Factors affecting trustworthiness of a Wallet in Bitcoin Network

First Author[#1], Second Author[*2], Third Author[#3]

[#]*First-Third Department, First-Third University*
Address Including Country Name
[1]first.author@first-third.edu
[3]third.author@first-third.edu

[*]*Second Company*
Address Including Country Name
[2]second.author@second.com

***Abstract*—:**
With the advent of technology, cryptocurrencies, especially Bitcoin, have seen a rise in use. Due to the anonymous nature of transactions in a peer-to-peer network like Bitcoin, it is essential to identify malicious wallets and thus take action to minimize risk. To understand the effect of different factors on trustworthiness of a wallet in a Bitcoin Network, several models from diverse fields such as the Peer-to-Peer trust models have been identified, studied and implemented. This paper focuses on the factors affecting the trustworthiness of a wallet in a Bitcoin Network. Centrality Measures have also been studied to find their effect on the trustworthiness of a wallet. The classification from the Eigen trust Model has been used as a reference to measure model performance for the identified models. A significant correlation between the Eigen Trust Score and the Centrality Measures has been observed. From this observation, it can be remarked that Centrality Measures can be utilized to identify a malicious wallet.

***Keywords*— centrality measures, bitcoin, trust, trust models, trustworthy nodes, eigen, goodness, fairness**

I. Introduction

Cryptocurrency enables monetary transactions without the intervention of the banks and the government. The anonymous nature of cryptocurrency transactions has led to their increase in popularity. But the security concerns and the risk involved in the transactions have become major impediments in cryptocurrency's path to becoming mainstream. The untraceability of the transactions has become a boon as well as a bane. The trust analysis of the cryptocurrency network is an effort to recognize the malicious wallet in the network and hence take action to reduce the security risks. The term Wallet and Node are used interchangeably in this paper as a Wallet is seen as a Node in the Bitcoin Network. Centrality measures of a node in a network are used to describe the importance of a node and its interactions with its neighbourhood. The goal of the paper is to determine the role of centrality measures of a node in determining its trustworthiness.

II. Literature Survey

This Literature Survey describes the various trust models referred to and implemented in the course of the study.

A. Trust between people [2]

Trust between people is known to be directly proportional to Credibility, Reliability, Intimacy and Inversely proportional to Self-Orientation. Here parallels can be drawn to establish a pattern between various trust factors. It can be concluded that Ratings (Credibility), Total interactions with other nodes (Reliability) and the Total number of interactions with the same node (Intimacy) are all directly proportional to Trustworthiness. It can also be derived that the trust of a known malicious node contributes less towards the trust score than the trust derived from a non-malicious node.

B. Eigen Trust Model [2]

The Eigen Trust model in a Cloud Computing application is used to evaluate the satisfactoriness of downloads from each peer. It states that

$$S_{(i,j)} = sat_{(i,j)} - unsat_{(i,j)} \qquad (1)$$

where $S_{(i,j)}$ is the measure of satisfaction of downloads and $sat_{(i,j)}$ is the number of satisfactory downloads and $unsat_{(i,j)}$ is the number of unsatisfactory downloads. A similar equation can be used to establish the trustworthiness of a node in a bitcoin network. The $sat_{(i,j)}$ can be used to represent the number of satisfactory transactions and $unsat_{(i,j)}$ can be used to represent the number of unsatisfactory transactions.

C. *The Goodness-Fairness Model* [3]

1. After each transaction between two nodes say u and v, each node provides feedback scores on how trustworthy and reliable the other node was.
2. This score is then used to calculate the fairness and goodness of a node based on the below formulae (Kumar et Al. [3])

$$g(u) = \frac{1}{|in(v)|} \sum_{u \in in(v)} f(u)w(u,v) \quad (2)$$

$$f(v) = 1 - \frac{1}{|out(u)|} \sum_{v \in out(u)} \frac{|w(u,v) - g(v)|}{2} \quad (3)$$

where,
$g(u)$, the Goodness of node u, indicates how trustworthy it is and $f(v)$, the fairness of node v, indicates how honest the node is in their feedback.

