Bayesian Vote Weighting in Crowdsourcing Systems

Manas S. Hardas and Lisa Purvis

311 Ray Street, Pleasanton, CA 94566, USA,
`mhardas@spigit.com, lpurvis@spigit.com`

Abstract. In social collaborative crowdsourcing platforms, the *votes* which people give on the content generated by others is a very important component of the system which seeks to find the best content through collaborative action. In a crowdsourced innovation platform, people vote on innovations/ideas generated by others which enables the system to synthesize the view of the crowd about an idea. However, in many such systems *gaming* or vote spamming as it is commonly known is prevalent. In this paper we present a Bayesian mechanism for weighting the *actual vote* given by a user to compute an *effective vote* which incorporates the voters history of voting and also what the crowd is thinking about the value of the innovation. The model results into some interesting insights about social voting systems and new avenues for gamification.

1 Introduction

In collaborative and social online environments, people post inputs and other people collaborate, comment, and vote on those inputs, resulting in a sense of what the crowd thinks about a particular input, topic, or idea. In particular in a social innovation platform like Spigit, people post innovations or *ideas* and others collaborate around those ideas to come up with the ones with the greatest perceived value. In order for such a collective intelligence platform to synthesize what the best ideas are and be able to move those forward into actionable results for a business, it is important to have a mechanism which captures the opinion of the crowd about a specific idea. The simplest of these is the rating/voting mechanism. Thus voting is central in any social collaborative innovations platform.

However, it is often found that as the size of such a social system increases so does the noise in the voting data, sometimes due to error but more often than not due to mischievous intent. People vote-up their own ideas or their friends ideas irrespective of what the actual perceived value of the idea may be. This leaves ample scope for gaming the system by allowing people to artificially prop up or beat down any idea or ideas irrespective or the true value of the idea. This is called as *gaming the system*. It is defined as the phenomenon when a person or a group votes for an idea based on not the true perceived value of the idea but rather the social relationship between the voter and the idea generator. This is inherently harmful for the rating mechanism and therefore for the whole system. Thus, gaming is when the true value of an idea is not a determining factor in the number of up or down votes it receives. Gaming can happen in the context of being untruthful about the true value of an idea. Only when a person thinks it is a good idea but still gives a down vote in order to malign the idea creator, is it gaming.

Conversely giving up votes to ideas generated by friends in spite of perceiving them as bad ideas is also gaming.

But this begs the question - what is the *"true"* value of an idea and how do we calculate it? It is impossible to know the value unless the idea is put into practice and the resultant performance is measured. However that is impractical in practice and also in theory because of the sheer number of resources required and variables involved. Crowdsourcing is the only way to know what the value of an idea really is, i.e., by gauging how a crowd reacts to the idea. Thus, if the crowd thinks the idea is good and a voter votes up the idea then that vote should get a higher weight. If, however, the voter votes the idea as bad when in fact the crowd thinks the idea is good as reflected by the up and down votes the idea receives, then, there may be a situation where the system is being gamed. Therefore the vote should get a lower weight. Of course there may be cases in which the voter may truly perceive the idea to be low valued than what the crowd thought without the intent of gaming the system. This creates a problem which we discuss later.

In this paper we present a novel Bayesian mechanism to weigh the vote given by a voter to compensate for the possibility of gaming or vote spamming (like giving up/down votes to an idea without much thought only based on the social relationship). The mechanism incorporates the voters history and the evidence about the idea i.e. what the crowd thinks about the idea while calculating the weight. It is seen that some interesting properties about voting emerge from the model which can be used to better voting systems in collaborative innovation platforms and also used for gamification.

2   Wisdom of crowds

An important assumption is made in the above discussion regarding the ability of the crowd to vote for the best idea. It is often seen that aggregating the judgements of a number of individuals often results in an estimate which is closer to the true answer than any of the "best" individual estimates. This phenomenon called as the *wisdom of crowds (WOC)* [Surowiecki 2004]. The WOC idea is currently used in several real world applications like prediction markets [Dani et al. 2006; Lee et al. 2009], vote spam filtering [Bian et al. 2008], image annotation, forecasting [Turner et al. 2011], decision making and problem solving [Yi et al. 2010]. Miller et al. (2009) show that the WOC idea is very effective finding solutions to rank ordering problems using many different methods of vote aggregation. It has been repeatedly shown that even with a very small population, the aggregation of solutions obtained from crowds are often better than any individual solution.

