Simulation of trust propagation in community validated gamification

Alin Brîndușescu[1, *], Vlad Monescu[2] , Nicușor Minculete[2]
West University of Timişoara, Romania[1]
Transilvania University of Brașov, Romania[2]
alin.brindusescu@e-uvt.ro

Abstract: *The new platforms coming out on the social media space are faced from the beginning with a fierce contest. The competition for user attention is hard and the main players (Facebook, TikTok, Instagram, Reddit) have a huge advantage. To overcome these problems many of the newly lunched platform are making use of gamification strategy to keep the user engaged. The gamification strategy usually provides the user with a form of digital reputation but on some platform, it can be even a form of digital currency which can be exchanged for benefits or real products. To validate the user actions there is not always an automatic process, and the platforms are relaying on the community to cross-validate the authenticity and the value of the respective actions. The validation of the user actions can be automated but not always. When the validation cannot be automated, and the task requires human intervention the process is in his most part outsourced to the community and only a small number of the validation will be done by platform operators. The validation done by the platform operators will give credits to the users whose actions were validated. After certain amount of credit is given to a user, he became trusted, and his action can validate the actions of other users. The trusted user will act almost as a platform operator by validating other users' actions. In this paper we simulate and analyse how the trust can be propagated into different social networks configuration. We start by defining a mathematical framework for modelling the users and the actions. After we introduce two different algorithms for propagating the trust in the community and finally for each configuration, we simulate the propagation of the trust and discuss the results.*
Keywords*: SOCIAL NETWORKS, CENTRALITY MEASURES, NETWORK ANALYSIS, GAMIFICATION, FRAUD-DETECTION, TRUST PROPAGATION, SIMULATION*

1. Introduction

Social media and social networks are associated by the normal population with the online social platforms like Facebook, Reddit, Instagram, X (ex-Twitter) but there are a lot of other platforms constructed in a similar way. Many of those other platforms are focused on solving real world problems by attracting contributors from all over the world. Unfortunately, all the existing social media platforms and the newcomers are competing for the same user pool and using creative ways to keep the user attention for as long as possible.

To compete with the established platforms the newcomers need to find innovative ways to fight for the user attention and keep them engaged in the platform. To achieve this, they have to understand the user behavior and to implement engagement strategies. The amount of data generated by the social media platform is incredibly high and processing the information pose a problem to human operators but also to the software processes. Before implementing the engagement, strategy is important to simulate the behavior of the social network and test the assumption. Most of the time the strategy cannot be tested over the complete network but in a simulated environment. One of the most popular engagement strategies used by many of the newly lunched platform is to make use of gamification. The gamification strategy usually provides the user with a form of digital reputation but on some platform, it can be even a form of digital currency which can be exchanged for benefits or real products.

2. Motivation

Gamification is a trendy and popular strategy for increasing the user motivation and engagement. It started by embedding concept from video games into normal activities [1] and it made his way up to education and corporate business. The concept is very simple the user is rewarded for performing different actions, the rewards are accumulated and function as social capital. The concept is widely used by social media platforms but in slightly different ways. Two different ways of usage are described for exemplification:

- On Reddit and Stack Overflow platforms the user is gaining or losing social capital, called karma on Reddit and reputation on Stack Overflow, based on the activity performed and the votes of the other participants. Both points systems are pure virtual and are only providing reputation to the user.
- The Fortunate Planet [2], an eco-friendly movement social media platform, awards users of the platform with eco-credits for performing eco-friendly actions (like adding a new disposal point to the platform, providing evidence that he has correctly disposed recyclable waste, spotting illegal disposed waste, cleaning plastic from nature) or validating the actions performed by other users. The eco-credits are at the base a reputation system, but they have a real-world value because can be used in the embedded marketplace and exchanged for discounts and products provided by the eco-friendly partner shops. The eco-credit system where user can get material benefits is a target for fraudulent behavior. Attacking the reputation-based platforms is, naturally, not so appealing but at the same time it cannot be completely ruled out. Both type of platform need mechanism for validation of the user action and detection of fraudulent behavior. The user actions can be automatically validated by the community but when a monetary incentive is built in the platform is possible that fraudulent groups will be created, and they will try to exploit the system. To overcome this problem new network groups cannot be validated each other actions at least until one "trusted" user validates some of their actions. Some possible validations/restrictions for defining of trusted users are:

- Restriction 1: a user is trusted if his social capital is greater than a specific δ trust defined by the network parameters.
- Restriction 2: a user is trusted if his actions were manually validated by a human operator from social media platform or by other three trusted users.

