches the top k, after which its score grows at a very rapid rate. To a first-order approximation, there is very little difference in the dynamics induced by assuming that *zero* attention moves beyond the top k and assuming that an exponentially small amount goes beyond the top k. For certain types of curation mechanisms, including the one used by Reddit and Hackernews, we can prove that performance is almost exactly the same under the assumption of no attention past the top k or a very small amount of attention past the top k. See appendix B for more detail. We can also easily extend our model to the case where each article gets a limited amount of feedback as it is submitted to the system. The extension does not fundamentally change the dynamics that we study. See appendix A for more detail.

We note that we can extend our model to the case where users don't always vote upon every article in the top k, for example, if the fraction of users who read slot i was a decreasing function of i. So long as no users read past the the kth article, then our results hold.

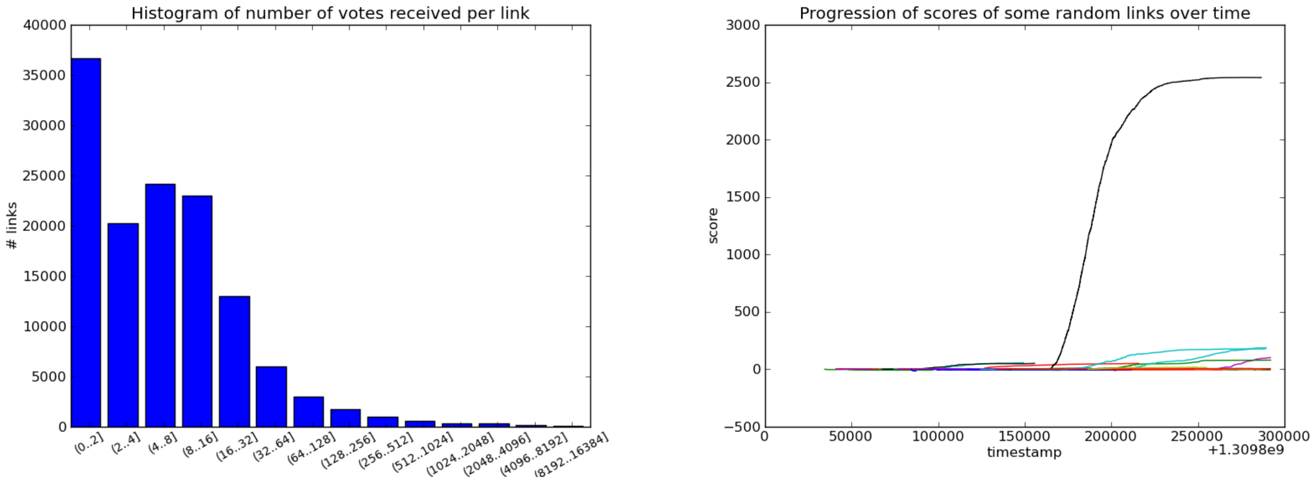
User Voting Given that article quality is defined by the fraction of users who would upvote that article, the user voting model is fairly simple. After a user reads an article with quality q_j, that user upvotes j with probability q_j and downvotes with probability $1 - q_j$. We assume that how a user at time t votes on article j is conditionally independent (conditioned upon the quality q_j) of all votes on article j from previous users and independent of how user t voted on all other articles $i \neq j$.

This model describes aggregate voting behavior, it doesn't directly describe how each specific user acts. That is, we don't assume that users necessarily vote according to this stochastic process but we assume that voting behavior can be explained in this manner in expectation. This allows our analysis to extend to any specific voting behavior which satisfies the above independence assumptions.

The Curation Mechanism In this paper, we study curation mechanisms which operate by assigning a score to each article, based on the accumulated upvotes and downvotes for each article, and rank articles according to the assigned score[2]. Again, this particular modeling choice reflects the sorts of mechanisms that are found on popular curation sites. For a given curation mechanism M, let $M(u, d)$ be the score M assigns to an article with u upvotes and d downvotes. Two common example of these curation mechanisms are $M(u, d) = u - d$, the difference rule, and $M(u, d) = \frac{u+1}{u+d+2}$, the fraction rule (the constants are there to handle the case when an article has 0 feedback). Formally, we make the following assumption on M:

1. *Sort by Score:* Let (u_i, d_i) and (u_j, d_j) denote the upvotes and downvotes for articles i and j. If $M(u_i, d_i) > M(u_j, d_j)$, then article i is ranked above article j.

[2] This is a small class of curation mechanisms. Other such approaches could operate via some multi-armed bandit process or displaying some articles according to newness,etc.



