Detailed analysis of Ethereum network on transaction behavior, community structure and link prediction

Anwar Said[1], Muhammad Umar Janjua[1], Saeed-Ul Hassan[2], Zeeshan Muzammal[1], Tania Saleem[1], Tipajin Thaipisutikul[3], Suppawong Tuarob[3] and Raheel Nawaz[4]

[1] Department of Computer Science, Information Technology University, Lahore, Pakistan

[2] Department of Computing and Mathematics, The Manchester Metropolitan University, Manchester, United Kingdom

[3] Faculty of Information and Communication Technology, Mahidol University, Salaya, Nakhon Pathom, Thailand

[4] Department of Operations, Technology, Events and Hospitality Management, Manchester Metropolitan University, Manchester, United Kingdom

ABSTRACT

Ethereum, the second-largest cryptocurrency after Bitcoin, has attracted wide attention in the last few years and accumulated significant transaction records. However, the underlying Ethereum network structure is still relatively unexplored. Also, very few attempts have been made to perform link predictability on the Ethereum transactions network. This paper presents a Detailed Analysis of the Ethereum Network on Transaction Behavior, Community Structure, and Link Prediction (DANET) framework to investigate various valuable aspects of the Ethereum network. Specifically, we explore the change in wealth distribution and accumulation on Ethereum Featured Transactional Network (EFTN) and further study its community structure. We further hunt for a suitable link predictability model on EFTN by employing state-of-the-art Variational Graph Auto-Encoders. The link prediction experimental results demonstrate the superiority of outstanding prediction accuracy on Ethereum networks. Moreover, the statistic usages of the Ethereum network are visualized and summarized through the experiments allowing us to formulate conjectures on the current use of this technology and future development.

Subjects Data Mining and Machine Learning, Data Science, Emerging Technologies

Keywords Ethereum, Graph Neural Network, Wealth Distribution, Network Community Structure

INTRODUCTION

Networks are ubiquitous data structures representing complex real-world scenarios that generally involve relationships among objects (*Hamilton, 2020*). Blockchain is one of the promising networks that have the potential to reform several conventional businesses. The first generation of blockchain, namely Bitcoin, has demonstrated that the global consensus can be completed without a trusted third party or central authority. As a result, many researchers have put a lot of effort into designing more powerful and

Submitted 25 August 2021
Accepted 23 November 2021
Published 10 December 2021

Corresponding author
Suppawong Tuarob,
suppawong.tua@mahidol.edu

Academic editor
Leandros Maglaras

Additional Information and Declarations can be found on page 21

DOI **10.7717/peerj-cs.815**

How to cite this article Said A, Janjua MU, Hassan S-U, Muzammal Z, Saleem T, Thaipisutikul T, Tuarob S, Nawaz R. 2021. Detailed analysis of Ethereum network on transaction behavior, community structure and link prediction. *PeerJ Comput. Sci.* **7**:e815 http://doi.org/10.7717/peerj-cs.815

multifunctional blockchain systems due to their high applications in numerous real-world settings.

Later, Ethereum (a system of a transaction-based state machine and a fully decentralized peer-to-peer) was developed in 2015 and became the second-largest blockchain platform, where the market value reached over 1,000 million dollars in 2020 (*Nakamoto, 2019*; *Wood, 2014*; *Ma et al., 2021*; *Akhtar et al., 2021*). After the development of Ethereum, it has been successfully used in a variety of applications, including transaction management, smart contracts, and industrial applications. Since Ethereum's growth in value and adoption in the market, critical enterprise applications based on programming frameworks, and the total number of users is increasing, the research community's attention is now focused on investigating and analyzing various aspects of the Ethereum system (*Wu et al., 2019*).

Although various statistical analyses on blockchain transactional networks have been performed, most of these methods focus on deanonymization (*Androulaki et al., 2013*; *Ober, Katzenbeisser & Hamacher, 2013*; *Said et al., 2019*), clustering (*Meiklejohn et al., 2013*; *Said et al., 2018*), and finding malicious activities (*Hirshman, Huang & Macke, 2013*; *Harlev et al., 2018*; *Möser, Böhme & Breuker, 2013*; *Ao et al., 2021*; *Rodriguez-Garcia, Sicilia & Dodero, 2021*) of Bitcoin system. However, such Bitcoin data analysis cannot be applied or performed directly on the Ethereum data because of the different protocols and designs.

Ethereum users' activities are encapsulated in the blocks as shown in Fig. 1 where each transaction inside a block includes the sending and receiving addresses and the transferred value. As an open shared ledger, Ethereum allows any user to store the history of the entire transaction. By using this history, special nodes (miner's node) can confirm new transactions. Miner's integrity is determined by a proof mechanism that validates miners' transactions. It notifies new transactions added to the Ethereum chain *via* blocks added at a constant rate between 10 and 20 s (*Gervais et al., 2016*).

Ethereum is difficult to calculate when changing a transaction (double spending) (*Rosenfeld, 2014*) that a user has already used since the processing information for all relevant blocks must be re-executed. All users of the Ethereum network receive and send transactions through ID or address generated by the Elliptic Curve Digital Signature Algorithm (ECDSA), which gives the private and public key pairs. The private key is used to send transactions to another address, and the public key is used to receive transactions from another address. Ethereum users can synchronize the nodes with the network to get information about every transaction. A transaction includes sender address, recipient address, amount (Ether), time, and other attributes as shown in Table 1. However, for security and anonymity, a user's real identity is not tied to an address, making analysis difficult.

Existing studies on Ethereum focus on the analysis of the transactional Ethereum data in terms of quantity, network in-degree, and out-degree distributions. For example, *Muzammal et al. (2019)* deployed the Decision Tree algorithm to predict future transactions by utilizing two features: "from" and "to", which demonstrated the capability of using the network theory to analyze the Ethereum transactional network. However,

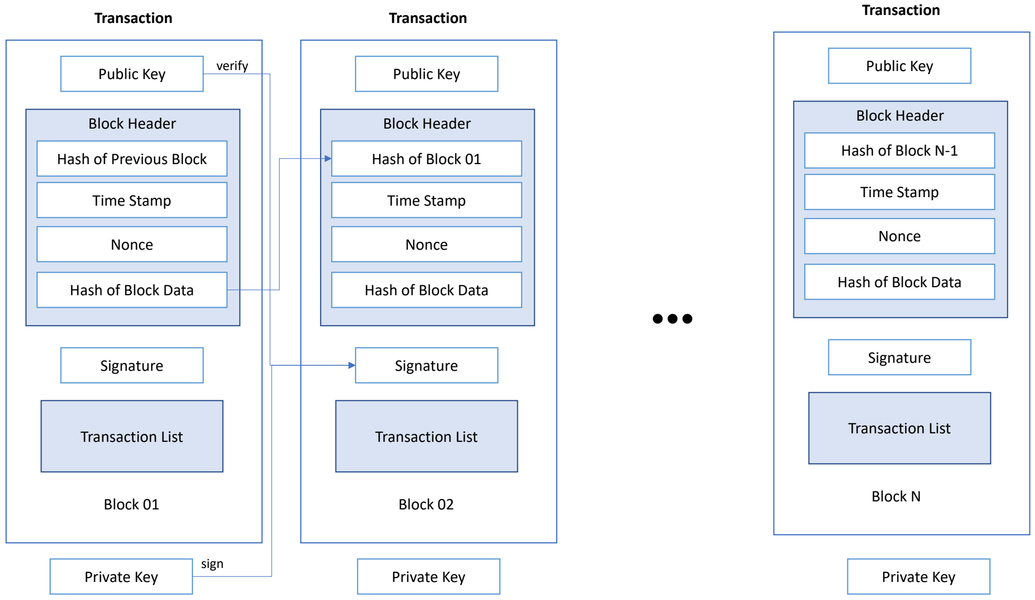


Figure 1 **The structure and transaction of Ethereum blockchain.**

Full-size DOI: 10.7717/peerjcs.815/fig-1

most studies in this area still overlooked detailed analyses of the network community structures. While extensive studies have been performed on blockchain networks such as Bitcoin (*Nerurkar et al., 2021*) due to its long establishment, network analyses on Ethereum are quite limited (*Li et al., 2020*). Such analyses could play a crucial role in wealth distribution, the network's relational structure, and the link predictability from heterogeneous network data.

This paper presents a sequence of studies on the Ethereum network, including detecting community structures and investigating link predictability on the transaction network using a graph structure learning technique. Specifically, we propose a **D**etailed **A**nalysis of Ethereum **Net**work on Transaction Behavior, Community Structure, and Link Prediction framework, namely DANET, as a unified platform to conduct various analyses simultaneously. Specifically, DANET consists of four main modules: (1) Ethereum Data Management; (2) Ethereum Transaction Behavior Analysis; (3) Ethereum Community Structure Analysis; and (4) Ethereum Link Prediction Analysis. In particular, Ethereum Data Management is designed to collect and filter the transactional data used in the experiments. At the same time, Ethereum transaction behavior analysis and Ethereum community structure analysis are proposed to better understand the network's characteristics, such as in-degree and out-degree relationships. Also, Ethereum Link Prediction Analysis is introduced to perform the graph construction and representation for the link prediction. The experimental results show some useful statistical characteristics of the Ethereum network in terms of the distribution of active addresses, traffic of Ether history per address, and the degree distribution. Also, we could achieve high accuracy from 80–90% on the link prediction task given the time-series snapshot graph as inputs.