3   Gaming in crowdsourced platforms

Collaborative social innovation platforms like Spigit draw heavily on active reader engagement such as voting, reviews and commenting to improve other peoples innovations. This feedback is central to ranking and filtering high valued innovations from a large set. Unfortunately as the number of individuals increases the quality of the feedback degrades sometimes due to noise error but many a times due to perceived personal

profits or *trolling*. This is commonly called as vote spam. The phenomenon of vote spam and methods to handle it have been research reasonably well in the recent years. Various machine learning algorithms have been devised to tackle this problem [Bian et al. 2008; Su et al. 2007; Jeon et al. 2006; Agichtein et al. 2008; Radlinski et al. 2006; Immorlica et al. 2005; Mehta et al. 2007]. Most of these methods try to learn a ranking function from the voting data without preprocessing the actual votes for erroneous votes. In our model the actual vote (V^A) is not considered as is but an effective vote (V^E) is computed from it.

$$V^E = p * V^A \qquad (1)$$

where *p* is the weight of the actual vote; $V^A = [-1; 0; 1]$ and $V^E = [-1 : 1]$.

Type	up	down
1-1	one specific voter 'up' votes a specific person in a planned mutual appreciation using each other to gain reputation regardless of the quality of the content	one specific voter 'down' votes a specific person in an attempt to malign the other regardless of the quality of the content
1-many	a voter always gives positive feedback to a specific group of people	a voter always gives negative feedback to a specific group of people
many-1	a group of people try to drum up the reputation of a specific voter	a group of people try to malign a specific voter
many-many	two groups of people try to drum up each others ideas	two groups of people try to malign each others ideas

Table 1. Types of gaming in voting systems

There are many ways in which a voting system can be gamed. According to the WOC phenomenon the crowd always comes up with the true value of an innovation. Therefore, in this model a vote which is against the direction in which the crowd is voting (i.e. a vote against determining the true value of an innovation) has the potential of being a spam vote. Therefore the weight of a vote directly depends upon what the crowd is thinking about an idea. If the crowd thinks that the idea is good and the voter gives it an up vote then the weight should be more. Similarly if crowd thinks it is a bad idea and the voter gives a down vote, then the weight should be more. In general voting with the crowd should matter more. This also prevents from gaming as the voter simply cannot give up votes to friends ideas because if the crowd thinks that the idea is bad then by giving it an up vote the vote is not going to be weighted a lot. Thus going against the crowd will negatively affect the weight of the vote. Table 1 shows the types of possible gaming in voting systems.

However a vote against the crowd doesn't necessarily mean that the system is being gamed. For instance, what if an idea generator produces a high valued idea (which cannot be known beforehand) and asks his/her friends to vote up the idea. Does this constitute gaming? A crowdsourcing innovation platform should encourage such behavior in which idea generators vie for the favourable opinion of others. However, according to our earlier definition of gaming, any voting done in the context of social relationships rather than perceived value of an idea, is gaming. What should be done in cases like

these? Subtle questions like these arise from the discussion, which are left to be tackled at a future time.

In this paper a crowd is defined by the majority, which may or may not be an ideal definition in itself. For instance, if 100 votes are cast on an innovation out of which 60 people vote the innovation as good whereas the rest 40 vote it down, then we say that the crowd is voting the innovation as good. However, there are clearly two crowds here, the one which votes up and the other of lesser magnitude which votes the idea down. If this definition that the majority constitutes as the "crowd" is correct or not is a matter of open debate. Kuncheva et al. (2003) briefly discuss this problem on the limits a majority vote accuracy.

4   Desirable characteristics of weighting model

When should the vote matter the most? Say an idea has received 97 up votes and 2 down votes. In this scenario giving it an up vote or a down vote is not going to matter a lot. Voting with the crowd and giving it a +1 (1 up vote) only confirms what is already known, that the crowd thinks its a good idea. Similarly, by giving it a -1 (1 down vote) does not at all change the facts that the crowd still thinks it is a good idea by far. In a case when the difference between the up votes and down votes is too large, the actual vote does not matter that much and that should reflect in the effective vote.