(a) Vote Histogram

(b) Score vs Time

Figure 1: Two plots from the Reddit blog showing the results of a statistical analysis of Reddit over a three day period . Figure 1a shows a histogram of the number of votes per link. The buckets on the x-axis are exponentially sized. This histogram reveals that an exponentially number of votes are focused on a very small set of articles Figure 1b shows the evolution of scores for random articles posted on Reddit. The steep line demonstrates the rapid increase in score when the associated article was promoted to the front page. Source: [9]

2. *Default Score:* For any $x > y$, $M(x, y) \geq M(0, 0) \geq M(y, x)$. That is, any article which has more upvotes than downvotes should be ranked above an article with no votes and any article with less upvotes than downvotes should be ranked below an article with no votes.

3. *Tie Breaking:* Ties are broken in favor of whichever article was ranked higher in the last round. That is, we break a tie in the favor of article i over article j in round t if i was ranked higher than j in round $t-1$. At the time $t = 0$, we break ties randomly.

Assumption 1 is our strongest assumption in the sense that it precludes the class of curation mechanisms that explicitly operate according to some explore-exploit principle. Despite the benefits that randomization or other sorts of exploration can have, many popular curation websites use very simple rules that adhere to the above assumptions. For example, analysis of the source code of Reddit and Hackernews shows that both use the difference scoring rule to rank articles ([13], [12]). Studying the class of mechanisms which adhere to these assumptions allows us to get a better picture of the dynamics of these popular curation mechanisms.

Performance Metrics We will measure the performance of a curation mechanism by two metrics, *curation quality* and *discovery efficiency*, which are metrics on the set of articles displayed in the top k when the system reaches its steady state. As we will show in the next section, these systems reach a point where the set of articles which appear in the top k will remain in the top k for all remaining time steps. We will denote the steady state set of top k articles by $\overline{topk}$.

1. *Curation quality*, denoted $Q_F(M)$, is the average quality of articles displayed in the final top k when curation mechanism M is run on articles with qualities drawn from quality distribution F. Formally,

$$Q_F(M) = \frac{1}{k} E[\sum_{i \in \overline{topk}} q_i]$$

Where the expectation is taken over any randomness in the draws from the quality distribution and the randomness in user voting.

2. *Discovery efficiency*, denoted $D_F(M)$ is the probability that the best article, the one with the highest quality, appears in the final top k when mechanism M is run on articles with qualities drawn from quality distribution F. Formally

$$D_F(M) = P(q^* \in \overline{topk})$$

Where q^* represents the article with the highest quality q and the probability is taken over any randomness in the draws from the quality distribution and the randomness in user voting.

We feel that studying both of these metrics add to our understanding of curation mechanisms. Curation quality essentially measures the average user satisfaction with the set of articles that appear on the front page. Alternatively, we can view this as the extent to which the mechanism is responsive to the user population of the sites. This is an important metric to keep in mind for the continued growth and popularity of a given curation mechanism. If the curation quality is too low, it is likely that many users will stop coming to the site. Indeed, many tech writers partially attribute the rapid decline Digg's popularity to a change in their mechanism which allowed news publishers to promote their content to the front page over the content which was voted up by the community. Discovery efficiency is arguably better suited to the general notion of curation. Ideally, users don't want good things, they want to be shown the very best things. One could also imagine employing these curation mechanisms to shift through large quantities of content in order to find the best articles and to archive them for use in the future. The discovery efficiency of a curation mechanism gives a sense of how well-suited it is to these sorts of tasks.

Additional Assumptions The final assumption that we make in this paper is that the number of articles, n, is much larger than k, the number of articles displayed on the front page. In particular, we assume the following:

$$\frac{k}{n} \leq \frac{1}{2}\left(1 - F(\frac{2}{3})\right)$$

Since F is the quality distribution, $1 - F(\frac{2}{3})$ is the fraction of articles which are liked by at least two thirds of the population. For example, if we assume that the front page shows 10% of all articles ($\frac{k}{n} = \frac{1}{10}$) then we assume that at least 20% of articles are approved of by at least two thirds of the population[3]. In practice, n is generally much larger than k and this is assumption is benign.

4. A COMMON MECHANISM

In this section, we analyze the simple yet common curation mechanism of ranking by difference of upvotes and downvotes, $M(u_i, d_i) = u_i - d_i$. Our goal is to quantify the performance of the difference scoring with respect to the two metrics we introduced above: the expected quality of an article in the final top k and the probability that the highest quality article appears in the final top k. In order to do this, we need to understand a few properties of these systems such as the probability that an article with quality q will remain in the top k if it ever gets to the front page, the expected number of articles that receive any votes, the probability that a random article will ever be promoted to the front page, etc. The main idea is to frame this process as a random walk over the scores of the articles and analyze hitting times for scores that push articles out of the top k. We then apply Bayes rule to derive the conditional distribution of qualities of articles that appear in the final top k. Using this conditional PDF, we can quantify the performance of the difference mechanism on a general distribution F. Although we study a particular simple mechanism in this section, the observations that drive this analysis hold for any curation mechanisms that satisfy our assumptions in Section 3