The main contributions of this manuscript are as follows:

Table 1 Block and transactions' attributes of the Ethereum data.

Attribute	Description
	Block Information
name	A unique block identifier
nonce	A hash of proof-of-work
hash	A unique hash of the block
miner	A beneficiary address who receives mining reward
total Difficulty	Indicating the total difficulty of the chain up to a specified block by an integer value
difficulty	Specifying the difficulty level by an integer value
extraData	A field containing additional data from a block
size	The block size in bytes
gasUsed	Total gas used by all transactions in a block
gasLimit	Maximum gas usage of all transactions in a block
timestamp	A UNIX timestamp when blocks were contrasted
transactions	Unique ID of the transaction or a hash array of 32-byte transactions
uncles	Uncle block hashes array
	Transactional Information
nonce	Before that transaction, total transactions made by similar sender
hash	A unique transaction hash
blockNumber	A unique block number for the committed transaction block
blockHash	A unique hash for the committed transaction block
from	A unique hash string considered as sender's address
to	A unique hash string considered as receiver's address, resulted null if creating contract is the purpose of received transaction
value	The transferred amount in (Wei) where Wei is unit of Ethereum
gasPrice	Sender provided gas proice in (Wei)
gas	Sender provided gas amount
input	Extra data sent with the transaction

- We propose DANET: A **D**etailed **A**nalysis of Ethereum **Net**work on Transaction Behavior, Community Structure and Link Prediction framework as a unified framework to return various aspects of analysis to support the understanding of the Ethereum network.
- We study the matter of Ethereum transaction tracking from a network perspective (*i.e.*, the influential addresses and community structure) which gives a deeper understanding of Ethereum transaction records and could contribute to the long-term evolution of the blockchain.

- We model Ethereum transactional data in the form of a heterogenous attributed network that preserves all the transactions' essential information with graph auto-encoders for Ethereum link prediction.
- We make the code and dataset available for research purposes at github.com/Anwar-Said/Link-Predictability-using-VGAE.

The rest of the paper is organized as follows. 'Introduction' outlines the Ethereum data analysis and network-based representation approaches. 'Related Work' discusses background and relevant literature. 'The Proposed Framework: DANET' presents the methodology used in this research. 'Experiment Results' provides the experimental results and relevant discussions. Finally, 'Conclusions' concludes the paper.

RELATED WORK

This section presents an overview of the recent advancements in Ethereum, Bitcoin, and Network representation, mainly divided into Ethereum data analysis and network representation. The first category of approaches involves studying Ethereum and Bitcoin data using different techniques, while the latter deals with learning network structures using deep learning (DL) based graph representation approaches.

Ethereum data analysis

Recently, many methods have been proposed to explore the Bitcoin network. *Gencer et al. (2018)* analyzed the number of Bitcoin users having large balances and studied graph-based Union-Find algorithm for finding addresses matching best to individuals. The authors also studied whether Bitcoin is primarily used for saving or routine transactions. *Karame, Androulaki & Capkun (2012)* presented a scenario for spending and avoiding double payments in Bitcoin transactions, by calculating the average "standard deviation" time, "transaction acceptance" time and "block generation" time of the network.

Similarly, *Chan & Olmsted (2017)* used a transaction-based graph that was configured on each node to analyze the behavior of each address. They also clustered the nodes using the similarity of the graph. The study concluded that Ethereum's new transaction input is independent of the output of past unspent transactions, unlike Bitcoin. *Gencer et al. (2018)* analyzed the distribution statistics of various blockchains by mining power distribution. The results have shown that 61% of the weekly mining power was shared by only three IDs, with 90% of the power being shared by 11 entities. Mining nodes' integrity was also evaluated by calculating the block numbers in the node that resulted in blocks of ankles(blocks that most miners rejected). *Koshy, Koshy & McDaniel (2014)* found that Bitcoin clients are designed for data collection where clients actively connect with their peers and collect all broadcast data along with IP information. The authors analyzed Bitcoin traffic, looked for anomalous relay patterns, and mapped Bitcoin addresses to IPs using the collected data. Moreover, anonymity links in the Bitcoin network were discovered using the aggregation method proposed by *Reid & Harrigan (2011)*. The aggregation method associates different bitcoin addresses with users by specifying multiple inputs, multiple outputs, regular transactions, and geographically co-located IP addresses within a period. By splitting the shared Bitcoin wallet into different units, *Meiklejohn et al. (2013)* worked on the identification of identities in the executed transaction chunk by introducing intelligent clustering. By using heuristics of participating payments and address changes, authors who identified approximately 3.4 million clusters were able to put nearly 2,000 names from them. Additionally, *Ober, Katzenbeisser & Hamacher (2013)* suggested a structural analysis technique for the prediction of graph anonymity of the Bitcoin transactions. The author used a global passive adversary that defines entities according to the linkability of a transaction. Global enemies were also using participatory payment and address to change reasoning.

After Bitcoin, Ethereum is perhaps the second most popular cryptocurrency-based network; both employ blockchain, a distributed ledger technology. Both Bitcoin and Ethereum are digital currencies; however, the fundamental aim of Ether (Ethereum transactional token) is to facilitate and monetize the operation of the smart contract and decentralized application platform, rather than establish itself as an alternative monetary system. While Bitcoin networks have been extensively investigated and analyzed in the previous literature, the recent emergence of Ethereum in 2015 has merely drawn attention from limited research, making it scarcely explored (*Li et al., 2020*). Some of the recent studies that are relevant to the Ethereum data analysis is discussed here.

Maeng, Essaid & Ju (2020) proposed a node discovery algorithm for the Ethereum network utilizing the P2P links discovery. Furthermore, they analyzed the collected Ethereum data to identify the relationship between nodes, heavily connected nodes, and nodes geo-distribution. *Farrugia, Ellul & Azzopardi (2020)* proposed an XGBoost based classification algorithm for detecting the illicit accounts on the Ethereum network. Their dataset comprised 2,179 illicit accounts flagged by the Ethereum community and 2,502 normal accounts. They have identified that top features associated with illicit activities include 'Time diff between first and last(Mins)', 'Total Ether balance', and 'Min value received'. *Li et al. (2020)* highlighted that all cryptocurrency and crypto-token transactions are permanently recorded on distributed ledgers and are publicly accessible, allowing for the development of a transaction graph and the analysis of connections between transaction graph characteristics and crypto price dynamics. They used the principles of persistent homology and functional data depth to study Ethereum crypto-tokens, particularly investigating price anomaly predictions and hidden co-movement between tokens. Using topological data analysis and functional data depth into blockchain data analytics, they discovered that the Ethereum network could provide valuable insights on price changes of crypto-tokens that are otherwise largely inaccessible with conventional data sources and traditional analytic methods. *Xie et al. (2021)* proposed to model Ethereum transaction records with a time-series snapshot network (TSSN) that captures the transactions' spatial and temporal aspects. The network was traversed using the temporal biased walk (TBW) algorithm that effectively embeds accounts *via* their transaction records. They further explored two problems: phishing node classification and link prediction using a number of graph embedding algorithms. This study, however, lacks the analysis of the global Ethereum transaction network. Closest to our research would be the study by *Wu et al. (2021)* where the community detection problem was examined in both the Bitcoin and

Ethereum networks. The low-rank community detection algorithm proposed by *Wai et al. (2018)* was used to detect communities in the Ethereum network. However, their study represented the Ethereum network as a graph of EoAs (users) and CAs (contracts) nodes since their objective was to identify sub-communities. Our research, on the other hand, also considers the Ethereum transactions as well.

Network representation and link prediction

Learning network structure has received considerable attention in the last few years due to its wide range of applications, including recommender systems, molecular structures, biological systems, and various physical systems (*Cai, Zheng & Chang, 2018*). Since the network structure is unordered, classical machine learning and DL approaches are not directly applicable. The DL application on graphs was first presented by *Scarselli et al. (2009)* where Graph Neural Networks (GNNs) was proposed. This idea was later refined and extended by *Gallicchio & Micheli (2010)* and *Kipf & Welling (2016a)*. GNNs methods generally involve several DL de facto standards such as random walks over networks, convolutions, recurrent neural networks, adversarial networks, message passing and autoencoders (*Cai, Zheng & Chang, 2018*; *Hamilton, Ying & Leskovec, 2017*; *Zhang, Cui & Zhu, 2020*; *Said et al., 2020*). These methods work in several settings in both supervised and unsupervised fashions. Various tasks can be performed over networks using these approaches, such as graph classification, node classification and link prediction (*Bojchevski & Günnemann, 2018*; *Kipf & Welling, 2016b*; *Ahmed, Hassan & Shabbir, 2020*). In the Ethereum network perspective, link predictability defines the ability to identify future transactions between two addresses. In other words, link prediction is a problem of identifying potential or missing links in a network.