III. Data Set

Source	Target	Rating	Time
7188	1	10	1407470400
160	1	10	1394683200
95	1	9	1384578000
377	1	7	1414728000

Figure 1: Bitcoin OTC dataset

The dataset has been obtained from Kumar et al. (2016)[3]. Members of Bitcoin OTC rate other members in a scale of -10 (total distrust) to +10 (total trust) in steps of 1 based on their transaction experience.

The signed attribute which signifies the Trustworthiness of a node can be used to implement several networking/machine learning models.

Each line has one rating, sorted by time, with the following format: *Source, Target, Rating, Time.*
Where,
Source: node ID of source, i.e., rater
Target: node ID of target, i.e., ratee
Rating: the source's rating for the target, ranging from -10 to +10 in steps of 1
Time: the time of the rating, measured as seconds since Epoch.

IV. Data Analysis

A. *Data Preprocessing*

The time column in the dataset is converted to human readable data.

	src	dst	rating	time	timeReadable
0	6	2	4	1.289242e+09	2010-11-09 00:15:11.728360
1	6	5	2	1.289242e+09	2010-11-09 00:15:41.533780
2	1	15	1	1.289243e+09	2010-11-09 00:35:40.390490
3	4	3	7	1.289245e+09	2010-11-09 01:11:17.369750
4	13	16	8	1.289254e+09	2010-11-09 03:40:54.447460

Figure 2: Snippet of dataset after preprocessing

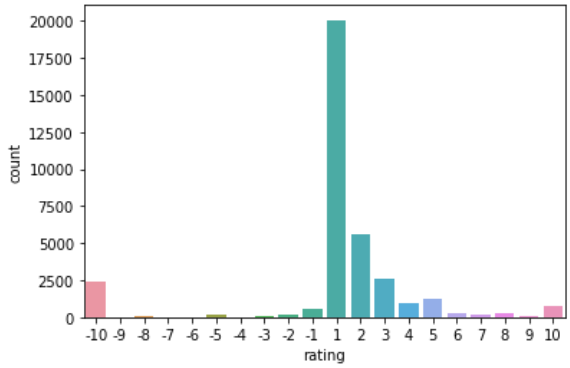
B. *Exploratory Data Analysis*

According to Figure 3 (Table 1 , Moindrot[4]), the ratings in the dataset follow the guidelines mentioned below.

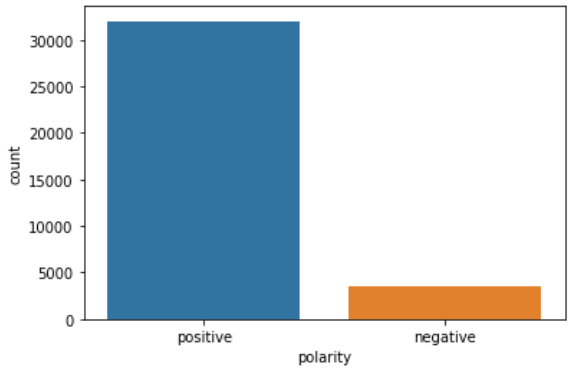
Rating	Fraction	Guideline
10	2.1%	You trust this person as you trust yourself.
9	0.30%	
8	0.78%	Large number of high-value transactions, long period of association, very trustworthy.
6, 7	1.3%	
5	3.6%	You've had a number of good transactions with this person.
2, 3, 4	25.5%	
1	56.3%	One or two good transactions with this person
-1	1.7%	Person strikes you as a bit flaky. Unreasonable/unexpected delays in payment, etc.
-2 to -9	1.5%	
-10	6.78%	Person failed to hold up his end of the bargain, took payment and ran, fraudster.

Figure 3: Rating guidelines from the OTC wiki and fraction of the overall ratings for each rating.

A countplot (Figure 4 (a)) on ratings reveals that the dataset is heavily skewed towards rating 1, which can be explained because many of the users in the bitcoin network participate in only one or two transactions with a particular node.



(a) Range of ratings



(b) Polarity of ratings

Figure 4: Count Plots on ratings

The dataset has 6000 nodes, 35592 transactions and 89% positive ratings. Using the python networkx module to visualize the network, a few interesting observations are found.

