WINGS' scientific whitepaper, version 0.4

Serguei Popov

September 28, 2016

Abstract

We describe our system, a forecasting method enabling decentralized decision taking via predictions on particular proposals, and resolutions via the results of proposals decisions, based on resulting value of particular assets.

Preliminaries and forecast ratings. Let B_i be the balance of the ith account, and $B = \sum B_i$ be the total number of tokens that exist at the moment (might be not constant, due to inflation). We assume that some account (referred to as the *proponent*) submitted a proposal, and a poll about a proposal is being held during some pre-established time period. Let X be the minimal amount of funds that the proponent deems as minimally necessary for the proposal execution[1]. Let F_i be the forecast of the ith account on how much the proposal will collect; $F_i = 0$ means that the proposal is marked as "spam" by this account, and (just a notational convention) $F_i = -1$ means that the account did not participate at all in the poll. It is assumed that F_i's are only revealed in the end of the forecasting period, that is, the accounts cannot adjust their forecasts based on the information about the other accounts' forecasts.

Among the key variables in our model are the forecast ratings, which measure the account owner's ability to issue accurate predictions. There are, in fact, two forecast ratings: the *numerical* one (which is changed after each vote), and the *qualitative* one, which indicate to which *category* the account belongs. It is similar to e.g. internet forums, where a user can be "junior member", "member", "senior member" etc.; the accounts of higher categories have more "power" in the system. Specifically, let $r_i \in [0, R]$ be the current numerical forecast rating for account i (r_i is a real non-negative number, and R is some fixed maximum value for it), and g_i

[1] as we comment below, the proponent will not need to specify the amount X if the community decides to remove the minimal amount required for proposal graduation; that rule is flexible and can be turned on or off

be the qualitative forecast rating: $g_i \in \{0, 1, 2, \ldots, G\}$ determined from r_i in the following way:

$$g_i = \begin{cases} 0, & \text{if } r_i < a_1, \\ 1, & \text{if } a_1 \leq r_i < a_2, \\ \ldots \\ G-1, & \text{if } a_{G-1} \leq r_i < a_G, \\ G, & \text{if } r_i \geq a_G, \end{cases} \tag{1}$$

where $0 < a_1 < a_2 < \ldots < a_G < R$ are parameters.

Marking a proposal as spam. Let

$$s = \frac{\sum\limits_{i:F_i=0} g_i}{\sum\limits_{i:F_i \geq 0} g_i} \tag{2}$$

be the weighted (by g_i's) proportion of voters that marked the proposal as spam. Let $s_0 > 0$ be some pre-established threshold for deciding that the proposal is marked as spam *by the whole system* (that is, the system decides that the proposal is spam whenever $s \geq s_0$). If the majority votes that that the proposal is spam, then the proposal's fee is divided between those that voted for spam, and no adjustment of ratings is made[2]. Note that it is not likely that a rational voter would mark a *good* proposal as spam, since this would give him no gains in case others do vote favorably, and the profit from supporting a good proposal is likely to be much better.

Value of the poll. This is a characteristics of a poll which measure how "important" it is for the system. If the system decides that the proposal is not spam, we calculate the *value of the poll* in the following way:

$$v = \ell^2 q, \tag{3}$$

where ℓ is the proportion of the tokens belonging to the accounts that took part in the poll, i.e., $\ell = B^{-1} \sum_{i:F_i \geq 0} B_i$, and $q = \frac{Q_3 - Q_1}{Q_3 + Q_1}$ is the so-called quartile coefficient of dispersion [1] of the data set $\{F_i : i \text{ is such that } F_i \geq 0\}$ weighted by g_i's[3]. The choice of q as a measure of the relative variation is justified by the fact that it is a robust statistics, so manipulating this quantity is very difficult. The value

[2] otherwise people would make spam proposals only to increase their forecast rating

[3] just repeat ith prediction g_i times for all i (effectively discarding those with $g_i = 0$) and use the usual definition of quartiles

of the poll is supposed to measure how "good and genuine" the poll is. Note that ℓ enters the formula squared; this is because 1 poll where everybody takes part should probably be considered more "important" than 100 polls where only 1% of people take part, and also to prevent spamming.