From a network perspective, the link prediction task is a widely studied problem where its approaches consist of three categories: heuristics methods, graph embedding methods, and feature learning methods. The heuristics methods usually compute node similarities using graph-theoretic methods and use them as a likelihood of links (*Zhang et al., 2020*). Among which preferential attachment (*Barabási & Albert, 1999*), Jaccard coefficient (*Liben-Nowell & Kleinberg, 2007*), and Katz index (*Katz, 1953*) are well-known methods. Graph embedding methods involve learning free-parameter node embeddings based on the predefined network in a transductive setting where they cannot be generalized on unseen nodes (*Grover & Leskovec, 2016*; *Hamilton, 2020*). The third category involves the powerful and recently emerged Graph Neural Networks (GNNs) methods which learn node features using message passing mechanism and generalize well on unseen nodes (*Kipf & Welling, 2016a*; *Kipf & Welling, 2016b*; *Said et al., 2021*; *Bojchevski & Günnemann, 2018*; *Hamilton, Ying & Leskovec, 2017*). In a supervised setting, Graph Auto Encoders (VGAE) is largely adopted specifically for link prediction (*Kipf & Welling, 2016b*). In link prediction, VGAE learns node embeddings in an unsupervised fashion with a negative sampling approach (*Yu et al., 2018*). *Kipf & Welling (2016b)* introduced an unsupervised framework for learning graph-structured data with variational auto-encoders and latent variables. These methods have shown promising results and are now considered to be

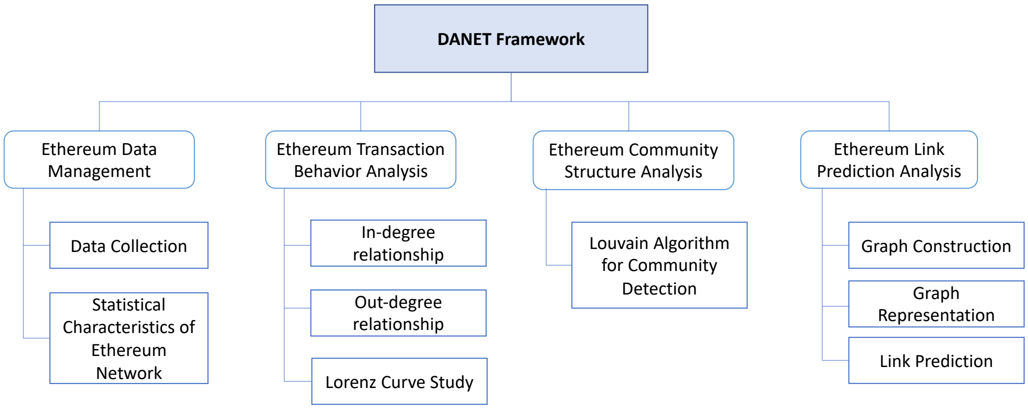

DANET Framework
Ethereum Data Management
Data Collection
Statistical Characteristics of Ethereum Network
Ethereum Transaction Behavior Analysis
In-degree relationship
Out-degree relationship
Lorenz Curve Study
Ethereum Community Structure Analysis
Louvain Algorithm for Community Detection
Ethereum Link Prediction Analysis
Graph Construction
Graph Representation
Link Prediction

Figure 2 **The proposed DANET framework architecture.**

Full-size DOI: 10.7717/peerjcs.815/fig-2

powerful tools for learning the graph-structured data (*Zhang, Cui & Zhu, 2020*; *Said et al., 2020*).

Unlike the existing works, we propose the framework named DANET to provide the Detailed Analysis of Ethereum Network on Transaction Behavior, Community Structure, and Link Prediction framework as a unified platform. Particularly, we adopt a unique approach to represent Ethereum data in the network form in the graph structure, allowing us to observe several exciting properties of the Ethereum network. We also considered the link predictability task on the constructed network and deployed VGAE (*Kipf & Welling, 2016b*), a powerful GNNs based learning model that yields outstanding link prediction results. We show that the Ethereum network consists of an exciting community structure, following the phenomenon of real-world networks.

THE PROPOSED FRAMEWORK: DANET

As shown in Fig. 2, to comprehensively analyze the Ethereum network and transaction records, we propose a consolidated framework: DANET, which includes four main modules to deliver the different analysis results. (1) Ethereum Data Management: to collect Ethereum transactional data for the experiments and compute the statistical characteristics of the Ethereum network; (2) Ethereum Transaction Behavior Analysis: to investigate the transaction behavior such as in- and out-degree relationships; (3) Ethereum Community Structure Analysis: to identify the trait of Ethereum community structure; (4) Ethereum Link Prediction Analysis: to evaluate the effectiveness of our framework on the Ethereum link prediction task. The details of each component are elaborated in the following subsections.

Ethereum data collection

For data collection, we synced the Ethereum full node to collect all the historical transactional data. We used a spark cluster with one master node and two worker nodes with Ubuntu 16.04 having 40 GB RAM on each machine and an Internet connection of 10Mbps. We used geth (https://geth.ethereum.org/). Ethereum client as a full node to collect all the historical blocks data. This node took 11 days to collect data till 2018. We used the web3 API to send RPC requests to the Ethereum node. Web3 is an Ethereum compatible JavaScript API that implements the general JSON RPC specification. JSON-RPC is a transport-agnostic protocol that can be used over sockets and HTTP. We defined the RPC port and address while configuring the Ethereum node. We used $web3.eth.getBlock(id, true)$ to retrieve blocks and extract transaction information from each block, and save the extracted information to a PostgreSQL database. The total collected Ethereum transactions data was from "2015-08-07″to "2019-01-01″comprising 189 million transactions in 55 blocks.

Ethereum transaction behavior analysis

We used the Ethereum transactions dataset used by *Muzammal et al. (2019)*. The dataset is 200 MB in size and was first downloaded and processed for understanding the Ethereum network. The raw data can be downloaded from Google BigQuery (https://tinyurl.com/7bmh3xkf). We constructed a directed Ethereum Transaction Featured Network (ETFN) where vertices represent addresses and edges represent the relationship in terms of transaction among the vertices. We also use the number of transactions among the pair of addresses (nonce), and the transferred amount (value) as a feature set over each edge to preserve transaction information. Formally, our ETFN is a attributed directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V} = \{v_1, v_2, v_3, \ldots, v_n\}$ and $\mathcal{E} = \{e_1, e_2, e_3, \ldots, e_m\}$ where $n = |V|$, $m = |E|$. Also, we define $e = (u, v, w)$ where u and v represent two nodes in $\mathcal{V}$, and w represents the weight of the edge between these two nodes .

Ethereum community structure analysis

Exploring the community structure of a network plays a vital role in understanding the underlying network structure. There is no universal definition of a community within a network. However, it is widely accepted that the community represents a sub-group of vertices that are densely intra-connected and sparsely interconnected with the rest of the network (*Said et al., 2018*). A community represents a set of individuals with common interests within a network. For example, in a protein-protein interaction network, proteins having common functionality may belong to the same community. A community may represent a particular region of the brain having dense neurons connectivity in a brain network. Similarly, in a transaction network, a community represents individuals who frequently make transactions with each other. Exploring a transaction network's communities can reveal individuals' potential and valuable information regarding their transaction patterns and time slots (if the network is time-variant) (*Newman & Girvan, 2004*; *Newman, 2006*; *Said et al., 2019*).

Due to numerous applications in a wide range of real-world settings, community detection has caught the research community's special attention, especially the Louvain community detection algorithm (*Blondel et al., 2008*). The Louvain algorithm is a greedy method based on the optimization of the modularity measure that has been extensively used to identify communities in crypto-currency networks, such as that of Bitcoin (*Remy, Rym & Matthieu, 2017*; *Zhang, Wang & Zhao, 2020*; *Gavin & Crane, 2021*). While the

Bitcoin network has some differences from the Ethereum network, it makes sense to follow similar protocols widely used to analyze these cryptocurrency networks. The Louvain algorithm is a greedy method based on optimization of the modularity measure, which can be defined for a simple undirected network as follows.

$$\mathcal{Q} = \sum_{c=1}^{k} \left[\frac{\mathcal{E}_c}{\mathcal{E}} - \left(\frac{\deg_c}{2\mathcal{E}} \right)^2 \right] \tag{1}$$

In the above equation, k is the total number of communities, $\mathcal{E}_c$ is the total number of edges, $\deg_c$ indicates the total degree in the community c, and $\mathcal{E}$ is the total edges in the network. The modularity value ranges between [-1,1], where the highest value indicates a good community structure and vice versa. The negative value means no community structure in the network. The value approaches zero if all the vertices are assigned to a single community (*Newman, 2006*).

The Louvain algorithm optimizes the modularity value of the network and consists of two phases. The first phase assigns a different community to each node and then attractively combines each node to its neighbors' community and evaluates the modularity score. In case of improvements in the modularity score, nodes are merged into a single community. This process is repeated until there is no gain in the modularity score. In the second phase, the first phase communities are compressed to a single node where the internal edges are used as self-links and repeat the first step. Once no further improvements are found, the algorithm stops and returns the identified community structure. Louvain community detection algorithm is known to be one of the scalable algorithms having $O(n\log n)$ where n is the number of nodes (*Blondel et al., 2008*).