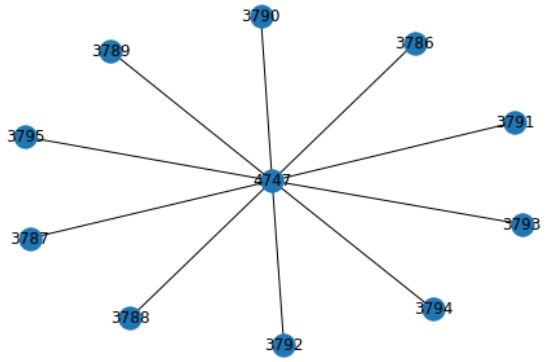


Figure 5: Cluster of nodes with -10 rating

The nodes in the above cluster all have rated each other a -10 indicating mutual Distrust.

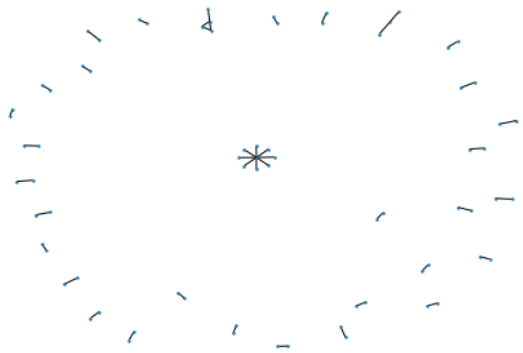


Figure 6: Clusters of nodes with +10 rating

Most of the node clusters shown in Figure 6 are disconnected but there is a cluster which rates each other a +10 in the centre. Single interaction could be a chance based incident but getting a cluster of +10 rating nodes is highly unlikely. It could be that these nodes have rated themselves positive to boost their trust scores. This suggests that the ratings by themselves aren't an accurate measure of trustworthiness. Considering other measures such as node interactions and time of transaction might provide better grounds for trust score calculations.

V. Implementation

The identified algorithms have been applied to the data set mentioned in section III.

A. Average Rating Model

The average rating model calculates the average ratings of a node, which is,

$$Avg.\ rating\ of\ node\ =\ \frac{sum\ of\ ratings\ received}{number\ of\ participated\ transactions} \quad (4)$$

The ratings indicate how trustworthy and reliable the node seemed during the transaction. To classify a node as trustworthy or untrustworthy, a threshold is chosen, which is the mean average ratings of all the nodes in the dataset.

B. EigenTrust Model

The Eigen Trust model is implemented according to the pseudocode shown in Figure 7 (Figure 3, Alexa et Al. [1]).

Basic EigenTrust Algorithm

$\vec{t}^{(0)} = \vec{p}$

repeat

$\vec{t}^{(k+1)} = (1-a)C^T\vec{t}^{(k)} + a\vec{p}$

$\delta = \|\vec{t}^{(k+1)} - \vec{t}^{(k)}\|$

until $\delta < \epsilon$

Figure 7: Eigen Trust Pseudocode

The nodes are classified as Trustworthy or Untrustworthy by using the threshold of $\frac{1}{Number\ of\ nodes}$ as prescribed by the Eigen Trust Model. Each node is initially given a rating of $\frac{1}{Number\ of\ nodes}$ and after processing, the node classification is done by comparing the final trust score with the initial trust score and the classification is stored as the trust label for the nodes.

C. Goodness and Fairness Model

The Goodness and Fairness value is calculated iteratively according to the pseudocode in Figure 8 (Figure 3, Kumar et al. [3])

Input: A WSN $G = (V, E, W)$
Output: Fairness and Goodness scores for all vertices in V
Let $f^0(u) = 1$ and $g^0(u) = 1, \forall u \in V$
$t = -1$
do
  $t = t + 1$
  $g^{t+1}(v) = \frac{1}{|in(v)|}\sum_{u \in in(v)} f^t(u) \times W(u,v), \forall v \in V$
  $f^{t+1}(u) = 1 - \frac{1}{2|out(u)|}\sum_{v \in out(u)} |W(u,v) - g^{t+1}(v)|, \forall u \in V$
while $\sum_{u \in V} |f^{t+1}(u) - f^t(u)| > \epsilon$ or $\sum_{u \in V} |g^{t+1}(u) - g^t(u)| > \epsilon$
Return $f^{t+1}(u)$ and $g^{t+1}(u), \forall u \in V$

Figure 8: Goodness-Fairness Model

The goodness and fairness values obtained are then used to classify the node as trustworthy or untrustworthy. The mean fairness value and mean goodness value of the nodes in the dataset are used as the threshold to classify the nodes.