Listing of the proposal. This is an optional step: as explained below, we may opt for listing *all* the proposals that were not marked as spam. Let

$$\alpha = \frac{\sum\limits_{i:F_i \geq X} g_i}{\sum\limits_{i:F_i > 0} g_i} \tag{4}$$

be the (weighted) proportion of voters (among those who did not vote spam) who predict that the proposal will achieve X. Let $\alpha_0 \in (0,1)$ (it may be fixed, say, $\alpha_0 = 0.3$, or defined by the proposal's author) be some pre-defined *approval threshold*. In case $\alpha < \alpha_0$ the proposal is rejected, and the fee of the proponent is divided between all voters i with $F_i > 0$. Also, note that those who voted still benefit since they are less likely to have their forecast rating penalized for inactivity. Note that we can also set $\alpha_0 = 0$, which means that *all* proposals not marked as spam get listed in the system (and the value of X need not be specified); this basically allows us to disable the minimal requirement.

To reduce spamming possibilities, one may consider the following improvement: calculate proposal listing fee according to the account's forecast rating, i.e., the higher the account forecast rating, the lower the listing fee will be, and vice versa.

Calculating the new forecast rating of voters. In the following, we describe how the account's numerical forecast rating is affected by the outcome of the poll. The idea is, of course, that "accurate" predictions increase the rating, while "inaccurate" ones decrease it. We need, however, to define the rules carefully, to give enough incentive to forecast and to avoid abuses.

Assume that $\alpha \geq \alpha_0$, so that the weighted proportion of voters who predict that the proposal will achieve its goal is enough to pass the approval threshold. Let us describe what happens next. Assume that, in fact, the proposal collected Y funds. Let $b_i = B_i/B$ be the *relative* balance of the ith account, and let us also define the *truncated* relative balance $\hat{b}_i = \min(b_i, \beta_1)$, where $\beta_1 \in (0,1)$ is a fixed number (e.g., $\beta_1 = 0.01$). Let m be the median of the data set $\{F_i : i \text{ is such that } F_i > 0\}$ (note the strict inequality; that is, this data set does not include the votes for spam) weighted by g_i's. Denote also

$$\Delta_i = \hat{b}_i v\Big(\gamma^2 - \frac{(F_i - Y)^2}{(|m - Y| + \theta q)^2}\Big), \tag{5}$$

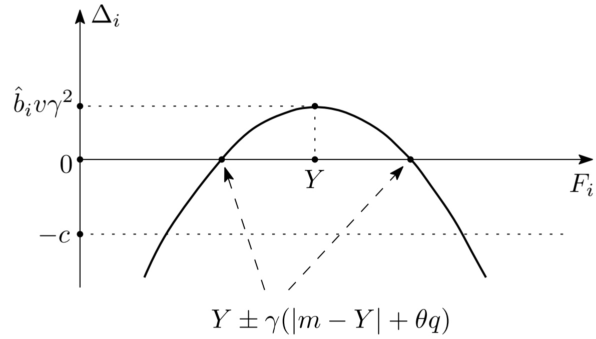

Figure 1: On the definition of Δ_i (as a funcion of the forecast value F_i)

where $\gamma, \theta > 0$ are parameters to be chosen; recall also that v is the value of the poll defined in (3). To explain (5), notice that it gives a positive reward to the forecast rating only in the case $|F_i - Y| \leq \gamma(|m - Y| + \theta q)$, see Figure 1. For example, in the case when $\gamma = 1$ and $\theta = 0$, and all the forecasts happened to be on one side of Y (so that $|m - Y|$ is large), then approximately a half of forecasters will have their rating increased. In practice, one may want to take γ to be slightly larger than 1 and θ to be slightly larger than 0, in order not to excessively penalize the forecasters in case $|m - Y|$ is small[4] (i.e., the wisdom of the crowd turned out to be really wise). Then, the *new* numerical forecast rating of ith account is calculated in the following way:

$$r_i^{\text{new}} = \begin{cases} 0, & \text{if } r_i + \Delta_i \leq 0, \\ R, & \text{if } r_i + \Delta_i \geq R, \\ r_i + \Delta_i, & \text{otherwise.} \end{cases} \tag{6}$$

To explain (5), observe that the reward Δ_i is positive if and only if $|F_i - Y| \leq Yq\sqrt{\gamma}$, that is, the forecast was "close enough" to the outcome Y. One can also consider some modifications of (5), e.g., use $\Delta_i' = \max(\Delta_i, -c)$ for some parameter $c > 0$, to limit the negative increments in the forecast rating (cf. Figure 1).

As will be discussed later, we allow the forecast ratings of the rich accounts to vary more rapidly; this is a necessary measure to avoid Sybil attacks. This can indeed create an impression of "unfairness" towards the users that are not so rich; observe, however, that this problem can be mitigated by the delegate system of the liquid democracy: good predictors will become popular delegates and will be able to increase their forecast rating more rapidly even if they are not so rich.

