

Illegal Community Detection in Bitcoin Transaction Networks

Dany Kamuhanda ^{1,2,3}, Mengtian Cui ⁴ and Claudio J. Tessone ^{1,2,*} 

¹ UZH Blockchain Center, University of Zurich, 8050 Zurich, Switzerland; dany.kamuhanda@uzh.ch

² Blockchain & Distributed Ledger Technologies Group, Department of Informatics, University of Zurich, 8050 Zurich, Switzerland

³ Department of Mathematics, Science and Physical Education, University of Rwanda-College of Education, Rwamagana P.O. Box 55, Rwanda

⁴ College of Computer Science and Engineering, Southwest Minzu University, Chengdu 610040, China; mengtian.cui@swun.edu.cn

* Correspondence: claudio.tessone@uzh.ch

Abstract: Community detection is widely used in social networks to uncover groups of related vertices (nodes). In cryptocurrency transaction networks, community detection can help identify users that are most related to known illegal users. However, there are challenges in applying community detection in cryptocurrency transaction networks: (1) the use of pseudonymous addresses that are not directly linked to personal information make it difficult to interpret the detected communities; (2) on Bitcoin, a user usually owns multiple Bitcoin addresses, and nodes in transaction networks do not always represent users. Existing works on cluster analysis on Bitcoin transaction networks focus on addressing the later using different heuristics to cluster addresses that are controlled by the same user. This research focuses on illegal community detection containing one or more illegal Bitcoin addresses. We first investigate the structure of Bitcoin transaction networks and suitable community detection methods, then collect a set of illegal addresses and use them to label the detected communities. The results show that 0.06% of communities from daily transaction networks contain one or more illegal addresses when 2,313,344 illegal addresses are used to label the communities. The results also show that distance-based clustering methods and other methods depending on them, such as network representation learning, are not suitable for Bitcoin transaction networks while community quality optimization and label-propagation-based methods are the most suitable.

Keywords: bitcoin; transaction networks; blockchain; cryptocurrency; community detection



Citation: Kamuhanda, D.; Cui, M.; Tessone, C.J. Illegal Community Detection in Bitcoin Transaction Networks. *Entropy* **2023**, *25*, 1069. <https://doi.org/10.3390/e25071069>

Academic Editor: Stanisław Drożdż

Received: 12 June 2023

Revised: 4 July 2023

Accepted: 10 July 2023

Published: 16 July 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Blockchain is a secured, distributed storage of records organized in a chain of blocks of transactions. Records cannot be modified or deleted once they have been added to the blockchain, which ensures data integrity. In permissioned blockchains [1], only authorized individuals can access transactions, whereas in permissionless blockchains such as Bitcoin [2], transactions are publicly available, and anyone can join the network. Bitcoin is a protocol launched in 2009 to facilitate the transfer of its cryptocurrency, Bitcoin (BTC), between users without the need for a central authority, as opposed to other digital payments that are governed by financial institutions [2]. Bitcoin transactions do not reveal personal information about the involved users; instead, users are represented by pseudonymous addresses.

A Bitcoin address consists of at least 26 alphanumeric characters beginning with 1, 3 or bc1 depending on the type of the address: (a) P2PKH, (b) P2SH, or (c) Bech32 [3]. By using such addresses, it is hard to trace the owner. There are concerns that Bitcoin and other cryptocurrency platforms can be used for illegal activities. Researchers and governments are interested in understanding interactions in cryptocurrency transaction networks. Interactions can be modelled as a network (graph) G consisting of a set of vertices