D. Centrality Measures

The Centrality Measures of the nodes can be obtained with the help of the python networkx package. The values obtained are normalized and used in the algorithms implemented in the next section.

E. Standard Machine Learning Algorithms

The standard Machine Learning Algorithms for classification such as Logistic Regression, KNN, Decision Tree, Agglomerative Clustering and Random Forest Ensemble Classification have been implemented on the Centrality Measures to observe the effect of Centrality Measures on the trustworthiness.

VI. Observations

The models implemented require a set of True Values that can be used to calculate the model efficiency for performance evaluation. Due to the wide acceptance of the Eigen Trust Model as a standard Trust Model in networks, and considering that bitcoin is primarily a network of nodes, the trust labels of the Eigen Model have been used as True Values for this section.

The relative performance metric observations made from all the model implementations are shown in Figure 9.

```
results
```

	cf matrix	accuracy	precision	recall	f1	auc
gf	[[20, 68], [74, 238]]	0.6450	0.212766	0.227273	0.649128	0.495047
agglomerative	[[53, 35], [192, 120]]	0.4325	0.216327	0.602273	0.470887	0.493444
avg rating	[[88, 0], [308, 4]]	0.2300	0.222222	1.000000	0.363636	0.506410
decision tree	[[20, 1], [6, 53]]	0.9125	0.769231	0.952381	0.851064	0.925343
knn	[[8, 3], [3, 66]]	0.9250	0.727273	0.727273	0.925000	0.841897
logistic	[[2, 19], [0, 59]]	0.7625	1.000000	0.095238	0.680871	0.547619
random forest	[[21, 0], [5, 54]]	0.9375	0.807692	1.000000	0.939442	0.957627

Figure 9: Performance matrix of implemented algorithms

From the performance matrix (Figure 9), we find that the goodness-fairness model is unable to predict the node trustworthiness with good accuracy and F1 score. The average rating model has very poor accuracy, proving our observation that ratings alone are not a good metric for a node's trustworthiness. The performance of the Machine Learning models applied over the Centrality Measures is significantly better than the goodness-fairness model and the simple average rating method. Of all the models applied over the Centrality Measures, the Random Forest Model shows the best performance with very high accuracy and F1 Score.

VII. Conclusion

From the analysis of the dataset, it has been found that the peer ratings by themselves aren't a good measure of the trustworthiness of a node. This calls for the need of a better model to classify nodes as trustworthy and untrustworthy.

The goodness-fairness model nor the average rating model predict the trust labels accurately.

The Random Forest model based on the centrality measures predicts the trust labels with good accuracy and F1 score despite not using the peer rating. As the Eigen Trust Model only considers the number of trustworthy and untrustworthy transactions, this shows that the Centrality Measures are critical factors in determining the trustworthiness of a node.

The results show that Centrality Measures are critical factors in determining a node's trust. It suggests that including Centrality Measures in a bitcoin trust model could result in a better performance.

VIII. Future Direction

The observations and conclusions obtained from the implementation are specific to the bitcoin dataset used. Further research into the importance of Centrality Measures in determining node trustworthiness should be conducted in various bitcoin and other cryptocurrency peer-rated datasets to establish the actual effect of Centrality Measures on trustworthiness.

From the observations, it is also suggested to consider Centrality Measures as important factors affecting trust when a trust algorithm is to be built.

Acknowledgment

References

[1] Alexa, A., & Theobald, A. "Reputation Management in P2P Networks : The EigenTrust Algorithm", Corpus ID: 17694602

[2] Firdhous, Mohamed & Ghazali, Osman & Hassan, Suhaidi.. "Trust Management in Cloud Computing: A Critical Review", *International Journal on Advances in ICT for Emerging Regions (ICTer)*, vol. 4, 2012, doi: 10.4038/icter.v4i2.4674.

[3] S. Kumar, F. Spezzano, V. S. Subrahmanian and C. Faloutsos, "Edge Weight Prediction in Weighted Signed Networks," *2016 IEEE 16th International Conference on Data Mining (ICDM)*, 2016, pp. 221-230, doi: 10.1109/ICDM.2016.0033.

[4] Moindrot, O. (2017). "Trust in bitcoin exchange networks." *Project for cs224w-2017, Stanford University*.