However, not all ideas start off that way. Ideas start off by slowly accumulating up and down votes and there are moments in the ideas lifetime when the difference between positive and negative votes is not too much. Obviously, voting on these ideas which are placed on such a balance is quite important and the weights for these votes should be more. These votes help in swaying the vote on either side and therefore are really important.

Consider this example. An idea has 50 up votes and 50 down votes. The 101st vote is either going to tilt the vote towards a positive or tilt it towards a negative. In cases like these, there is no clear definition of what a crowd constitutes. For example, in case of 97 up and 2 down, there is clearly a crowd, as defined by the majority which is saying that the idea is more good than bad. However, when there are 50 ups and 50 downs there is no clear definition of a crowd. In these cases the voting person is going to help form the crowd either in the positive or negative. Therefore the weight of this vote should matter more. On a side note, what makes a crowd is another interesting question which we reserve for the future. That being said there are statistical methods for ranking which can neutralize the effect varying voting populations like Wilson's algorithm or other confidence proportion rating methods.

5   Factors that affect the weight of a rating

For example consider a transaction in As table,

Table 2 shows an example of transactions stored at for every person. The table stores all the votes that the ideas generated by a person has received. For example, B has given idea2 generated by A a vote of -1. The objective of the model is to correctly determine

ideaId	voter (ϑ)	V^A	V^E
idea1	B	1	0.98
idea1	C	1	0.8
idea2	B	-1	-0.98
idea2	C	-1	-0.8
idea2	D	1	0.6

Table 2. Example of a transaction table of person A

the weight of this vote should get after adjusting for the possibility of a spam vote by B. The two factors which affect the weight of the vote are;

1. The history of the voter to side with the crowd
2. The evidence about the idea in terms of what the crowd thinks

Let these two be defined by events, Event C (the hypothesis) = vote with crowd.

Event I (the data/evidence about the value of the idea) = the cumulative crowd sentiment about the idea

The overall sentiment about an idea is computed as follows,

$$P(event\ I) = P(cumulitive\ sentiment\ about\ idea) = \frac{|up\ votes - down\ votes|}{total\ votes} \tag{2}$$

For example, is an idea gets 7 up votes and 3 down votes the probability that crowd thinks its a good/bad idea is $|7 - 3|/10 = 0.4$. Similarly other way round if it gets 3 up votes and 7 down votes then the probability that crowd thinks its a good/bad idea is $|3 - 7|/10 = 0.4$. This value captures the cumulative feeling about the idea.

Now we can compute the probability of the hypothesis i.e. voting with the crowd given what the crowd is thinking about the idea. This probability is modeled using Bayesian inference,

$$p = P(C|I) = \frac{P(C) * P(I|C)}{P(I)} \tag{3}$$

where

1. $P(C|I)$=Is called as the posterior probability which we are trying to calculate. It is called as posterior because it is calculated after taking the data into account.
2. $P(C)$=The prior probability. It is the prob. of the person voting with the crowd i.e. persons history of voting for the good idea. This probability can be initialized depending upon the predefined ROLE of the node in the social network. For example, it can be assumed that the prior of an EXPERT can be very high as he is expectedly assumed to pick the good idea. However, the prior can be approximated upon over time using recursive Bayesian inference.
3. $P(I|C)$=Is called as the likelihood i.e. the probability of the idea being a good/bad idea given voter votes with the crowd. If the +1 votes $\geq$ -1 votes, then this is the probability that crowd thinks its a good idea given voter votes +1, and vice versa when -1 votes $\geq$ +1 votes.