Ethereum link prediction analysis

Graph construction

To perform link predictability on our Ethereum Transaction Featured Network (EFTN), we employ the Variational Graph Auto-encoder (VGAE) (*Kipf & Welling, 2016b*) as our primary model. Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with $N = |\mathcal{V}|$ vertices, let $A \in \{0,1\}^{N\times N}$ denote the adjacency matrix of $\mathcal{G}$ where $A_{ij} = 1$ if v_i and v_j are neighbors and 0 otherwise. Let $\mathbf{D}^{N\times N}$ denote the degree matrix of $\mathcal{G}$.$\mathbf{D}$ is a diagonal matrix where its diagonal values $D_{i,i}$ equals the degree of v_i. Similarly, let $A\mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}$ be the normalized adjacency matrix. Let N_i denote the network neighborhood of a vertex v_i, $X^{N\times d}$ represents node features matrix and z_i is a stochastic latent variable summarized in an $N \times d$ matrix Z. Note that N_i can be either complete v_i's neighborhood or it can be generated through a neighborhood sampling strategy $\mathcal{S}$, where the sampling strategy is a technique to randomly select a subset of vertices or edges from the original graph. The network embedding is a function $\phi : V \rightarrow \mathcal{R}^d$ that maps the vertices to a feature representation. Here d indicates the dimension of our feature presentation for each vertex. Therefore, ϕ is a matrix of size $N \times d$ parameters.

Graph representation

Variational Graph Auto-Encoders (VGAE) is a GCN-based link prediction method over networks. The algorithm has recently been adopted to learn graph representation of the

Bitcoin network (*Shah et al., 2021*; *Zhang et al., 2021*). VGAE's framework first learns vertex embeddings of the entire network using GCNs, and then the aggregation of source and target nodes is performed to predict the target link (*Kipf & Welling, 2016b*). The method uses the standard notion of variational auto-encoders while learning $\mu$and σ to generate the desired output. The architecture includes two layers of GCNs where the first layer generates the latent variables **Z** and the second layer generates $\mu$and σ. Then the standard parameterization trick is used to calculate **Z**. Given the input **A** and **X**, the first layer of GCN is defined as follows.

$$X = GCN(\mathbf{X}, \mathbf{A}) = ReLu(\hat{\mathbf{A}}\mathbf{X}\mathbf{W}_0). \tag{2}$$

The second layer of GCN generates $\mu$and σ from $\hat{\mathrm{X}}$ as follows.

$$\mu = GCN_{\mu}(\mathbf{X}, \mathbf{A}) = \hat{\mathrm{A}}X\mathbf{W}_1 \tag{3}$$

$$\sigma = GCN_{\sigma}(\mathbf{X}, \mathbf{A}) = \hat{\mathrm{A}}X\mathbf{W}_1 \tag{4}$$

where $\mathbf{W}_0$ and $\mathbf{W}_1$ are the model weight matrices. Each element $\mathbf{W}_{i,j}$ in $\mathbf{W}_0$ and $\mathbf{W}_1$ represents the weight of the edge between the ith vertex and the jth vertex.

The decoder model is simply $\hat{\mathrm{A}} = \sigma(zz^{\intercal})$, where $\sigma(.)$ is a logistic sigmoid function. The overall encoder–decoder model is defined as follows.

$$q(z_i|\mathbf{X}, \mathbf{A}) = N(z_i|\mu, \mathrm{diag}(\sigma)^2) \tag{5}$$

and the decoder is represented as

$$p(\mathbf{A}_{ij} = 1|z_i, z_j) = \sigma(z_i^{\intercal} z_j). \tag{6}$$

The loss function of VGAE is similar to the standard variational auto-encoders and is defined below.

$$\mathcal{L} = \mathbb{E}_{q(\mathbf{Z}|\mathbf{X},\mathbf{A})}\left[\log p(\mathbf{A}|\mathbf{Z})\right] - \mathrm{KL}\left[q(\mathbf{Z}|\mathbf{X}, \mathbf{A})||p(\mathbf{Z})\right]. \tag{7}$$

The first part is the reconstruction loss between the original and the constructed adjacency matrix, while the second part is the KL divergence for $p(\mathbf{Z}) = N(0, 1)$.

Link prediction

This section describes the experimental setup and results for the link predictability task on our Ethereum network. Recall that our EFTN network consists of 2.7 million vertices and 4.6 million edges. Also, the network is attributed where it contains nonce and value as features on each edge. For nodes' features, we used one-hot degree encoding; however, we fixed the size of the feature vector to 100.

We observed that few nodes (less than 10) had large degrees, playing the role of hubs in the network. Thus, to avoid sparsity in our feature matrix, we fixed the size and assigned a degree of 100 if a node's degree is greater than 100. Due to the memory limitation, we constructed two different networks while choosing a chunk from the whole data. We only considered 20 days of transactions: from 2016-12-1 to 2016:12:20 where the total number

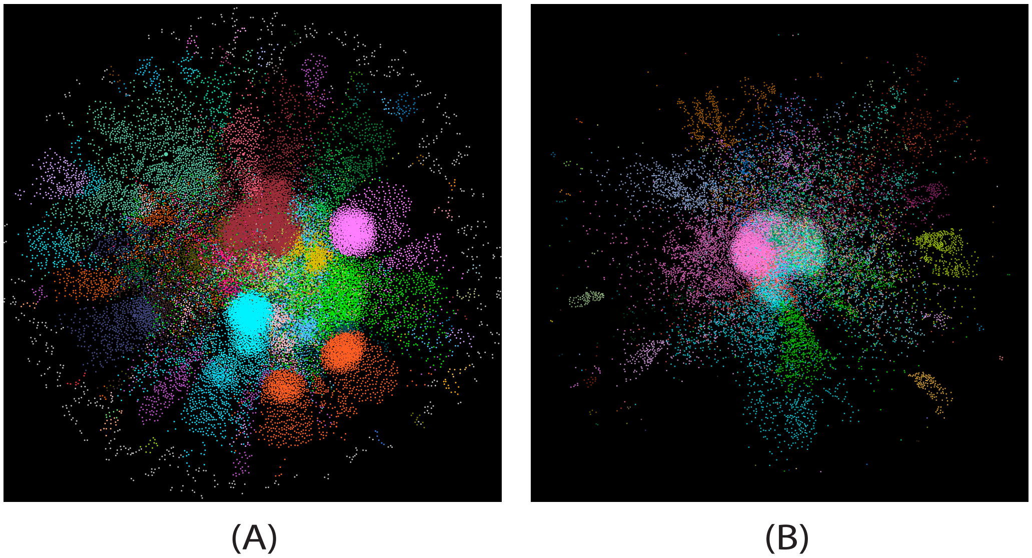


Figure 3 The visualizations of $\mathcal{G}_1$ (A) and $\mathcal{G}_2$ (B) EFTN networks.
Full-size DOI: 10.7717/peerjcs.815/fig-3

of records was 0.42 million. The first 15 days comprise around 0.210 million transactions, while the remaining five days have 0.211 million transactions. We constructed two networks $\mathcal{G}_1$ and $\mathcal{G}_2$ separately from this data. The numbers of nodes and edges in $\mathcal{G}_1$ were 33,989 and 53,261, respectively, while there were 37,175 nodes and 56,987 edges in $\mathcal{G}_2$. Please note that we consider the chunk from the data randomly; however, we believe that the slice of data at any point can be considered and would produce similar results. Also, we consider both the networks as undirected, as we want to predict a transaction among two addresses made from either side. We show the visualizations of both the constructed networks in Fig. 3.

We considered a two-layer network in GCN architecture and considered 100 and 8 neurons in the encoder layer. As mentioned previously, our decoder layer is simply the dot product of the learned feature vectors of the corresponding vertices. We used negative sampling for preparing the training and test data (*Mikolov et al., 2013*). The ratio of the train and test splits was set to $67:33$ accordingly. We set the number of epochs to 100, and the learning rate to 0.001.

EXPERIMENT RESULTS

In this section, we provide a complete set of analyses based on the Ethereum network as follows.

Statistical characteristics of ethereum network

As shown in Table 2A, we can notice that the majority of addresses (88%) are associated with less than 10 transactions each. Also, 39 addresses are frequently used on the network and are associated with at least $50,000$ transactions. On the other hand, 32% of addresses participate in a transaction only once. There are also six active addresses participating in over $1,000,000$ (30%) transactions. We investigate the six active addresses: ENS-Registrar, Table 2 (A): The distribution of active addresses. Min and max represent the minimum and maximum of the transactions. (B): The breakdown of per address transactions.

min	max	#addresses
1	2	1,115,238
2	4	1,509,244
4	10	1,102,949
10	100	364,406
100	1,000	47,711
1,000	5,000	3,307
5000	10,000	219
10,000	50,000	236
50,000	100,000	39
100,000	500,000	40
500,000	1,000,000	8
1,000,000		6

min	max	#addresses
1	2	1,700,413
2	4	1,066,002
4	10	320,416
10	100	194,338
100	1,000	33,090
1,000	5,000	1,546
5000	10,000	114
10,000	50,000	127
50,000	100,000	19
100,000	500,000	21
500,000	1,000,000	3
1,000,000		2

YoCoin, Bittrex_2, Acronis_Contract, Poloniex_1, and Kraken_5 and found them to be contract addresses.

Similarly, as shown in Table 2B, $1,700,413$ (49%) support transactions were received only once in history, concluding that most wanted to remain anonymous as they changed their addresses after each transaction. Considering the distribution of total transferred transactions per address (Table 3), we noticed that less than 10 transactions were received from $156,304$ (90%) addresses. The study found that the total number of Ethers received from most addresses was barely significant.

