CrowdRank: A Simple Ranking Algorithm for Crowdsourced Rating Systems with Uneven Participation

Jonathan Bartlett

June 3, 2019

1 Introduction

Public rating systems are difficult to score well. Voting systems tend to simply favor what is already popular. Averaging systems tend to have significant variance if there are not enough people scoring.

For instance, let's say that I run a songwriting contest and have 100 entries. I then put it out to a public vote on the Internet to see who wins. Most people are not going to listen to all 100 songs. If I do a simple "thumbs up" approach and count how many votes a song has, then whichever songwriter has the best existing following will simply tell their fans to vote for them, and it will simply devolve into a popularity contest.

Let's say instead I do a rating system where you can rate a song between 0 and 100. Now, songs by popular artists will actually be negatively weighted because they will have more visibility for negative ratings. It is not hard for a few votes to be all 100s, but it is hard for a thousand votes to be that way. Thus, those who have fewer ratings have an advantage.

The goal, then, is to come up with a fair way of handling public ratings which takes into account both the average score that people assign and the relative certainty that we have that the score is representative of the "true" score.

2 The Model

This problem actually becomes rather easy once an appropriate mental model is devised. Assuming a normal distribution of actual scores that come in around a "true" value for a particular score for an entry, what is the range of possible score values based on the scores that have been submitted so far?

Take a concrete example. Let's say that Song A has 12 votes with an average score of 60. What is the range that the "real" score should lie in? The main open question when dealing with statistics is what confidence level we want to deal with. For this example, let's say that we want to maintain a 95% confidence interval. That means that we want to know what the range is of two standard deviations from the mean.

With only 12 samples, this leads to a fairly wide interval, with the real score being between 32 and 88. However, as we add more samples, this range narrows in to the average. If we have 24 samples and maintain the same average, then our range is restricted to between 40 and 80. At 144 samples, the range narrows to 52–68.

So, with a few scores, the possible "real" score has a very wide range. However, as more and more scores come in, the range narrows further and further.

Now, even though these rankings get tighter variances with more scores, the average value for the scores remain what they were. So how do we convert this into a more legitimate ranking system than we had before?

What we can do is simply rank the songs using their lowest possible scores according to the chosen confidence interval. That is, we have established statistically what the lower bound for their score is. Therefore, we can definitively give them that score because we know they have earned *at least* that score.

This minimal defensible score will be called the CrowdRank score.

Let's say that Song A has 144 rankings that average to 60, and Song B has 25 rankings that average to 70. Which song should be ranked higher? As we have already noted, Song A's "real" score has a potential range of 52–68. Song B, because it has fewer score submissions, has a wider potential range of 50–90. Since the lowest defensible score of Song A is 52, and the lowest defensible score of Song B is 50, that means that Song A will be ranked higher than Song B.

The actual ranking will be dependent on the confidence level that is chosen for the rankings. The higher confidence levels will take many more rankings for the scores to approach their averages.

3 The Calculation

The calculation of each entry's score is fairly straightforward. It is basically the inverse of standard statistical scores.

p The population size

n The number of samples (i.e., number of rankings on a particular entry)

z The confidence level desired, expressed as a z-value (the number of standard deviations that a given confidence level uses—2.58 for 99% confidence, 1.96 for 95% confidence, etc.)

e The margin of error for the confidence interval, expressed as a decimal (i.e., 0.25 for ±25%)

s The average score of the samples (expressed as a real number between 0 and 1—in the present example we would divide all scores by 100)

m The expected value. Choosing 0.5 is a "most-safe" value.

Typically, the number of needed samples is determined from the desired margin of error, using

$$n = \frac{z^2 m(1-m)}{e^2}. \tag{1}$$

Rearranging to find the margin of error from the sample size, we find

$$e = \sqrt{\frac{z^2 m(1-m)}{n}}. \tag{2}$$

Since our results are distributed as a percentage anyway (a score of zero to one), the crowdrank is just the score $s - e$. Simplified using $m = 0.5$, the CrowdRank calculation for a particular entry is

$$\text{CrowdRank} = s - \sqrt{\frac{0.25\, z^2}{n}}. \tag{3}$$

If the samples are taken from a restricted population of size P (say, all the members of a club), you can get an even better measurement from the following:

$$\text{CrowdRank} = s - \sqrt{\frac{0.25\, z^2}{n}\frac{P-n}{P-1}} \tag{4}$$

4 Difficulties

There are two primary difficulties with this system. The first is that, if there are too few rankings for each entry, the confidence level will fall off to zero. This can be mitigated by varying the desired confidence level based on the average rankings per entry.

The other difficulty is in communicating the results to end-users. It is difficult for them to understand why having 144 people all giving a ranking of 60 might translate to a CrowdRank of 52. Having scores whose origin is not transparent can lead to a lack of confidence in the system. However, because the discount to the scores is fixed for the number of entries, you can communicate this as the number of points that are discounted for a given number of entries. For instance, if you are using the 95% confidence interval, then you can post that receiving 23–25 entries will result in a 20 percentage point discount.

5 Conclusion

This paper introduced a system of averaging crowdsourced rankings that appropriately discounts ranking averages based on the number of submissions.

This can be used in any place where a variable number of crowdsourced rankings might be received. It removes the "popularity contest" problem of simple voting, as well as the problem of having too few rankings available in a generic averaging system.