4. $P(I)$=This is the data/evidence about the idea. It is the probability that the idea is a good/bad given voter votes with/against the crowd respectively. Therefore,
$P(I) = P(I|C) * P(C) + P(I| \sim C) * P(\sim C)$

5.1 Scenarios

Example 1: Voting with the crowd Consider this scenario in which an expert has a prior of voting with the crowd 0.9 and an idea gets 7 up votes and 2 down votes. Therefore, $P(C) = 0.9$ and $P(\sim C) = 0.1$

Now, $P(I|C)$=is the probability that crowd thinks idea is good/bad given voter voted with the crowd. 7 people think its a good idea whereas only 2 think is not a good idea. So the crowd seems to think it is a good idea. So if the voter votes with the crowd then he would give a +1 to this idea. Now, up votes are 8 and down votes are 2. Therefore, $P(I|C) = |8 - 2|/10 = 0.6$

$P(I| \sim C)$=is the probability that crowd thinks idea is good/bad given voter voted against the crowd. So even though 7 people think its a good idea and only 2 people think it is not a good idea, the voter gives a -1 siding with the minority instead of the crowd. Now the total count of up votes is still 7 but count of down votes is 3. Therefore now, $P(I| \sim C) = |7 - 3|/10 = 0.4$

Therefore the total probability that crowd thinks idea is good/bad is,
$P(I) = 0.6 * 0.9 + 0.4 * 0.1 = 0.58$
Using Bayes theorem the probability of the hypothesis is,
$P(C|I) = \frac{0.9*0.6}{0.58} = 0.931$

Thus there is a 93.1% percent chance that the person is voting with the crowd given his history and the evidence about the idea that 7 out of 9 people think that its a good idea. Observe that the initial probability that the voter will side with the crowd was 90%. It now increases to 93.1% since the voter votes with the crowd. Similarly if the crowd thinks that the idea is bad as reflected by the number of down votes it receives and it voter gives a -1 to this idea, then the weight of the vote (p) goes up to 93.1

Example 2: Voting against the crowd What if the voter votes against the crowd? Consider that like in example 1, crowd thinks it is a good idea but the voter decided to go against the crowd giving -1 when crowd thinks idea is good and giving +1 when crowd thinks idea is bad. In this case $P(I|C) = 0.4$ and $P(I| \sim C) = 0.6$. Therefore,
$P(C|I) = \frac{0.9*0.4}{0.4*0.9+0.6*0.1} = \frac{0.36}{0.42} = 0.8571$
In both examples if the voter decides to go against the crowd weight of the vote decreases from 90% to 85.71%.

6 Simulation

Figure 1 shows the behavior of probability of the hypothesis i.e. probability of voting with the crowd against the number of up and down votes. The simulation is set up as follows, the number of up votes is set constant at 50 while the number of down votes is varied from 1-100 and the prior probability of the voter voting with the crowd is constant at 0.9. In the four sub graphs four different scenarios are created.
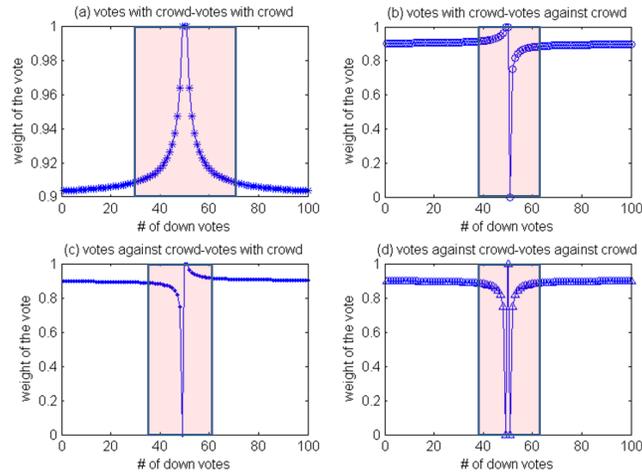
Fig. 1. up/down votes versus weight of the vote

Scenarios:

1. In 1 (a) the voter votes with the crowd when $up > down$ as well as $down > up$ votes. So in the first half when $up = 50$ and $down < 50$, the voter always votes with the crowd. As it can be seen as the difference between the up and down votes decreases, the weight of the vote starts to increase too till it reaches a maximum of 1 at $up = 50$ and $down = 49$. At this point voter votes with the crowd to make it $up = 51$ and $down = 49$ and thus $p = 1$.

   At the center point when $up = 50$ and $down = 50$ there is no crowd. The voters vote is going to help create the crowd and therefore is very important. Therefore p at this point is assumed as 1.

   In the second half when $up = 50$ and $down > 50$ again it can be seen that as the difference between the votes increases the value of p again come down from 1 to 0.9. In this part too the voter votes with the crowd always voting down since $down > up$ and therefore in both cases his p value does not go below the original 0.9.