We begin by noting that since the score for each article i is $s_i = u_i - d_i$, we can model the movement of an article's score as a random walk with probability q_i of moving to $s_i + 1$ and probability $1 - q_i$ of moving to $s_i - 1$ whenever article i receives a vote. However, only the top k articles with highest score actually receive a vote, so the score of the remaining articles will remain constant. At first glance, it might seem that we need to consider the joint random walk between all n articles but it turns out that we can actually analyze each random walk independently. Since we are only concerned with the set of articles in the final top k, the relative positions of any two articles within the top k do not matter. On the other hand, the relative scores between an article in the top k and an article outside the top k do matter. An article in the top k at time t will only leave the top k at time $t + 1$ if its score falls below the highest score of any article outside the top k. As we show in the next lemma, the maximum score of any article outside the top k will always be the *default score*, the score that is assigned to articles which have 0 upvotes and 0 downvotes. In this case, the default score is 0.

LEMMA 1. *Let π^t denote the ranking of articles at time t and let π^t_i denote the article in position i in the ranking. Let $M(0, 0) = x_0$ denote the default score. Then $\pi^t_{k+1} = x_0$ for all t.*

PROOF. Assume for now that v_F, the total number of articles that are viewed during this process is less than n. We show this is true in the proof of lemma 3. When an article is removed from the top k, it must have accumulated one more downvote than upvote and is thus placed at the bottom of the ranking (a consequence of assumptions 2 and 3), below all articles with default score. Since there are more articles n than articles viewed, there will always exist an article with no votes, and hence the default score. Thus the score of the k + 1st article will always be the default score. □

This suggests a relatively simple dynamic. An article i in the top k will only fall out of the top k when $s_i = -1$. When this happens, article i will be placed at the bottom of the ranking and a new article will be promoted into the top k. If the article never hits state -1, then it will remain in the top k for the duration of time, hence becoming part of the final top k articles.

Observation 1. *An article i which enters the top k will appear in the final top k articles if and only if the associated random walk never hits state -1.*

A direct way to calculate this value is to figure out the probability of a simple random walk eventually hitting state -1, and then subtracting that value from 1. In the language of random walks, this can be computed via *hitting time* analysis. The hitting time of a state s, denoted H^s_i, is defined as the first time the random walk enters state s if it starts from state i. Then the probability that a random walk never enters a state is exactly the probability that state has an infinite hitting time. For simple random walks, hitting times have nice closed form solutions.

Let $p(\overline{topk}|q)$ denote the probability that an article remains in the top k permanently, given that its quality is q. Then hitting time analysis yields the following

$$p(\overline{topk}|q) = \begin{cases} 2 - \frac{1}{q} & q > \frac{1}{2} \\ 0 & q \leq \frac{1}{2} \end{cases} \tag{1}$$

So if an article with quality $q > \frac{1}{2}$ enters the top k, the

[3] This condition is sufficient for our results to hold but it is not necessary

probability that it will remain in the top K after all users have visited is $2 - \frac{1}{q}$. Articles with quality below $\frac{1}{2}$ have probability zero of remaining in the top k.

When one article is kicked out of the top k, a brand new article is promoted in it's place. The quality of that article is distributed according to f, the PDF of the quality distribution (since the new article has no votes, we don't need to do any conditioning). Using this fact and Equation 1, we can apply Bayes rule to actually compute the distribution of qualities of articles in the final top k.

Let $f(q|\overline{topk})$ be the PDF of the quality of an article that is in the final top K. Then using Bayes rule

$$f(q|\overline{topk}) = \frac{p(\overline{topk}|q)f(q)}{p(\overline{topk})}$$

The quantity $p(\overline{topk}|q)$ is exactly what we computed in equation 1, and we can compute $p(\overline{topk})$ by marginalizing over quality. This yields our first theorem.

THEOREM 1. *Assume that k and n satisfy the sufficient articles assumption. Then for any curation mechanism M that satisfies the assumption in Section 3, the conditional distribution of article qualities in the final top k is given by*

$$f(q|\overline{topk}) = \frac{(2 - \frac{1}{q})f(q)}{\int_{.5}^{1}(2 - \frac{1}{x})f(x)\delta x} \tag{2}$$

PROOF. Standard application of Bayes rule. The limits of integration in the denominator are take from $\frac{1}{2}$ to 1 since equation 1 shows that articles with quality below $\frac{1}{2}$ have 0 probability of remaining in the top k. □

4.1 Curation Quality

Given the distribution of top k article qualities, it is now straightforward to calculate the expected quality of an article that appears in the final top k.