Table 4 shows that 28% of addresses send less than one accumulated Ether in a transaction. In its history, 48% of addresses send less than 10 Ether, and 63% of addresses receive less than 100 Ether.

Table 5 shows that 1 or less Ether was received by 32% of all addresses $(1,088,717)$, less than 10 Ether were received by 58% of addresses, and less than 100 Ether received by 75% of addresses.

Table 3 Breakdown of total transactions sent per address.

min	max	#addresses
1	2	1,319,452
2	4	984,028
4	10	419,211
10	100	156,304
100	1,000	16,630
1,000	5,000	1,069
5000	10,000	91
10,000	50,000	112
50,000	100,000	20
100,000	500,000	20
500,000	1,000,000	5
1,000,000		2

Table 4 Breakdown of outgoing accumulative Ether history per address.

Total Ether (≥)	Total Ether (<)	Number of addresses
0	1	917,327
1	10	695,867
10	100	469,766
100	1,000	224,543
1,000	10,000	548,540
10,000	50,000	39,202
50,000	100,000	899
100,000	500,000	648
500,000	5,000,000	128
5,000,000	50,000,000	25
50,000,000		1

Table 5 Breakdown of incoming accumulative Ether history per address.

Total Ether (≥)	Total Ether (<)	Number of addresses
0	1	1,088,717
1	10	863,216
10	100	537,756
100	1,000	242,315
1,000	10,000	546,260
10,000	50,000	36,344
50,000	100,000	717
100,000	500,000	607
500,000	5,000,000	131
5,000,000	50,000,000	26
50,000,000		2

Table 6 The breakdown of Ether balance per address (until May 15, 2017).

Total Ether(≥)	Total Ether (<)	Number of addresses
0	0.01	2,493,480
0.01	0.1	288,026
0.1	1	193,895
1	10	193,057
10	100	87,533
100	1000	28,418
1000	10,000	6,079
10,000	50,000	781
50,000	100,000	98
100,000	500,000	119
500,000	2,500,000	35
2,500,000		16

Table 7 Ethereum network's transaction size distribution.

Total Ether(≥)	Total Ether(<)	Number of addresses
0	0.001	6,552,962
0.001	0.1	4,360,858
0.1	1	8,585,043
1	10	12,544,316
10	100	2,358,529
100	1,000	1,245,886
1,000	10,000	607,476
10,000	50,000	10,815
50,000	100,000	1,040
100,000	500,000	696
500,000	2,500,000	41
2,500,000		2

Table 6 shows that nearly 96% of the addresses' current (May 15, 2017) balance is less than 10 Ether, but this number drops to 82% when looking at the maximum balance that can be seen during the life of these addresses. Table 6 states that only $1,049$ (0.2%) addresses have a balance of $10,000$ or more.

Table 7 represents the distribution of the transaction sizes of the network. At other times, many transactions are very small, and it is noticeable that less than 1 Ether has been received by 53% of transactions. Similarly, considering medium-sized quantities, less than 10 Ether were received by 88% of transactions. Moreover, Table 7 shows that only $1,788$ transactions received greater than $50,000$ Ether.

Ethereum transaction behavior analysis

We analyzed the transaction flow by breaking the data into two phases, in-degree and out-degree relationships. We considered each year (2016, 2017, and 2018) as a single phase and constructed the corresponding network. Since the network grows over time,

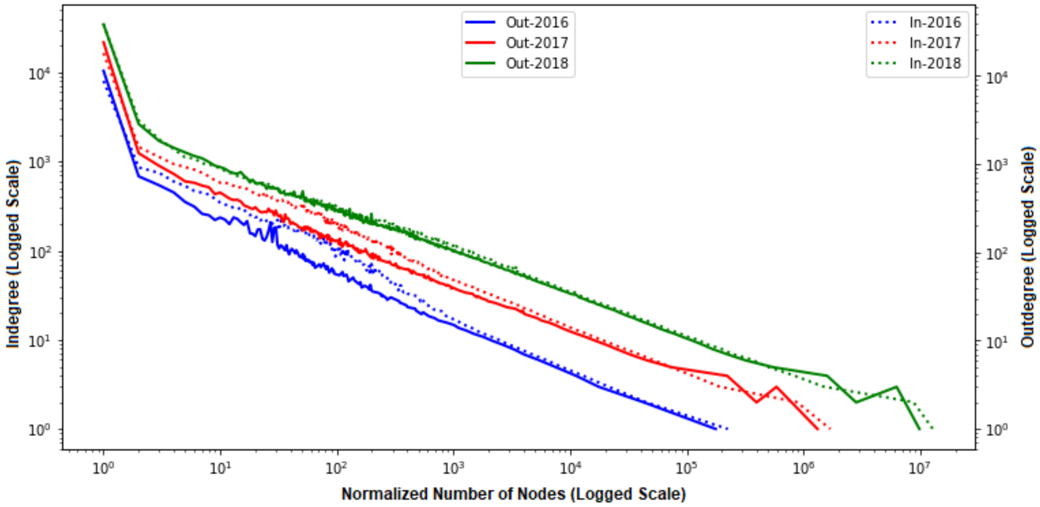


Figure 4 **Degree distributions of various time periods.**

Full-size DOI: 10.7717/peerjcs.815/fig-4

we are also interested in measuring network growth. We first measured our constructed Ethereum networks' degree distributions, as shown in Fig. 4. from the distribution, we approximated to the power law and observed that both the out-degree and in-degree are relatively uniform. Also, the number of nodes and their degrees are increasing with time passing.

Figure 5 draws a Lorenz curve (a graphical representation of the Gini Coefficient) to additionally characterize the evolution of the order distribution and calculate the "Gini Coefficient" with other timestamps. In order to measure the inequalities that present in the breakdown of wealth, we used such a scale because it is also used to calculate the heterogeneity of the empirical data. In general, the Gini coefficient is calculated as follows:

$$G_c = \frac{2\sum_{j=1}^{t} jx_j}{t\sum_{j=1}^{t} x_j} - \frac{t+1}{t}.$$

Here, x_j is the jth sample from t data points, and x_j is ordered monotonically, *i.e.*, $x_j \leq x_{j+1}.G_c = 1$, implies complete inequality and $G_c = 0$ indicates perfect equality in wealth distribution, *i.e.*, all nodes have the same wealth amount.

Figure 6A shows that the Ethereum network is changing with time. The line of equality is indicated using the Yellow line. If other lines get closer to it, this means that the system is moving to equality. As we see from the figure, EFTN moves towards equality as the curves get closer to the Lorenz curve as time passes. The Gini coefficient computed for out-degree each year was $G^{out} \simeq 0.96$, $G^{out} \simeq 0.92$ and $G^{out} \simeq 0.85$ respectively for years 2016, 2017 and 2018.

Similar behavior for network in-degree was also observed, as shown in Fig. 6B. Gini Coefficient values were $G^{in} \simeq 0.95$, $G^{in} \simeq 0.90$ and $G^{in} \simeq 0.83$ for each year 2016, 2017 and 2018 respectively. For both in and out degrees, the Gini Coefficient values are close to 1 for each year under consideration. This implies large inequality among sending

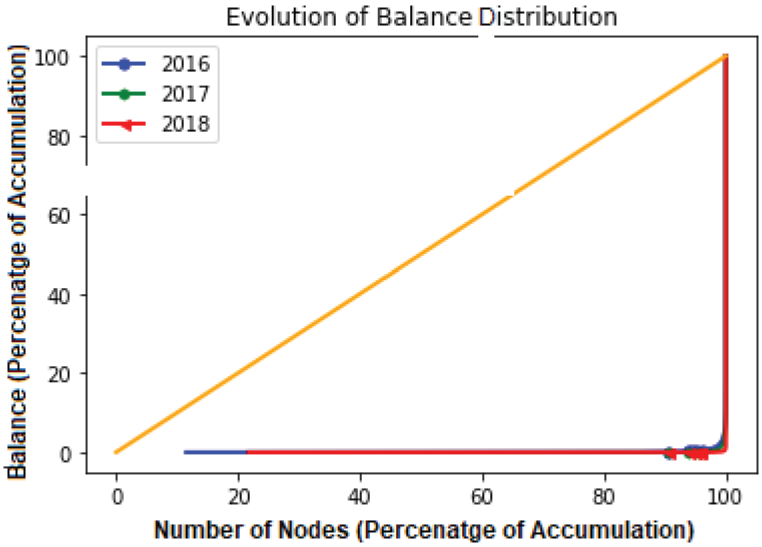


Figure 5 The Lorenz curve of the address balance at other moments.
Full-size DOI: 10.7717/peerjcs.815/fig-5

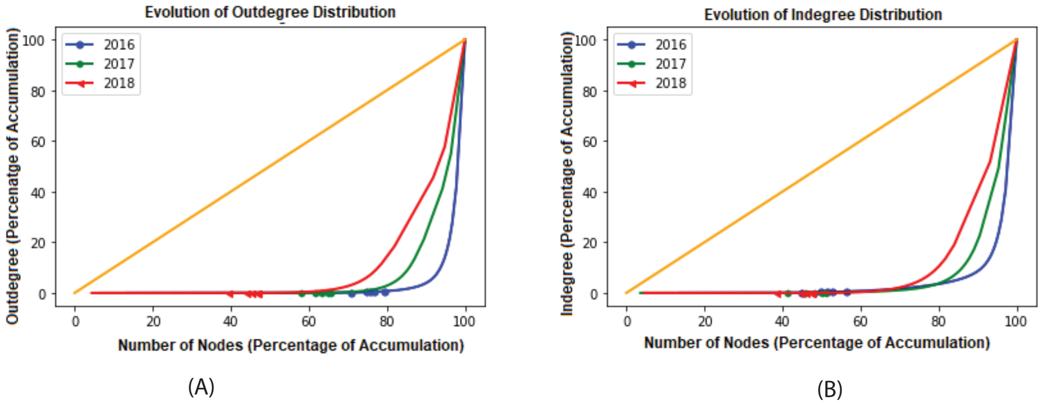


Figure 6 Different time frames of Lorenz curves for out-degree and in-degree.
Full-size DOI: 10.7717/peerjcs.815/fig-6

and receiving transactions distributions. Apart from the in-degree and out-degree distributions, we can observe lacking balance among addresses, as shown in Fig. 5. The figure indicates only a few addresses own a major part of the Ethers representing perfect inequality in the distribution.