2. In 1 (b) the voter first votes with the crowd till $up = 50$ and $down < up$ and then votes against the crowd when $up = 50$ and $down > up$. Again it can be seen that while voting with the crowd the p value increases from 0.9 to 1 for $up = 50$ and $down = 49$. At $up = 50$ and $down = 50$, there is no crowd and the voter helps create the crowd therefore the weight of the vote is max at 1.

   Now for $up < down$, the voter starts voting against the crowd. It means that even though there are less up votes than down, the voter starts voting up. Consequently at $up = 50$, $down = 51$ the voter votes against the crowd making it $up = 51$ and $down = 51$. This makes the $p = 0$. Intuitively it is very obvious. Before the person voted, there was a crowd ($up = 50, down = 51$) but this persons vote is neutralizing the crowd so that in the next step ($up = 51, down = 51$) there is no

crowd. Thus, this persons vote is actually harmful to the process of generating a sentiment about the idea as it neutralizes the current sentiment. Therefore the value of p comes to 0.

It can be seen that as the difference between the up and down votes increases again the weight of the rating beings to increase rapidly in the beginning when the difference is small but slowly afterward till it reaches the original 0.9. However since the vote is always against the crowd it never goes beyond the original 0.9.

3. In 1 (c) the voter first votes against the crowd for $up > down$ and then with crowd for $up < down$.

While voting against the crowd it is seen that p value goes on decreasing from 0.9 as the difference between the votes decreases. For $up = 50$ and $down = 49$, the voter votes against the crowd making it $up = down = 50$. This vote neutralizes the sentiment and therefore its p value is 0.

However, in the next half when $down > up$ the voter votes with the crowd i.e. always gives a down vote. For $up = 50$ and $down = 51$, the voter votes with the crowd making it $up = 50$ and $down = 52$, thus consolidating the sentiment even more. Therefore the p value for this scenario is maximum 1. As the difference between the votes begins to increase and the voter keeps on voting with the crowd, the p value starts to decreases and finally reaches the original at 0.9. It does not go below 0.9 since the voter is voting with the crowd.

4. In 1 (d) voter votes against the crowd irrespective of $up > down$ or $up < down$ votes.

When $up > down$ the voter always gives a down vote. Therefore the weight decreases from the original 0.9. This decrease in weight is more pronounced for lower differences between up and down votes until for $up = 50$ and $down = 40$, the voter neutralizes the vote by making it $up = down = 50$ and the weight jumps to 0.

Same thing happens for $up = 50$ and $down = 51$, voter votes against the crowd making it $up = down = 51$ and therefore p remains at 0.

After that the crowd begins to form but since the voter is always voting against the crowd the p values never goes beyond the original 0.9.

7   The critical vote window

From figure 1 the behavior of p against voter opinion (up and down votes) can be seen. It takes off after a certain point in the vote difference range when the opinion about the idea is the most uncertain. When the crowd has already made up its mind about the idea e.g. if the idea has 50 up votes and 5 down votes or 50 up and 95 down votes, the weight of the vote does not matter too much. Since the prevalent sentiment about the idea is already established, the new vote is not going to help sway the vote in either side. It will either consolidate what is already known or add to the minor dissent. Either way it is not too important.

However when the difference between the up and down votes is small in some range, it means that the crowd has not made up its mind about the idea. There is no exact sentiment about this idea and almost as many people like it as those who hate it. In this scenario, if the voters vote helps in tilting the sentiment about the idea in either

ways it is extremely important. Therefore the weight of the idea matters more. As it can be seen from the graphs, the weight takes off towards 1 or towards 0 depending upon whether the person votes with or against the crowd. The votes that help sway the majority are the most important. This window of votes in which the difference between the up and down votes is very small is called as the *critical vote window* shown by the shaded portion in Figure 1. The time at which a person votes on an idea that is on the edge is purely coincidental and random. If a person happens to vote on an idea in the *critical vote window* and the person votes with the crowd then the weight of the vote will be more and vice versa. The critical vote window can be used to project ideas which need immediate attention or gamification by designing contests around these ideas.