LEMMA 2. *The curation quality for a mechanism M on distribution F is given by*

$$Q_F(M) = \frac{1}{\int_{.5}^{1}(2 - \frac{1}{x})f(x)\delta x}\int_{.5}^{1} q(2 - \frac{1}{q})f(q)\delta q$$

PROOF. Follows from the definition of expectation and Theorem 1. □

For example, when article qualities are distributed uniformly over $[0, 1]$, i.e. $f(q) = 1$, we get

$$E[q|\text{top } k] = \frac{1}{\int_{.5}^{1}(2 - \frac{1}{x})\delta x}\int_{.5}^{1} q(2 - \frac{1}{q})\delta q \ \approx .814$$

4.2 Discovery efficiency

Recall that the discovery efficiency of a mechanism M on distribution F is the probability that the best article, the one with the highest quality, appears in the final top k. When F has full support over $[0, 1]$ and n is sufficiently large, the quality of the best article is essentially 1. For the sake of simplicity, we'll assume this is the case but it is straightforward to extend this analysis when the expected maximum quality is less than 1.

LEMMA 3. *The discovery efficiency of a mechanism M with quality distribution F, $D_F(M)$, is the probability that the best article appears in the final top k. This is given by*

$$D_F(M) = \frac{k}{n} \times \frac{1}{\int_{.5}^{1}(2 - \frac{1}{q})f(q)\delta q}$$

PROOF. If an article has quality 1, then it will appear in the final top k if it is ever promoted to the top k. Thus we need to figure out the probability that the best article is promoted to the top k at some point. Let V_F denote the expected number of articles that are promoted to the top k at some point in this process (even if the article is later voted out). Articles have no feedback before they enter into the top k, so all qualities are promoted into the top k with equal probability. Thus if V_F articles are promoted in total, and there are n articles, the probability that any particular article is promoted is equal to $\frac{V_F}{n}$.

It remains to calculate V_F. For each randomly promoted article, there's a $p(\overline{topk})$ probability that it remains in the top K permanently. If we had a single slot in the top k, we would need to view in expectation $\frac{1}{p(\overline{topk})}$ articles in order to find one that would remain in the top k permanently. Since we have k slots, $V_F = k \times \frac{1}{p(\overline{topk})}$. Plugging in the equation for $p(\overline{topk})$ yields the result.

We finally show that $V_F < n$ under the assumption that $\frac{k}{n} < \frac{1}{2}(1 - F(\frac{2}{3}))$ as we assumed in Section 3. We have that

$$V_F = \frac{k}{\int_{.5}^{1}(2 - \frac{1}{q})f(q)} \leq \frac{1}{\frac{1}{2}(1 - F(\frac{2}{3}))} \leq n$$

Since $2 - \frac{1}{q}$ at $q = \frac{2}{3}$ is equal to $\frac{1}{2}$, so the integral over .5 to 1 is at least $\frac{1}{2}$ times $1 - F(\frac{2}{3})$. The last inequality follows from the assumption. □

For example, if the quality distribution is the uniform distribution, the discovery efficiency is $\frac{k}{n} \times 3.25$.

4.3 Discussion

We observe that although we did the above analysis mainly in the context of the the curation mechanism being $M(u, d) = u - d$, the proof of Theorem 1 only relied upon the hitting times of scores that cause articles to be pushed out of the top k. Lemma 1 showed that for all mechanisms which satisfy our assumptions in Section 3, score of the $k + 1$st article is always the default score for that mechanism. Finally, assumption 2 implies that for any curation mechanism which satisfies these assumptions, the score of an article only falls below the default score only when that article accumulates one more downvote than upvote. Thus so long as this property is satisfied by two different curation mechanisms, they will have the exact same dynamics in the top k model. For example, another common mechanism is the proportional rule $\frac{u+1}{u+d+2}$. The default score is $\frac{1}{2}$ and it is easy to verify that an article receives a score below $\frac{1}{2}$ if and only if it has more downvotes than upvotes.

Lemmas 2 and 3 demonstrate our second claim: these curation mechanisms have the potential to deliver a high curation quality, but do very poor in terms of finding the best articles. This is evident based on the fact that curation quality has no dependence on n or k, it only depends on the quality distribution F. On the other hand, discovery efficiency decreases quickly as the number of articles submitted to the mechanism grows. For any quality distribution F and k, the discovery efficiency eventually goes to 0 as n increases.

Despite the fact that curation quality scales well, there is still a "rich-get-richer" effect which hinders the ability to provide a higher curation quality. The function $p(\overline{topk}|q)$ is an increasing function of q, so higher quality articles have a higher chance of remaining in the top k, but it is also a concave function, so the marginal advantage of a higher quality diminishes as q increases. Any article which has a quality greater than $\frac{2}{3}$ has at least a fifty percent chance of remaining in the top k after it has been promoted there. If the distribution happens to have lots of probability mass around $q = \frac{2}{3}$, then there is an effect where the "good" articles will crowd out the "great" articles. We discuss this effect more in the next section.