We analyzed nodes with a high degree compared to other nodes in the network. Nodes with higher order are assumed to have higher balances. In Fig. 7A, it is noticed that higher proportion is associated with higher in-degree nodes till date 2018-04-25. However, there is no relation between out-degree and the balance as depicted in Fig. 7B. Therefore, we concluded that the distribution of the Ether is associated more with the in-degrees rather than the out-degrees.

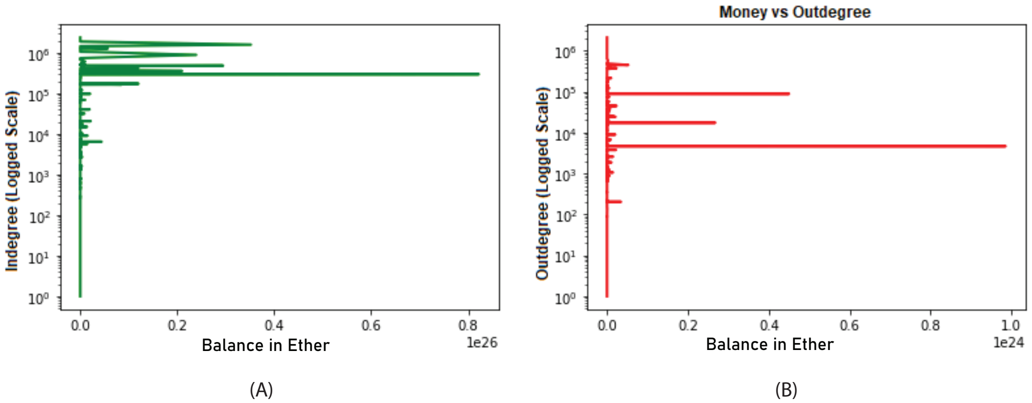


Figure 7 **Relation between balance and in- and out-degrees (until 2018-04-25).**
Full-size DOI: 10.7717/peerjcs.815/fig-7

Ethereum community structure analysis

To explore our ETFN network's community structure, we deployed the Louvain algorithm in our experimental setting. The histogram representation of the network community structure is shown in Fig. 8 depicting an exciting observation. On the x-axis, the number of communities is shown, while the y-axis represents individuals' count (addresses in this case) in each community. We can see that the entire network comprises five major communities while a few other smaller communities. The community distribution shows quite interesting observation resembling the community distribution of most of the real-world networks (*Said et al., 2018*). Moreover, one central community contains many influential addresses and covers most of the network (around 30%). These results indicate that EFTN consists of some excellent community structure, and thus, various network theory measures can be deployed to mine further hidden information from it.

Ethereum link prediction analysis

This section considers standard Area Under the Curve (AUC) and Average Precision (AP) matrices for evaluation. The performance of VGAE in terms of AUC and AP on both the networks is shown in Fig. 9. We can see that the VGAE model has shown outstanding performance while achieving 87.6% AUC on $\mathcal{G}_1$ and 91.59% AUC on $\mathcal{G}_2$ networks. Similarly, it shows 88.28% and 88.5% AP on $\mathcal{G}_1$. These results demonstrate the effectiveness of VGAE on the Ethereum transaction data. Furthermore, we observe that both the networks have similar statistics and structures; ergo, the performance of the models is also quite closed using both evaluation metrics.

DISCUSSION AND FUTURE WORK

In this study, we provided a set of analyses based on the Ethereum network as follows. First, we noticed that most Ethereum addresses are associated with a few transactions when analyzing the outgoing and incoming accumulative Ether history per address. Second, we observed that the number of nodes and their degrees during 2016-2018 increased with time regarding the measurement of the in-degree and out-degree transaction relationship. Specifically, we discovered that the distribution of Ether is more associated

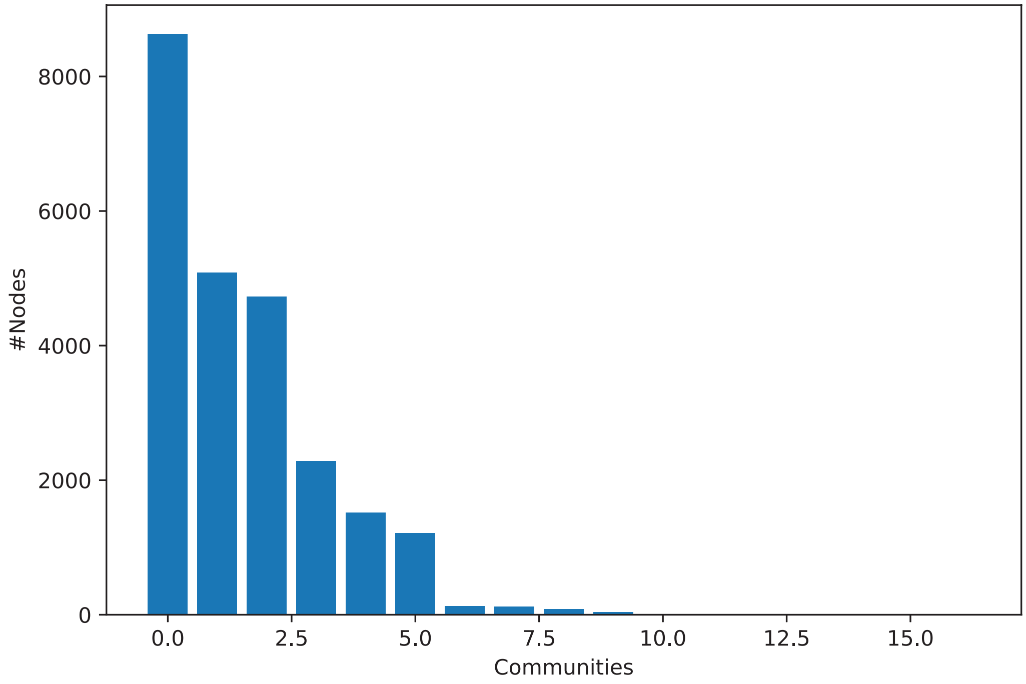


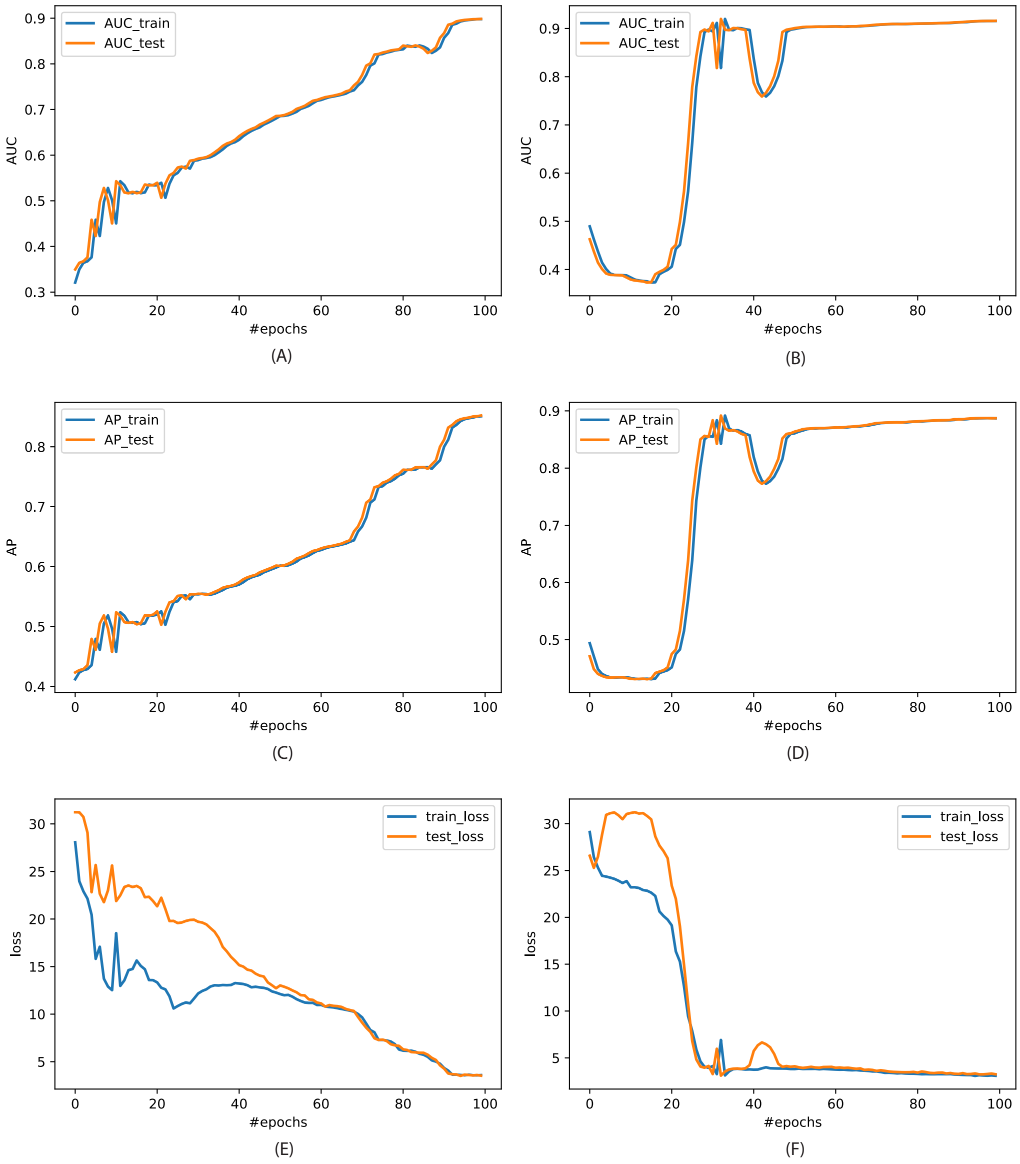
Figure 8 **Histogram representation of EFTN community structure.**
Full-size DOI: 10.7717/peerjcs.815/fig-8