For the second type of restriction is important to function with as less as possible human interaction and from here the need to properly detect the right candidates for manual verification.

3. Methodology

A. Network Model

To have a common ground for the trust propagation and fraud detection algorithms we start by modelling our gamification process. We defined the U as the set of users and A the set of actions performed by the users. Even if in the real platform there exist direct symmetric or asymmetric relation between the users, due to friendship or followship, we discard this type of relations for the current analysis and consider a bipartite graph $G = (U, A, E)$. On the considered graph the only type of relation possible is between the users and actions and we define $E = \{(v, a) \mid v \in V, a \in A\}$.

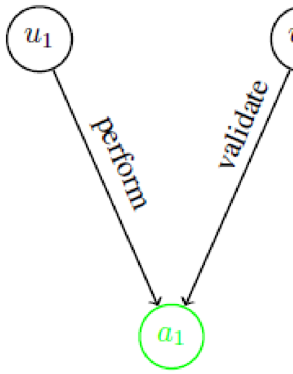

Fig. 1. Relation between the users and the actions

In Figure 1 we have represented the minimal interaction for a gamification process with community validation, here the action *a1* was created by user *u1* and the user *u2* validated the action. Two users *u1* and the user *u2* are considered connected if any of them validated an action of the other. The type of actions available to the users are limited by the design of the platform and we define T = {*ta is a type of action available on the platform*} as the set of type of actions. The users are rewarded for performing an action and for this reason we introduce a function $r : T \rightarrow \mathbf{R}_+$ which will associate to each type of action a reward. Naturally the users should be rewarded for the validation too, the function $v : T \rightarrow \mathbf{R}_+$ will associate a reward to the validation depending on the type of action that was validated. To support the validation process we need to introduce several auxiliary functions:

- $ua : A \times U \rightarrow \{0, 1\}$ defined as:

$$ua(a,u) = \begin{cases} 1, & \text{if } u \text{ has performed action } a, \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

- $va : A \times U \rightarrow \{0, 1\}$ defined as:

$$va(a,u) = \begin{cases} 1, & \text{if } u \text{ has validated action } a, \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

- $valid : A \rightarrow \{0, 1\}$ defined as:

$$valid(a) = \begin{cases} 1, & \text{if } a \text{ has been validated,} \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

- $connected : U \times U \rightarrow \{0, 1\}$ defined as:

$$con(u1,u2) = \begin{cases} 1, & \exists a \in A, va(a,u_1) = 1, \\ 1, & \exists a \in A, va(a,u_2) = 1, \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

B. Validation and trust propagation algorithms

Using the previous definitions we can introduce a naive algorithm, which is not considering the trusted users, for the validation process.

Function Validate (a, u_1, u_2)**:**
  if *not* $valid(a)$ **then**
    if $u_1 \neq u_2$ **then**
      $a \rightarrow valid$
    end
  end

  return

Algorithm 1: A naive validation algorithm

The validation algorithm proposed does not consider any information beside the user who performed and the user who is validating the action. This naive implementation is open to the simplest attack possible: two friends are performing fake activities and cross validating their actions and, in this way, artificially increasing their social capital. A simple protection mechanism, as defined previously in Restriction 1, can be built in by simply allowing the verification only if the user has accumulated a social capital over a specific threshold. For this we define the user trust capital as:

Definition 3.1: Trust capital is the defined as the sum of all the rewards received by a user for performed and validated activities.

$$trustcapital(u) = \sum_{a \in A} ua(u,a)r(a) \quad (5)$$

A simple trust propagation mechanism could be to allow only the user having a trust capital over a specific threshold value to perform validations. By introducing this restriction new users would not be able to cheat the system directly after joining. By adding the threshold value to the validation process the new algorithm will be:

Function DeltaTrustValidate (a, u_1, u_2, δ_{trust})**:**
  if *not* $valid(a)$ **then**
    if $u_1 \neq u_2$ **then**
      if $trustcapital(u_2) \geq \delta_{trust}$ **then**
        $a \rightarrow valid$
      end
    end
  end

  return

Algorithm 2: A simple trust capital validation algorithm

The simple method based on a fix threshold will stop the new users to cheat the system but once one user acquires ufficient trust capital nothing will stop him to use his new gained power to help other users to cheat the system. One more option, as defined previously in Restriction 2, would be to manually validate a number of user and then to let them propagate the trust into the network. For this algorithm we introduce the trusted user as:

Definition 3.2: Trusted used is the defined as a user who's actions were manually verified by the network operator or his actions were validated by the at lest 3 trusted users.

$$trusteduser(u) = \begin{cases} 1, & \text{if } u \text{ has been validated by the network operator or his actions were validated by at least 3 trusted user} \\ 0, & \text{otherwise.} \end{cases} \quad (6)$$

With the new function defined in the previous paragraph the trust propagation algorithm will have two functions: one to validate the user action and 2 to propagate the trust in the network.