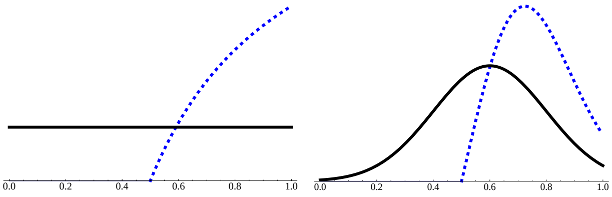
5. QUALITY DISTRIBUTION

In this section, we examine the role of the quality distribution in determining curation quality and discovery efficiency. Figures 2a and 2b each show both the PDF for the original quality distribution, uniform and normal respectively, and the PDF for the top k quality distribution. As these graphs show, the shape of the original distribution has a large influence on the resulting distributions of quality within the top k. Our goal in this section is to establish some relationship between two distributions F and G where we can conclude that discovery efficiency or curation quality improves if we changed the quality distribution from F to G.

These comparative static results are practically relevant. As we noted in Section 3, the quality distribution is determined by both the set of articles which are being submitted to the system and the preferences and behavior of the user population. Thus a change in user behavior or in the set of article being submitted can induce a shift in the quality distribution. We show that there are natural changes to behavior and article submission that cause an improvement to curation quality and discovery efficiency because those changes induced an appropriate shift in the quality distribution. For user behavior and preferences, we show that a little more downvoting can improve discovery efficiency. We also find that diversity in article submissions can improve curation quality and discovery efficiency. These results suggest that active moderation of user behavior can positively affect curation mechanisms.

User Preferences and Behavior

Loosely speaking, more downvotes in curation mechanisms can help curation quality and discovery efficiency. There's a simple intuition behind this reasoning: downvotes promote more exploration while upvotes serve to preserve the status quo. We consider a shift in the quality distribution to one where there are comparatively less good quality articles. This change can be brought about by active moderation efforts to enforce community guidelines. For example, Reddit has a guideline to upvote or downvote based on whether an article is "well-written and interesting" and not to vote based on the opinion that the article expresses. This guideline is often ignored, resulting in poorly-written articles that express popular opinions receiving a large quantity of upvotes. If moderators were able to strictly enforce this guideline[4], there would effectively be more downvotes in the system.



(a) Uniform (b) Normal

Figure 2: Both plots show the PDF of the quality distribution $f(q)$ as the thick line and the PDF of the top k quality distribution, $f(q|\overline{topk})$ as the dashed line. Figure 2a shows the uniform distribution and figure 2b shows the normal distribution with mean .6 and variance .2. The input distribution has a strong effect on the shape of $f(q|\overline{topk})$

We say that a distribution F *first order stochastically dominates* a distribution G if F places more probability mass on higher quality content than G does. Formally,

Definition [FOSD] 1. A distribution F is said to First Order Stochastically Dominate (FOSD) a distribution G if $F(x) \leq G(x)$ for all $x \in [0,1]$ with at least one point where the inequality is strict.

Under this definition, we have the following interpretation: fix some quality level $q \in [0,1]$, and take a random draw from q_F from F and random draw q_G from G. Then if F FOSD G, the probability that $q_F \geq q$ is greater than the probability that $q_G \geq q$. Under this condition, we prove that the discovery efficiency of F, $D_F(M)$ is *lower* than $D_G(M)$, the discovery efficiency on G.

LEMMA 4. *Let F first order stochastically dominate G. Then for a curation mechanism M, $D_F(M) < D_G(M)$, i.e.*

$$\frac{k}{n} \times \frac{1}{\int_{.5}^{1}(2-\frac{1}{q})f(q)\delta q} < \frac{k}{n} \times \frac{1}{\int_{.5}^{1}(2-\frac{1}{q})g(q)\delta q}$$

PROOF. We can define a new function $h(q)$ where $h(q) = 0$ if $q \leq \frac{1}{2}$ and $h(q) = 2 - \frac{1}{q}$ if $q > \frac{1}{2}$. Then $\int_{.5}^{1}(2-\frac{1}{q})f(q)$ is equal to $E_F[h(q)]$. An equivalent definition of FOSD is that $E_F[h(q)] > E_G[h(q)]$ for all increasing functions ([11]), which $h()$ is. The result follows. □

This result is actually fairly intuitive; if there is a greater frequency of higher quality articles (implied by F FOSD G),

[4] It is not exactly clear on how to enforce such a guideline, especially when "interesting and well-written" is a subjective measure, but we ignore this issue for now.

then a random article is more likely to remain in the top k once it enters, which in turn reduces the total number of articles explored by the mechanism.

Next we show that if all users raise their standards by a little, just by sometimes downvoting an article that they would have normally upvoted, this can actually increase the average quality of articles that appear in the final top k. Specifically, we define a modification to user behavior called ϵ-random downvoting: When a user decides to upvote an article, she will instead downvote it with probability ϵ. We show that for any distribution, there exists an $\epsilon > 0$ where ϵ-random downvoting raises the curation quality of the mechanism. See Appendix C for the proof.