[4]they should not be penalized because they were close to the right answer, but (5) without the term θq may lead to this unfortunate event

Forecast rating of the proponent. We may also choose to influence the proponents' forecast ratings using the "quality" of their proposals.

First of all, if the proposal was marked as spam, the proponent's forecast rating can be severely penalized (say, even zeroed, or set to a_1, recall (1)). On the other hand, it seems to reasonable to reward the proposals not marked as spam according to (5)–(6) with $F_i = Y$ (i.e., receive the "maximal reward"), thus leading to the reward

$$\Delta_i = \hat{b}_i v \gamma^2. \tag{7}$$

That is, if the proposal not marked as a spam, the higher is its resulting quality (many forecasters participating with disperse forecasts), the higher is the reward for the proponent's forecast rating.

Penalizing accounts for inactivity. It is necessary that the accounts' forecast rating should decrease in case the account remains inactive for some time: otherwise, a malicious user may accumulate many different accounts with high forecast rating (by increasing accounts' ratings successively, and then leaving them inactive until the time comes), and then perform a Sybil attack on the system. Let us discuss how this penalizing should work. A reasonable way to do this is to consider the values of all the polls during a given time period, and compare it to the values of all polls where a given account took part. If the ratio of the latter to the former is less than some threshold, then there is a penalty on the forecast rating. Just as the reward, this penalty should be proportional to the (truncated) relative balance and the polls' values, compare to (5). Also, probably, the votes for "spam" should not receive the "full credit" for participating in the poll (maybe, only some partial one).

It is reasonable, however, to include a mechanism that permits for an account to remain inactive for some time without being penalized, because, for example, the account's owner is on vacations. Possible solutions to it include leasing the balance and/or the forecast rating to some active account for a fee, or even declaring the account to be "frozen" for some period of time. One must, however, require that the leased or frozen account maintains at least some fixed amount of tokens on it. This is necessary in order to prevent that an owner of high amount of tokens inflates the forecast rating of many different accounts successively, and then leaves them "frozen" until the time he controls enough accounts with "good" forecast rating, which would make the Sybil attack possible.

The reward inflation model. As a further improvement (meant to motivate the participants to submit proposals), we are considering the idea to introduce some inflation (with respect to the total number of tokens) into the model.

The system will issue new project tokens at the end of the forecasting action, distributing them to:

- proposal creator;
- proposal forecasters (curators),

The proposal creator will be rewarded according to:

- how many forecasts were submitted (e.g. how many unique users have participated), taking into account the forecasters tokens amounts (to prevent Sybil attacks);
- how wide spread (dispersed) the forecasts were, the more opposite they are, the more controversial the proposal (hence of a higher quality) and the higher the reward.

The proposal forecasters will be rewarded according to:

- how close their forecasts were to the proposal resolution;
- the quality of the proposal they have forecasted on.

All these goals can be achieved, for example, with the following approach: if a proposal was accepted, and the resulting outcome was positive (measured by resulting token price for example, or another parameter provided by an Oracle or another immutable method), then both its author and the participating accounts receive some amount of newly issued tokens, proportionally to Δ_i provided it is positive (for the forecasters)[5].

Attack scenarios. One may suggest the following attack scenarios:

1. An account can try to manipulate its forecast rating by submitting many proposals.
   This will not work because: junk proposals will be marked as spam, and reduce the forecast rating; uninteresting proposals will not have a good value, so the gain will not be considerable; also, since we require a good spread, conspiracies of the form "hey guys, vote exactly Z for my proposal, I'll then invest exactly Z and everybody wins" are unlikely.

[5]recall that the above quantity for the proponent is calculated according to (7) and is always positive, unless the proposal was marked as spam

2. The attacker can create many Sybil identities and inflate their forecast rating, to gain more “decision power”.
   This will not work because: we require to maintain some amount of WINGS tokens on the account in order to be able to increase the forecast rating, and the “empty” accounts will lose the forecast rating; so, unless the attacker is very rich, he won’t be able to do this.

3. The attacker can try to grab most of the “new” (i.e., created because of the inflation) tokens.
   This will not work because: the proposal must have a high value for that, and the concept of value is designed in such a way that it is difficult to manipulate.

Acknowledgments

The author thanks the whole WINGS team for many discussions we had, and also Sérgio Demian Lerner for the comments and suggestions on an earlier draft of this paper.

References

[1] `https://en.wikipedia.org/wiki/Quartile_coefficient_of_dispersion`