For the *Algorithm 3* to perform optimally is important to select the right candidates from the network to be manually validated. When the network is small and can be graphically represented the candidate users can be visually determined but in the social media platform this is not the case and the right candidates must be determined in an automatic way. The most widely used and researched methods, for determining the important users from the network, are the centrality measures.

Function `TrustedUserValidateAndPropagate` $(a,\ u_1,\ u_2,\ \delta_{trust})$ **:**

- **if** *not* $valid(a)$ **then**
  - **if** $u_1 \neq u_2$ **then**
    - **if** $trusteduser(u_2)$ **then**
      - $a \rightarrow valid$ **if** u_1 $valid\ actions \geq> 3$ **then**
        - Add u_1 to trusted users group
      - **end**
    - **end**
  - **end**
- **end**

return

Algorithm 3: A trust propagation and validation algorithm

C. Centrality Measures

Over the last decades a lot of research has been done on the network centrality measures, many centrality measures were proposed ranging from simple ones like degree centrality, and betweenness centrality [3], and PageRank centrality [4], to exotic ones based on quantum theory concepts [5]. Each centrality measure has its own definition and capture a different perspective of the node centrality.For simulation of the trust propagation degree centrality, equation (7) was used. It is equal to the number of connections user *u* has with other users in the network. In the trust validation case considering the users with a lot of connections as candidates for validation the degree centrality can be a very effective tool for selecting the right candidates.

$$C_{Degree}(u) = \sum_{u_i \in U} con(u, u_i) \tag{7}$$

4. Results

To simulate the trust propagation networks structure with 128,256 and 512 randomly connected users has been created using networkx [6] python library. Two simulation were constructed:

• *Random selection:* 5% random users were selected and considered as trusted

• *Central selection:* top 5% users returned using degree centrality 7 were selected and considered as trusted

and using the selected network the trust propagation algorithm 3 was applied.

The simulations were graphically represented for visual inspection and analysis. The initial selected nodes were represented with using a green color, the nodes where the trust was propagated using yellow color and the rest of the nodes from the network using blue color. In the figures 2 and 3 the trust propagation on a network with 128 users and 5% initial trusted users is represented. It can be observed that only a small part of the network will reach a trusted status and the randomly selection of nodes performs better than using the degree centrality to select the nodes.

Based on the initial results of selecting only 5% of the nodes the next set of test was build by selecting 10% of the nodes to be validated and considered as trusted. In the results displayed in [4] it can be observed that the trust propagation covers now a lot more users.

To quantitatively analyse the results the trust propagation ratio was calculated as the percent of network nodes reach the trusted status after the propagation algorithm 3 has been applied to the complete network.