LEMMA 5. *Let $Q_F(M, \epsilon)$ denote the curation quality of mechanism M run on input distribution F when all users follow ϵ-random downvoting. Then for all distributions F, there exists an $\epsilon > 0$ such that $Q_F(M, \epsilon) > Q_F(M)$.*

This result shows that more downvoting can actually raise curation quality, effectively making all users in the population happier with the set of articles displayed in the top k. Raising ϵ decreases the probability than any article remains in the top k but the decrease in probability is more severe for lower quality articles, resulting in a rise in curation quality. While it seems odd to ask users to randomly downvote articles, the same effect is achieved by encouraging users to downvote a bit more aggressively. For example, if ϵ is 10% and the input distribution is the uniform distribution, then ϵ-random downvoting can raise the curation quality from .81 to .84. We note that raising ϵ to high values (like .5) actually increases curation quality according to our metrics because all articles, except those of quality near 1, will be kicked out of the top k. While this does increase curation quality, it greatly increases the time it takes to reach a steady state and causes a low curation quality at intermediate points of this process. In this work, we do not address the tradeoff between final curation quality and intermediate curation quality but it seems that curation mechanisms would not want to sacrifice too much intermediate quality for final curation quality.

Article Submissions

When user preferences and behaviors are fixed, the quality distribution can still be altered by changing the set of articles which are being submitted to the curation mechanisms. We show below that if two distribution F and G have the same mean but G is "riskier" than F, then curation quality and discovery efficiency of the mechanism is better on G than on F.

What is the interpretation of a risker distribution in this context? Consider one distribution F where users mostly submitted pictures of a very similar nature, perhaps all pictures of cats, and another distribution G where the content was a bit more diversified, so less pictures of cats but more pictures of different subjects. Under the distribution F, the quality distribution should be concentrated around the quality of a cat picture, where the quality distribution G would be more evenly spread out over all qualities. Then, so long as F and G have the same mean, the curation quality is better for G than for F. In this sense, diversity of articles is a desirable property for these systems.

What does it mean for one distribution to be riskier than another? For a simple example, imagine there was only one type of article in a distribution F, which always has quality .75. In the distribution G, there are two equally likely types of articles, one which has quality .6 and the other which was quality .9. F and G have the same expected value (.75) but there is more uncertainty over the quality of a random article from G. The concept of *second order stochastic domination* extends and formalizes this example to general distributions.

Definition [SOSD] 1. Let F and G have support over $[a, b]$. F second order stochastically dominates G (F SOSD G) if and only

$$\int_a^x G(q) - F(q)\ \delta q \geq 0$$

for all $x \in [a, b]$, with strict inequality at (at least) one point.

In this work, F and G are defined over $[0, 1]$, however all of our applications only care about F and G over the range $[\frac{1}{2}, 1]$, so we can truncate the distributions to fit in those ranges. It may be possible that F does not dominate G over $[0, 1]$ but does dominate it over $[\frac{1}{2}, 1]$ and for our purposes we could consider that F stochastically dominates G.

LEMMA 6. *For two distributions F and G, if F second order stochastically dominates G over $[\frac{1}{2}, 1]$ and $E_F[q|q \geq \frac{1}{2}] = E_G[q|q \geq \frac{1}{2}]$ then $Q_F(M) < Q_G(M)$.*

The above lemma says, in the context of a user-generated content website, is that it's better for curation quality if users submit a wide range of content instead of just following the popular trend of the week (as usually happens e.g. with internet memes on websites like Reddit). For example, a user could submit an article that will have quality .75 for sure or another article that has an equal chance of having quality .6 or quality .9. From the curation mechanism's perspective, the riskier article is a better choice since if the realized quality is .9, we have a very high quality article, but even in the case where the realized quality is .6, there is only a low chance that it will remain in the top k. By contrast, if the article with sure quality .75 is submitted, then it will be likely to stay in the top k, potentially crowding out higher quality articles.

If the goal of a submitter is to maximize the exposure of their article (such as the submitter model in [7]), they would prefer to submit the safe article of quality .75 in order to maxmize their chance that their article remains in the top k. Thus the goal of the user and the goal of the curation mechanism are not well-aligned. A deeper understanding of the incentive properties of these curation mechanisms is an interesting direction for future research.

6. CONCLUSIONS AND FUTURE WORK

In this paper we introduced a robust and simple model that accurately represents the curation mechanisms used by many popular link aggregators. With this model, we analyze the dynamics of these curation mechanisms and make precise predictions about the articles which appear in the top k articles. With this analysis, we showed that the "front page" website design is an extremely important design decision because many curation mechanisms have the same properties and dynamics under this design. Finally, we showed theoretical evidence in support of active moderation of user behavior.

The analysis in this work can serve as the basis for a number of interesting future research directions. In earlier sections, we showed theoretical evidence that incentivizing certain user behaviors will positively impact the curation quality of these mechanisms but finding the correct way to incentivize these behaviors is an open question. We believe this is an important question because there are websites that are currently attempting to incentivize positive user behavior but end up doing so at the expense of curation. For example, the comment system of Hackernews only allows users with a sufficiently high reputation score to downvote but it allows all users to upvote. This design creates a scenario with many more upvotes and downvotes, which degrades curation quality. Our analysis in this work provides a tool to examine these sorts of design decisions.