with the in-degrees rather than out-degrees. Third, we recognize five major communities from the entire network. Lastly, the performance of VGAE on Ethereum's link prediction in terms of area under the curve and average precision matrices is outstanding, with over 80% on sub-networks over time.

In the future, we plan to use our findings in this study as a groundwork for comparing the statistical features from more Ethereum data, examining the evolution of temporal properties in the transaction network, and gaining a better understanding of the complex interaction between the transaction network and the social network. In addition, we could investigate graph algorithms that can handle the community detection and link prediction problems altogether, using either traditional graph analysis (*Lü & Zhou, 2011*) or graph representation learning methods (*Choong, Liu & Murata, 2018*; *Liu et al., 2020*). Also, this study could lay the direction for further research on optimizing and managing the optimal usage of the Ethereum network for better network maintenance. Finally, more recent data could be collected and processed to investigate the evolution of the network behavior over time.

CONCLUSIONS

In this paper, we proposed a Detailed Analysis of Ethereum Network on Transaction Behavior, Community Structure and Link Prediction framework (DANET) to track the evolution of Ethereum transactional data from the perspective of graph analysis. Also, we investigated wealth distribution over Ethereum in terms of network degree and



(A)

(B)

(C)

(D)

(E)

(F)

Figure 9 **(A & B): Area Under the Curve (AUC) of VGAE model on both $\mathcal{G}_1$ and $\mathcal{G}_2$ for 100 epochs.** (C & D): The performance in terms of Average Precision (AP). (E & F): The corresponding loss curves.
Full-size DOI: 10.7717/peerjcs.815/fig-9

explored the network's community structure showing a piece of exciting information. We further performed link prediction using variational graph auto-encoders on a small set of transaction data. The model showed impressive prediction accuracy on the link prediction task. By examining these graphs through several metrics, we gain many new observations and insights, which could assist the understanding of the Ethereum network.

ACKNOWLEDGEMENTS

This research project is supported by Mahidol University, Thailand, and Blockchain Lab, ITU National Centre for Cyber Security (NCCS), Pakistan.

ADDITIONAL INFORMATION AND DECLARATIONS

Funding

This research project is supported by Mahidol University, Thailand, and Blockchain Lab, ITU National Centre for Cyber Security (NCCS) Pakistan. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Grant Disclosures

The following grant information was disclosed by the authors:
Mahidol University, Thailand, and Blockchain Lab, ITU National Centre for Cyber Security (NCCS) Pakistan.

Competing Interests

The authors declare there are no competing interests.

Author Contributions

- Anwar Said conceived and designed the experiments, performed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, authored or reviewed drafts of the paper, and approved the final draft.
- Muhammad Umar Janjua performed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, and approved the final draft.
- Saeed-Ul Hassan conceived and designed the experiments, performed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, authored or reviewed drafts of the paper, and approved the final draft.
- Zeeshan Muzammal performed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, authored or reviewed drafts of the paper, and approved the final draft.
- Tania Saleem performed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, and approved the final draft.
- Tipajin Thaipisutikul and Suppawong Tuarob analyzed the data, prepared figures and/or tables, authored or reviewed drafts of the paper, and approved the final draft.
- Raheel Nawaz analyzed the data, authored or reviewed drafts of the paper, and approved the final draft.

Data Availability

The following information was supplied regarding data availability:

The code and dataset are available at GitHub: https://github.com/Anwar-Said/Link-Predictability-using-VGAE.

REFERENCES

Ahmed A, Hassan ZR, Shabbir M. 2020. Interpretable multi-scale graph descriptors via structural compression. *Information Sciences* **533**:169–180 DOI 10.1016/j.ins.2020.05.032.

Akhtar MM, Khan MZ, Ahad MA, Noorwali A, Rizvi DR, Chakraborty C. 2021. Distributed ledger technology based robust access control and real-time synchronization for consumer electronics. *PeerJ Computer Science* **7**:e566 DOI 10.7717/peerj-cs.566.

Androulaki E, Karame GO, Roeschlin M, Scherer T, Capkun S. 2013. Evaluating user privacy in bitcoin. In: Sadeghi AR, ed. *Financial Cryptography and Data Security. FC 2013. Lecture Notes in Computer Science*. vol. 7859. Springer: Berlin, Heidelberg DOI 10.1007/978-3-642-39884-1_4.

Ao X, Liu Y, Qin Z, Sun Y, He Q. 2021. Temporal high-order proximity aware behavior analysis on Ethereum. *World Wide Web* **24**:1565–1585 DOI 10.1007/s11280-021-00875-6.

Barabási A-L, Albert R. 1999. Emergence of scaling in random networks. *Science* **286(5439)**:509–512 DOI 10.1126/science.286.5439.509.

Blondel VD, Guillaume J-L, Lambiotte R., Lefebvre E. 2008. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment* **2008(10)**:P10008 DOI 10.1088/1742-5468/2008/10/P10008.

Bojchevski A, Günnemann S. 2018. Deep gaussian embedding of graphs: unsupervised inductive learning via ranking. In: *International Conference on Learning Representations*. 1–13.

Cai H, Zheng VW, Chang KC-C. 2018. A comprehensive survey of graph embedding: problems, techniques, and applications. *IEEE Transactions on Knowledge and Data Engineering* **30(9)**:1616–1637 DOI 10.1109/TKDE.2018.2807452.

Chan W, Olmsted A. 2017. Ethereum transaction graph analysis. In: *2017 12th international conference for internet technology and secured transactions (ICITST)*. Piscataway: IEEE, 498–500.

Choong JJ, Liu X, Murata T. 2018. Learning community structure with variational autoencoder. In: *2018 IEEE international conference on data mining (ICDM)*. Piscataway: IEEE, 69–78.

Farrugia S, Ellul J, Azzopardi G. 2020. Detection of illicit accounts over the Ethereum blockchain. *Expert Systems with Applications* **150**:113318 DOI 10.1016/j.eswa.2020.113318.

Gallicchio C, Micheli A. 2010. Graph echo state networks. In: *The 2010 international joint conference on neural networks (IJCNN)*. Piscataway: IEEE, 1–8.

Gavin J, Crane M. 2021. Community detection in cryptocurrencies with potential applications to portfolio diversification. ArXiv preprint. arXiv:2108.09763.

Gencer AE, Basu S, Eyal I, van Renesse R, Sirer EG. 2018. Decentralization in Bitcoin and Ethereum Networks. *CoRR*. ArXiv preprint. arXiv:1801.03998.

Gervais A, Karame GO, Wüst K, Glykantzis V, Ritzdorf H, Capkun S. 2016. On the security and performance of proof of work blockchains. In: *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*. 3–16.

Grover A, Leskovec J. 2016. node2vec: scalable feature learning for networks. In: *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. New York: ACM, 855–864.

Hamilton W, Ying Z, Leskovec J. 2017. Inductive representation learning on large graphs. In: *Advances in neural information processing systems*. 1024–1034.

Hamilton WL. 2020. Graph Representation Learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning* **14(3)**:1–159.

Harlev MA, Sun Yin H, Langenheldt KC, Mukkamala R, Vatrapu R. 2018. Breaking bad: de-anonymising entity types on the bitcoin blockchain using supervised machine learning. In: *Proceedings of the 51st Hawaii international conference on system sciences.*.

Hirshman J, Huang Y, Macke S. 2013. Unsupervised approaches to detecting anomalous behavior in the bitcoin transaction network. 3rd Ed. Technical Report. Stanford University, Stanford, CA, USA.

