

SONM User Reputation Model

Andrei Zavgorodnii, Anastasia Ashaeva

Jan, 2017

Contents

1 Overview	1
1.1 Basic notions	1
2 Technical details	2
2.1 Reputation	2
2.2 Private deals	3
3 Appendix	3
3.1 Basic reputation scenarios	3

1 Overview

This document describes the reputation model that is used to characterise agents in the SONM network. Motivation: lost profit for sellers, lost time for buyers, etc.

1.1 Basic notions

The SONM ecosystem consists of *buyers* and *sellers*. Buyers rent computational resources from sellers to run arbitrary *tasks*; a deal is made for a specific resource configuration and a specific period of time (i.e., not per task).

When looking for a seller, buyer searches the *marketplace*. Deals made via marketplace are called *public* deals¹. When a deal is made, a certain amount of funds is reserved on both buyer's and seller's accounts. For buyer, it's the *full cost* of the deal; for seller, it's a fraction of the full cost (either default or negotiated), hereinafter *the deposit*.

Both buyers and sellers have *reputation* that is based on their activity in the SONM network, i.e., on the outcomes of public deals they had. Any public deal can have three possible outcomes:

- **Mutual satisfaction.** Buyer is satisfied with the service provided by seller. Buyer's and seller's reputation increase in proportion to deal cost.

¹See 2.2 for *private* deals.

- **Settled dispute.** Buyer is *not* satisfied with the service provided by seller, seller admits its fault. Buyer pays seller in proportion to the time it held seller's resources, and seller returns buyer the deposit.
- **Claim.** Buyer is *not* satisfied with the service provided by seller, seller refuses to admit its fault. Both buyer and seller keep their money, but lose their ratings in proportion to deal's cost.

Both buyers and sellers are motivated to have high reputation. Firstly, reputation (combined with deals history) defines the maximum cost of a deal you can make (see 2.1). Secondly, reputation defines the order in which offers are displayed on the market, so in case of automatic matching agents with higher reputation will have better deals.

2 Technical details

2.1 Reputation

SONM's reputation model is based on beta distribution's expected value based on the ideas proposed in [1]. The beta-family of probability density functions is a continuous family of functions indexed by the two parameters α and β . The beta distribution $f(p|\alpha, \beta)$ can be expressed using the gamma function Γ as:

$$\phi(p|\alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1} (1-p)^{\beta-1}, \text{ where } 0 \leq p \leq 1, \alpha > 0, \beta > 0. \quad (1)$$

The probability expectation value of the beta distribution is given by:

$$E[p] = \frac{\alpha}{\alpha + \beta}. \quad (2)$$

Suppose that we take n and m to be the number of positive and negative outcomes for user X respectively; then we define α and β :

$$\alpha = a^X + 1, \beta = b^X + 1. \quad (3)$$

Positive outcomes correspond to successful deals, and negative outcomes correspond to claims. Then reputation for user X is defined as probability expectation value of (1)²:

$$\text{Rep}^X = E[\phi(p|\alpha, \beta)] = \frac{a + 1}{a + b + 2}. \quad (4)$$

In (4) we assume that n and m are calculated by simply counting positive and negative outcomes, but it's natural to take deal cost into consideration. Suppose that ξ_X is the history of X 's deals, and ξ_{X_i} is the cost of i -th deal. Let's define helper functions to tell successful deals from unsuccessful:

²See how this definition behaves in different scenarios in 3.1.

$$\sigma^+(\xi, i) = \begin{cases} \xi_{Xi}, & \text{if } \xi_{Xi} \text{ is successful,} \\ 0 & \text{otherwise.} \end{cases} \quad (5)$$

$$\sigma^-(\xi, i) = \begin{cases} -1 \cdot \xi_{Xi}, & \text{if } \xi_{Xi} \text{ is not successful,} \\ 0 & \text{otherwise.} \end{cases} \quad (6)$$

Then deal outcomes can be weighted by their cost as:

$$n_\lambda^X = \sum_{i=1}^n \sigma^+(\xi_X, i) \quad (7)$$

$$m_\lambda^X = \sum_{i=1}^n \sigma^-(\xi_X, i), \quad (8)$$

where n is the total number of deals made by X .

Now, we might want older outcomes to contribute less to user's reputation. We introduce another parameter, λ , as the *forgetting factor*:

$$n_\lambda^X = \sum_{i=1}^n \sigma^+(\xi_X, i) \cdot \lambda^{(n-i)} \quad (9)$$

$$m_\lambda^X = \sum_{i=1}^n \sigma^-(\xi_X, i) \cdot \lambda^{(n-i)}, \quad (10)$$

where $0 \leq \lambda \leq 1$. It's easy to see that if $\lambda = 1$, nothing is forgotten.

2.2 Private deals

3 Appendix

3.1 Basic reputation scenarios

References

- [1] Josang, Audun and Ismail, Roslan. *The beta reputation system*. Proceedings of the 15th bled electronic commerce conference, Vol. 5, p.2502-2511, 2002.