Another interesting direction is to examine the role of diverse preferences over articles. In this work we assumed a simple user preference model but it would be interesting to employ a more detailed user model and see how the diversity of preference types affects the distribution of articles in the top k. Is the distribution of articles in top k reflective of the distribution over preference types or is it the case that the largest preference type disproportionately determines the top k articles? An answer to this question would indicate whether or not these curation mechanisms are a good source for finding diverse sets of articles or if they only represent the opinions of the majority type.

7. REFERENCES

[1] C. Brandt, T. Joachims, Y. Yue, and J. Bank. Dynamic ranked retrieval. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 247–256. ACM, 2011.

[2] J. Cho and S. Roy. Impact of search engines on page popularity. In *Proceedings of the 13th international conference on World Wide Web*, pages 20–29. ACM, 2004.

[3] A. Das Sarma, A. Das Sarma, S. Gollapudi, and R. Panigrahy. Ranking mechanisms in twitter-like forums. In *Proceedings of the third ACM international conference on Web search and data mining*, pages 21–30. ACM, 2010.

[4] A. Ghosh and P. Hummel. A game-theoretic analysis of rank-order mechanisms for user-generated content. In *12th ACM Conference on Electronic Commerce (EC)*, 2011.

[5] A. Ghosh and P. Hummel. Implementing optimal outcomes in social computing: a game-theoretic approach. In *Proceedings of the 21st international conference on World Wide Web*, pages 539–548. ACM, 2012.

[6] A. Ghosh and P. Hummel. Learning and incentives in user-generated content: Multi-armed bandits with endogenous arms. 2012.

[7] A. Ghosh and P. McAfee. Incentivizing high-quality user-generated content. In *Proceedings of the 20th international conference on World wide web*, pages 137–146. ACM, 2011.

[8] T. Hogg and K. Lerman. Stochastic models of user-contributory web sites. In *Proc. Int. Conference on Weblogs and Social Media*, pages 95–97, 2009.

[9] David King. The tale of the life of a link on reddit. `http://blog.reddit.com/2011/07/nerd-talk-tale-of-life-of-link-on.html`, July 2011.

[10] S. Pandey, S. Roy, C. Olston, J. Cho, and S. Chakrabarti. Shuffling a stacked deck: The case for partially randomized ranking of search engine results. In *Proceedings of the 31st international conference on Very large data bases*, pages 781–792. VLDB Endowment, 2005.

[11] M. Rothschild and J.E. Stiglitz. Increasing risk: I. a definition. *Journal of Economic theory*, 2(3):225–243, 1970.

[12] Amir Salihefendic. How hacker news ranking algorithm work. `http://amix.dk/blog/post/19574`, October 2010.

[13] Amir Salihefendic. How reddit ranking algorithms work. `http://amix.dk/blog/post/19588`, November 2010.

[14] E.I. Sparling and S. Sen. Rating: how difficult is it? In *Proceedings of the fifth ACM conference on Recommender systems*, pages 149–156. ACM, 2011.

[15] M. Vojnovic, J. Cruise, D. Gunawardena, and P. Marbach. Ranking and suggesting popular items. *Knowledge and Data Engineering, IEEE Transactions on*, 21(8):1133–1146, 2009.

[16] F. Wu and B. Huberman. Popularity, novelty and attention. *Available at SSRN 1087132*, 2008.

[17] D. Zhang, R. Mao, H. Li, and J. Mao. How to count thumb-ups and thumb-downs: user-rating based ranking of items from an axiomatic perspective. *Advances in Information Retrieval Theory*, pages 238–249, 2011.

APPENDIX

A. INITIAL FEEDBACK

Many crowdcuration websites use a "New" page, where users can view and vote on submitted articles that have a very small amount of votes (typically 0), to gather initial feedback on recently submitted articles. Such a feature primarily serves the purpose of ensuring that low quality content does not rise to the front page but it can also help in discovering high quality content but the "new page" feature will not qualitatively change the dynamics of these mechanism. We model this feature in the following manner: Before the main voting process starts, each article receives a constant number, say c, of votes. After that, everything proceeds in the same manner as described in Section 3. We are now essentially where we started with the original model, with the exception that the initial ranking of articles will be according to the scores received during the initial feedback. Instead of drawing articles from the quality distribution F, we would draw articles from F conditioned upon having a high score during the initial feedback stage. This will certainly increase curation quality and discovery efficiency but the dynamics are fundamentally the same as above.

B. ATTENTION PAST THE FRONT PAGE

One of the main assumptions of our model and analysis was that the attention of all users is exclusively focused on the main page. We relax this assumption here to a more realistic one, and show that our analysis is still exactly right with probability $O(e^{-n})$. The assumption that we are making in this section is that *the amount of attention that the k articles of the front page receive is exponentially larger than the attention that the $(k+1)st$ article receives.* This is a well justified assumption by all empirical data.