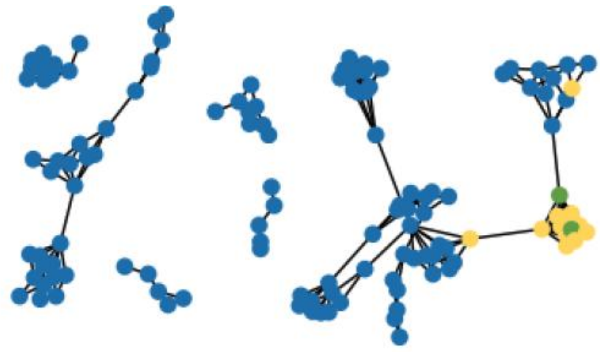

Fig. 2. Trust Propagation with Central selection

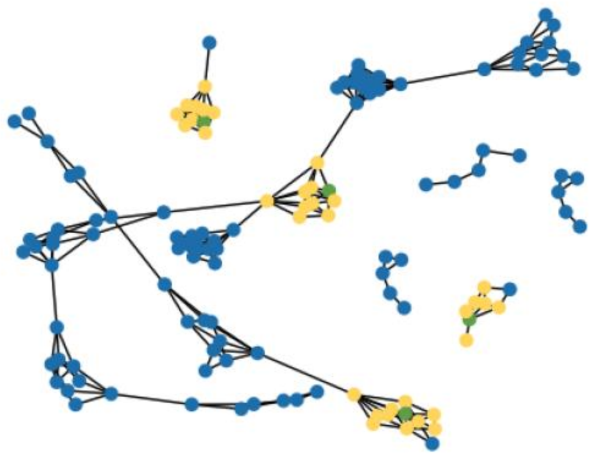

Fig. 3. Trust Propagation with Random selection

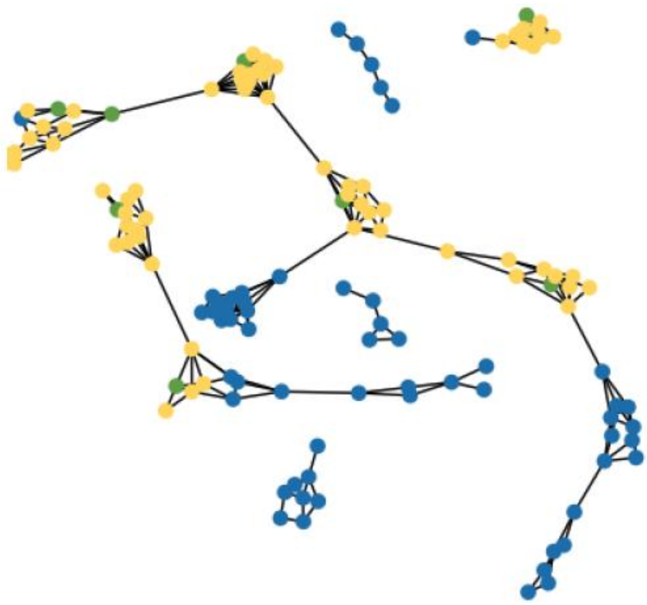

Fig. 4. Trust Propagation with Random selection

For small network size the random selection performs significantly better when comparing with the degree centrality method of selecting the initial nodes but when the network size increases both methods will return similar results. It can be noticed that 5% as initial trusted population is too small and at least 10% will be required to propagate the trust to atleast 50% of the network users.

TABLE I
TRUST PROPAGATION RATIO

Network Size	Selection Method	Initial Size	Ratio
128	Random	5%	27%
128	Degree Centrality	5%	12%
128	Random	10%	55%
128	Degree Centrality	10%	28%
256	Random	5%	30%
256	Degree Centrality	5%	31%
256	Random	10%	51%
256	Degree Centrality	10%	53%
512	Random	5%	34%
512	Degree Centrality	5%	26%
512	Random	10%	52%
512	Degree Centrality	10%	41%

5. Conclusion

Our results highlight the trust propagation algorithm 3 can validated up to 50% of the network when starting with an initial trusted population of 10% equivalent with a reduction of manual work with a factor of 5. The degree centrality measure is not offering any advantages over the random selection of the initial trusted users at the simulated network sizes. Our work can be extended by using other centrality measures to simulate the propagation of the trust in the network or a combination of random and centrality selection. Also, other trust propagation algorithms can be proposed. In future studies, it would be interesting to introduce potential fraud behaviour in the network structure and determine how the trust propagation algorithm behave in that specific scenario.

6. References

[1] S. Deterding, R. Khaled, L. E. Nacke, D. Dixon et al., “Gamification: Toward a definition,” in CHI 2011 gamification workshop proceedings, vol. 12. Vancouver, Canada, 2011, pp. 1–79.

[2] T.F.P. AG, “Tfp: The fortunate planet”, https://thefortunateplanet.com/, 2022

[3] L. C. Freeman, “A set of measures of centrality based on betweenness,” Sociometry, pp. 35–41, 1977.

[4] S. Brin and L. Page, “The anatomy of a large-scale hypertextual web search engine,” Computer networks and ISDN systems, vol. 30, no. 1-7, pp. 107–117, 1998.

[5] Y. Wang, S. Xue, J. Wu, and P. Xu, “Continuous-time quantum walk based centrality testing on weighted graphs,” Scientific Reports, vol. 12, no. 1, p. 6001, 2022.

[6] A. Hagberg, P. Swart, and D. S Chult, “Exploring network structure, dynamics, and function using networkx,” Los Alamos National Lab. (LANL), Los Alamos, NM (United States), Tech. Rep., 2008.