Karame G, Androulaki E, Capkun S. 2012. Two Bitcoins at the price of one? Double-spending attacks on fast payments in bitcoin. *IACR Cryptology EPrint Archive* **2012(248)**.

Katz L. 1953. A new status index derived from sociometric analysis. *Psychometrika* **18(1)**:39–43 DOI 10.1007/BF02289026.

Kipf TN, Welling M. 2016a. Semi-supervised classification with graph convolutional networks. ArXiv preprint. arXiv:1609.02907.

Kipf TN, Welling M. 2016b. Variational graph auto-encoders. ArXiv preprint. arXiv:1611.07308.

Koshy P, Koshy D, McDaniel P. 2014. An analysis of anonymity in bitcoin using p2p network traffic. In: Christin N, Safavi-Naini R, eds. *Financial Cryptography and Data Security. FC 2014. Lecture Notes in Computer Science*. vol. 8437. Berlin, Heidelberg: Springer DOI 10.1007/978-3-662-45472-5_30.

Li Y, Islambekov U, Akcora C, Smirnova E, Gel YR, Kantarcioglu M. 2020. Dissecting ethereum blockchain analytics: what we learn from topology and geometry of the ethereum graph? In: *Proceedings of the 2020 SIAM international conference on data mining*. Philadelphia: SIAM, 523–531.

Liben-Nowell D, Kleinberg J. 2007. The link-prediction problem for social networks. *Journal of the American Society for Information Science and Technology* **58(7)**:1019–1031 DOI 10.1002/asi.20591.

Liu F, Xue S, Wu J, Zhou C, Hu W, Paris C, Nepal S, Yang J, Yu PS. 2020. Deep learning for community detection: progress, challenges and opportunities. In: *Proceedings of the twenty-ninth international joint conference on artificial intelligence, IJCAI-20*. 4981–4987 DOI 10.24963/ijcai.2020/693.

Lü L, Zhou T. 2011. Link prediction in complex networks: a survey. *Physica a: Statistical Mechanics and Its Applications* **390(6)**:1150–1170 DOI 10.1016/j.physa.2010.11.027.

Ma R, Huang B, Huang Z, Zhang Z. 2021. Genome-wide identification and analysis of the YABBY gene family in Moso Bamboo (Phyllostachys edulis (Carrière) J. Houz). *PeerJ* **9**:e11780 DOI 10.7717/peerj.11780.

Maeng SH, Essaid M, Ju HT. 2020. Analysis of ethereum network properties and behavior of influential nodes. In: *2020 21st Asia-Pacific network operations and management symposium (APNOMS)*. Piscataway: IEEE, 203–207.

Meiklejohn S, Pomarole M, Jordan G, Levchenko K, McCoy D, Voelker GM, Savage S. 2013. A fistful of bitcoins: characterizing payments among men with no names. In: *Proceedings of the 2013 conference on Internet measurement conference*. 127–140.

Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J. 2013. Distributed representations of words and phrases and their compositionality. In: *Advances in Neural Information Processing Systems, volume 26*. Curran Associates, Inc, *Available at https://proceedings.neurips.cc/paper/2013/file/9aa42b31882ec039965f3c4923ce901b-Paper.pdf* .

Möser M, Böhme R, Breuker D. 2013. An inquiry into money laundering tools in the Bitcoin ecosystem. In: *2013 APWG eCrime researchers summit*. Piscataway: IEEE, 1–14.

Muzammal Z, Janjua MU, Abbas W, Sher F. 2019. Wealth distribution and link predictability in ethereum. In: *IEEE/WIC/ACM international conference on web intelligence-companion Volume*. New York: ACM, 184–192.

Nakamoto S. 2019. Bitcoin: a peer-to-peer electronic cash system. *Technical report, Manubot. Available at https://bitcoin.org/bitcoin.pdf* .

Nerurkar P, Patel D, Busnel Y, Ludinard R, Kumari S, Khan MK. 2021. Dissecting bitcoin blockchain: empirical analysis of bitcoin network (2009–2020). *Journal of Network and Computer Applications* **177**:102940 DOI 10.1016/j.jnca.2020.102940.

Newman ME. 2006. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences* **103(23)**:8577–8582 DOI 10.1073/pnas.0601602103.

Newman ME, Girvan M. 2004. Finding and evaluating community structure in networks. *Physical Review E* **69(2)**:026113 DOI 10.1103/PhysRevE.69.026113.

Ober M, Katzenbeisser S, Hamacher K. 2013. Structure and anonymity of the bitcoin transaction graph. *Future Internet* **5(2)**:237–250 DOI 10.3390/fi5020237.

Reid F, Harrigan M. 2011. An analysis of anonymity in the bitcoin system. In: *Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE third inernational conference on social computing (SocialCom), 2011 IEEE third international conference on*. Piscataway: IEEE, 1318–1326.

Remy C, Rym B, Matthieu L. 2017. Tracking bitcoin users activity using community detection on a network of weak signals. In: Cherifi C, Cherifi H, Karsai M, Musolesi M, eds. *Complex Networks & Their Applications VI. COMPLEX NETWORKS 2017. Studies in Computational Intelligence*. vol. 689. Cham: Springer DOI 10.1007/978-3-319-72150-7_14.

Rodriguez-Garcia M, Sicilia M-A, Dodero JM. 2021. A privacy-preserving design for sharing demand-driven patient datasets over permissioned blockchains and P2P secure transfer. *PeerJ Computer Science* **7**:e568 DOI 10.7717/peerj-cs.568.

Rosenfeld M. 2014. Analysis of hashrate-based double spending. ArXiv preprint. arXiv:1402.2009.

Said A, Abbasi RA, Maqbool O, Daud A, Aljohani NR. 2018. CC-GA: a clustering coefficient based genetic algorithm for detecting communities in social networks. *Applied Soft Computing* **63**:59–70 DOI 10.1016/j.asoc.2017.11.014.

Said A, Bowman TD, Abbasi RA, Aljohani NR, Hassan S-U, Nawaz R. 2019. Mining network-level properties of Twitter altmetrics data. *Scientometrics* **120**(**1**):217–235 DOI 10.1007/s11192-019-03112-0.

Said A, Hassan S-U, Abbas W, Shabbir M. 2020. NetKI: a kirchhoff index based statistical graph embedding in nearly linear time. *Neurocomputing* **433**:108–118.

Said A, Hassan S-U, Tuarob S, Nawaz R, Shabbir M. 2021. DGSD: distributed graph representation via graph statistical properties. *Future Generation Computer Systems* **119**:166–175 DOI 10.1016/j.future.2021.02.005.

Scarselli F, Gori M, Tsoi AC, Hagenbuchner M, Monfardini G. 2009. The graph neural network model. *IEEE Transactions on Neural Networks* **20**(**1**):61–80 DOI 10.1109/TNN.2008.2005605.

Shah RS, Bhatia A, Gandhi A, Mathur S. 2021. Bitcoin Data Analytics: scalable techniques for transaction clustering and embedding generation. In: *2021 international conference on communication systems & NETworkS (COMSNETS)*. Piscataway: IEEE, 1–6.

Wai H-T, Segarra S, Ozdaglar AE, Scaglione A, Jadbabaie A. 2018. Community detection from low-rank excitations of a graph filter. In: *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. Piscataway: IEEE, 4044–4048.

Wood G. 2014. Ethereum: a secure decentralised generalised transaction ledger. *Ethereum Project Yellow Paper* **151**(**2014**):1–32.

Wu J, Lin D, Zheng Z, Yuan Q. 2019. T-EDGE: temporal weighted multidigraph embedding for Ethereum transaction network analysis. ArXiv preprint. arXiv:1905.08038.

Wu SX, Wu Z, Chen S, Li G., Zhang S. 2021. Community detection in blockchain social networks. *Journal of Communications and Information Networks* **6**(**1**):59–71.

Xie Y, Zhou J, Wang J, Zhang J., Sheng Y, Wu J, Xuan Q. 2021. Understanding ethereum transactions via network approach. In: *Graph data mining*. Singapore: Springer, 155–176.

Yu W, Zheng C, Cheng W, Aggarwal CC, Song D, Zong B, Chen H, Wang W. 2018. Learning deep network representations with adversarially regularized autoencoders. In: *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. New York: ACM, 2663–2671.

Zhang Z, Cui P, Zhu W. 2020. Deep learning on graphs: a survey. *IEEE Transactions on Knowledge and Data Engineering* DOI 10.1109/TKDE.2020.2981333.

Zhang M, Li P, Xia Y, Wang K, Jin L. 2020. Revisiting graph neural networks for link prediction. ArXiv preprint. arXiv:2010.16103.

Zhang Y, Wang J, Zhao F. 2020. Transaction community identification in bitcoin. In: *2020 13th international symposium on computational intelligence and design (ISCID)*. Piscataway: IEEE, 140–144.

Zhang Y, Yang Z, Yu B, Chen H, Li Y, Zhao X. 2021. Structure-enhanced graph representation learning for link prediction in signed networks. In: Qiu H, Zhang C, Fei Z, Qiu M, Kung SY, eds. *Knowledge Science, Engineering and Management. KSEM 2021. Lecture Notes in Computer Science*. vol. 12815. Cham: Springer, 40–52 DOI 10.1007/978-3-030-82136-4_4.