(nodes) V and a set of edges E , where $n = |V|$ and $m = |E|$. For example, the Bitcoin transaction network can be modelled as a set of transactions: https://www.blockchain.com/explorer/api/blockchain_api (accessed on 28 February 2023) with a directed link from an input transaction to output transactions. It can also be modelled as a network of users where users are vertices, and a link between users indicates the flow of BTC from one user to another. Vallarano et al. [4] used this modelling to investigate the relationship between users' behaviour and cryptocurrency pricing in exchange markets by focusing on Bitcoin, while De Collibus et al. [5,6] focused on understanding the growth of the Ethereum network. Other works focused on the forensics, such as entity identification [7,8] and using machine learning to detect illegal transactions [9–11]. Another way to analyse blockchain transaction networks is to understand the cohesive sub-groups that exist in such networks. Previous work on cluster analysis in blockchain transaction networks focused on address clustering, a task that uses heuristics to group addresses that are assumed to be controlled by the same user [3,12]. Community detection was used to cluster users of similar characteristics [13] but the obtained communities could not be interpreted further as they consisted of pseudonymous addresses. This work combines community detection and off-chain information to quantify the presence of illegal addresses in daily Bitcoin transactions.

This work provides the following contributions: We address the illegal community detection problem in blockchain transaction networks with an emphasis on Bitcoin. Using a set of known illegal addresses linked to illegal darknet markets that have been shut down, we quantify the presence of illegal communities from daily Bitcoin transactions. Illegal community detection is essential in Bitcoin forensics as it helps to identify the addresses most related to known illegal addresses. We benchmark different community detection methods and explain why some are not suitable for Bitcoin transaction networks. From various community detection methods [14,15], we benchmark representative methods from distance-based clustering, network representation learning, spectral clustering, community quality optimization, label propagation and clique-based methods using the conductance, modularity and running time.

The rest of the article is organized as follows: Section 2 discusses the preliminaries, which include blockchain, community, and community evaluation measures. Section 3 discusses materials and methods, which include the data, community detection methods, and the approach used to detect illegal communities. Section 4 consists of experiments and results. Section 5 provides a conclusion.

2. Preliminaries

Traditional electronic payments rely on financial institutions as trusted third parties, Bitcoin was launched in 2009 as a decentralized, trustless electronic cash system that offer individuals the ability to transact without a trusted third-party [2]. This was achieved by using a public blockchain to store transactions on a distributed network of nodes and the proof-of-work consensus mechanism to confirm transactions rather than a central authority. When a user initiates a transaction, it is propagated over the network as an unconfirmed transaction. Miners on the network use their powerful computing hardware to compete in solving complex mathematical puzzles (mining) to validate blocks of unconfirmed transactions. Once validated, a block is disseminated over the network and appended to the previous block. The miner of the block receives rewards in the form of minted BTC and transaction fees. Once a block is confirmed and added to the blockchain, its contents become immutable, ensuring the integrity of the recorded data.

While most blockchains store data linearly, emerging ways of storing data using directed acyclic graphs (DAG) have been used [16,17]. A blockchain transaction refers to an action that adds data to the blockchain. It can be a payment in cryptocurrency blockchains; registration or transfer of land ownership in a blockchain-based land management system; or a vote in a blockchain-based voting system, which can be used to ensure the immutability of votes and transparency. As opposed to banking systems or traditional voting systems

where users can be identified with personal information, in blockchain-based systems, users use pseudonymous addresses, which make it difficult to trace the owner [7].

User interactions such as friendships in social networks and blockchain transactions are often represented with a network. A network is modelled as a graph $G = (V, E)$ where V is a set of n vertices (nodes): $V = \{v_1 \cdots v_n\}$ and E a set of m edges: $E = \{e_1 \cdots e_m\} \subseteq V \times V$. Network nodes can be organized into k sets of related nodes known as communities $C = \{C_1 \cdots C_k\}$. We define a Bitcoin transaction network as a graph in which nodes are Bitcoin addresses and the edges denote the flow of bitcoins (BTC) between addresses.