Let's call a_{k+1}, q_{k+1} and s_{k+1} the article in the $(k+1)st$ spot, it's quality and it's score. Similarly define a_k, q_k and s_k. The event that we need to avoid is when $s_{k+1} > s_k \geq 0$, because in this case a_{k+1} should enter the front page but it can't because a_k is not evicted from the front page. After n steps the maximum score that a_{k+1} can possibly have is $\log n$. This is in the case that quality of a_{k+1} is 1 (and hence every vote it receives is an upvote) and because we assume that the attention that a_{k+1} is receiving is exponentially smaller than the attention that a_k is receiving.

Now we need to bound the probability that $s_k < \log n$. We apply Hoeffding's Inequality.

THEOREM [HOEFFDING'S INEQUALITY] 1. *Let $X_1, X_2, \dots, X_n$ be independent random variables such that $X_i \in [a_i, b_i]$ and let $\overline{X} = \frac{X_1+X_2+\ldots+X_n}{n}$ be their empirical mean. Then $Pr[|\overline{X} - E[\overline{X}]| \geq t] \leq 2exp\{-\frac{2t^2n^2}{\sum_{i=1}^{n}(b_i-a_i)^2}\}$.*

In our case the X_i's can take value $+1$ or -1 and they represent the upvotes or downvotes that a fixed article a receives during the n time steps of the mechanism. So $(b_i - a_i) = 2$ for all $1 \leq i \leq n$. Moreover $E[\overline{X}] = \frac{\sum_{i=1}^{n} E[X_i]}{n} = \frac{\sum_{i=1}^{n} 2q-1}{n} = 2q-1$, where q is the quality of the fixed article a.

The event that we want to avoid is that of $\overline{X} < \frac{\log n}{n}$. Applying Hoeffding's inequality above get that $Pr[\overline{X} < \frac{\log n}{n}] = Pr[|E[\overline{X}] - \overline{X}| \geq 2q - 1 - \frac{\log n}{n}] \leq 2exp\Big\{-\frac{2\frac{(n(2q-1)-\log n)^2}{n^2}n^2}{\sum_{i=1}^{n} 2^2}\Big\} = 2exp\{-\frac{n^2(2q-1)^2+\log^2 n - 2n\log n}{2n}\} \in O(e^{-n})$.

C. MISSING PROOFS

Here, we present the proof for Lemma 5 from Section 5.

PROOF. Under ϵ-random downvoting, an article with quality q receives an upvote with probability $(1-\epsilon)q$. We now recompute curation quality using this probability

$$Q_F(M,\epsilon) = \frac{\int_L^1 q(2 - \frac{1}{(1-\epsilon)q})f(q)}{\int_L^1 (2 - \frac{1}{(1-\epsilon)q})f(q)}$$

where $L = \frac{1}{2(1-\epsilon)}$. Rearranging and pulling out ϵ yields

$$Q_F(M,\epsilon) = \frac{\int_L^1 q(2-\frac{1}{q})f(q) - 2\epsilon\int_L^1 qf(q)}{\int_L^1 (2-\frac{1}{q})f(q) - 2\epsilon\int_L^1 f(q)}$$

Now we want to know under what conditions $Q_F(M,\epsilon) - Q_F(M) > 0$. Substituting in the above definitions and rearranging yields

$$Q_F(M,\epsilon) - Q_F(M) > 0 \leftrightarrow [Q_F(M)|q > L] > E_F[q|q > L]$$

$[Q_F(M)|q > L]$ gives a higher portion of its probability mass to higher quality items since $(2 - \frac{1}{q})$ is an increasing function than $E_F[q|q > L]$ □

Here, we present the proof for Lemma 6 from Section 5.

PROOF. Let $h(q) = 2 - \frac{1}{q}$. Over the range $q \in [.5, 1]$, h is a concave function. An equivalent definition of SOSD is that F SOSD G if $E_F[h(q)] > E_G[h(q)]$ for all concave and increasing functions. Then we wish the following to be true

$$\frac{\int_{.5}^1 q(2-\frac{1}{q})g(q)}{\int_{.5}^1 (2-\frac{1}{q})g(q)} > \frac{\int_{.5}^1 q(2-\frac{1}{q})f(q)}{\int_{.5}^1 (2-\frac{1}{q})f(q)}$$

$$\leftrightarrow E_F[h(q)](E_G[q|q > .5] - (1 - G(.5)))$$
$$> E_G[h(q)](E_F[q|q > .5] - (1 - F(.5)))$$

By assumption $E_G[q|q > .5] = E_F[q|q > .5]$ and $G(.5) > F(.5)$ by SOSD over $[.5, 1]$, implying that $-(1 - G(.5)) > -(1 - F(.5))$. $E_F[h(q)] > E_G[h(q)]$ by SOSD. Thus all terms on the left hand side of the above inequality are greater than the right hand side of the inequality. □