8 The problem of herd behavior

In this system the implicit assumption is that voting with the crowd is is equivalent to voting for the true value of the idea. This means that if majority of the people think it is a good idea and some voter thinks it is a bad idea and gives a -1 vote, then the voter will be penalized for this behavior. The voter cannot view the number of up/down votes an idea had received until after they have voted. Thus, although the voter is oblivious to this, conformism is encouraged and independent thinking is punished in the system. We term this as the phenomenon of *herd behavior*. It is a problem when a voter goes against the crowd with no intention of gaming the system. However we also contend that this outcome is also desirable as it appeals to the crowdsourcing aspect of a collaborative voting system. Since the "true value" of an idea cannot be determined the problem is fundamental and depends upon what the definition of a "good idea" exactly is. In the herd behavior phenomenon, the talent of being able to spot a good idea among bad ideas may be lost but what is gained is the emergence of only those ideas whose true value is high in the face of gaming.

9 Conclusion

We showed that gaming can potentially be controlled by weighting the importance a vote gets. This weight is calculated by a simple Bayesian mechanism which incorporates the history of the voter in voting for the high valued idea and the evidence about the idea in terms of what the crowd is thinking. This model leads to some interesting resultant observations like the existence of the *critical vote window* which can potentially be employed to improve the process of voting and in designing games around voting.

References

1. E. Agichtein, C. Castillo, D. Donato, A. Gionis, and G. Mishne. Finding high-quality content in social media with an application to community-based question answering. In Proceedings of WSDM, 2008.
2. J. Bian, Y. Liu, E. Agichtein, and H. Zha. A few bad votes too many?: towards robust ranking in social media. In Proc. 4th Intl. Workshop on Adversarial Information Retrieval on the Web (AIRWeb), pages 5360, 2008.

3. Dani, V., Madani, O., Pennock, D.M., Sanghai, S.K., & Galebach, B. (2006). An Empirical Comparison of Algorithms for Aggregating Expert Predictions. In Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI).

4. N. Immorlica, K. Jain, M. Mahdian, and K. Talwar. Click fraud resistant methods for learning click-through rates. In Workshop on Internet and Network Economics (WINE), 2005.

5. J. Jeon, W. Croft, J. Lee, and S. Park. A framework to predict the quality of answers with non-textual features. In Proceedings of SIGIR, 2006.

6. Kuncheva, L.I., Whitaker, C.J., Shipp, C.A., 2003. Limits on the majority vote accuracy in classifier fusion. Pattern Analysis & Applications 6 (1), 22e31.

7. Lee, M.D., Grothe, E., & Steyvers, M. (2009). Conjunction and Disjunction Fallacies in Prediction Markets. In N. Taatgen, H. van Rijn, L. Schomaker and J.Nerbonne (Eds.) Proceedings of the 31th Annual Conference of the Cognitive Science Society. Mahwah, NJ: Lawrence Erlbaum.

8. B. Mehta, T. Hoffmann, and P. Fankhauser. Lies and propaganda: detecting spam users in Collaborative Filtering. In Proc. of the 12th International Conference on Intelligent User Interfaces (IUI), 2007.

9. Miller, B., Hemmer, P. Steyvers, M. & Lee, M.D. (2009). The Wisdom of Crowds in Ordering Problems. In: Proceedings of the Ninth International Conference on Cognitive Modeling. Manchester, UK.

10. F. Radlinski and T. Joachims. Minimally invasive randomization for collecting unbiased preferences from clickthrough logs. In Proc. of the National Conference on Artificial Intelligence (AAAI), 2006.

11. Q. Su, D. Pavlov, J. Chow, and W. Baker. Internet-scale collection of human-reviewed data. In Proc. of the 16th international conference on World Wide Web (WWW2007), 2007.

12. Surowiecki, J. (2004). The Wisdom of Crowds. New York, NY: W. W. Norton & Company, Inc.

13. Turner, B., & Steyvers, M. (2011). A Wisdom of the Crowd Approach to Forecasting. 2nd NIPS workshop on Computational Social Science and the Wisdom of Crowds.

14. Yi, S.K.M., Steyvers, M., Lee, M.D., & Dry, M. (2010). Wisdom of Crowds in Minimum Spanning Tree Problems. Proceedings of the 32nd Annual Conference of the Cognitive Science Society. Mahwah, NJ: Lawrence Erlbaum.