Networks usually consist of cohesive subgroups of nodes known as communities. A community is generally defined as a group of nodes that are densely connected between themselves with sparse connections to the rest of the network [18]. This definition focuses on disjointed communities. In overlapping communities, if most of the community members belong to other communities, the overlapping section can become denser than the community. In this case, a community is considered to be a group of nodes with a high probability of being connected to one another than to any other node of the network [14]. An illegal community $C_i \in C$ consists of one or more members who have been involved in illegal activities, such as money laundering, scam, terrorism financing and many more.

The conductance of a community C_i of a network G , is a community evaluation measure indicating a fraction of community edges that point outside the community [19]:

$$\varnothing(G, C_i) = \frac{e'(C_i)}{\min(e(C_i), e(V - C_i))} \quad (1)$$

$e(x)$ denotes the total number of edges pointing to nodes in x , while $e'(x)$ denotes the total number of edges that link x to the rest of the network. The lower the conductance, the better the community.

The modularity Q measures the strength of the division of a network G into its communities C . As opposed to conductance, the higher the modularity, the better the communities. For disjointed communities, the modularity is defined as [20]:

$$Q(G, C) = \frac{1}{2m} \sum_{ij} \left[\left(A_{ij} - \frac{d_i d_j}{2m} \right) \delta(h_i, h_j) \right] \quad (2)$$

where A is the adjacency matrix, d_i is the degree of the node i , m is the number of edges of the network and $h \in \mathbb{N}^n$ is a community assignment vector for n nodes. The function $\delta(h_i, h_j)$ is 1 if the community assignment value h_i for node i is the same as the community assignment h_j for node j . An adapted version has been proposed for overlapping communities [21]:

$$Q(G, C) = \frac{1}{k} \sum_{i=1}^k q(C_i) \quad (3)$$

$$q(c) = \frac{1}{n_c} \sum_{i \in c} \frac{\sum_{j \in c, j \neq i} A_{ij} - \sum_{j \notin c} A_{ij}}{d_i \cdot k_i} \cdot \frac{m_c}{\binom{n_c}{2}} \quad (4)$$

where k is the number of communities in G ; k_i the number of communities containing node i ; d_i the degree of a node i ; n_c the number of nodes in the community c ; m_c the number of edges in the community c and A_{ij} an entry in the adjacency matrix A for nodes i and j .

3. Materials and Methods

Bitcoin transactions are publicly accessible using blockchain data APIs: https://www.blockchain.com/explorer/api/blockchain_api (28 February 2023). As a single user often owns multiple addresses, heuristics are used to create a user transaction network (user detection) from the original transaction network. Community detection can be used to

group related network users into clusters, which can be labeled using off-chain information to interpret what the communities represent.

3.1. Illegal Community Detection

We use community detection to detect a group of users associated with illegal Bitcoin addresses. In the cryptocurrency sense, we define a community as a group of users who often send coins between themselves, rather than conducting transactions with the rest of the network. Ground-truth illegal communities could be useful when evaluating the performance of algorithms in detecting illegal communities, but such labeled communities are not available for blockchain transaction networks. However, the known illegal addresses could be used to identify illegal communities, which help to identify more users that are likely to be associated with illegal activities.

We used a two-step approach: (1) we used a community detection method to detect unlabeled communities; (2) then, we used known illegal addresses to label the detected communities. In the first step, we initially benchmarked the representative algorithms discussed in Section 3.4 on Bitcoin user transaction networks in Table 1. The networks were obtained by applying the multi-input heuristic on raw Bitcoin transactions. For each network in Table 2, we used the selected algorithm with the best modularity, average conductance, and speed results to detect the unlabeled communities. In the second step, we iterated through the detected communities, checking the presence of a user with illegal addresses linked to darknet markets in Table 3. Some communities consist of users associated with many illegal addresses, while other communities are associated with one illegal address. Although illegal activities such as money laundering can be carried out by one community member (by sending BTC to addresses he/she controls), other members that often transact with the member might be subject to investigation or become victims. For this reason, we mark the entire community as illegal.

Table 1. Bitcoin transaction networks we used to investigate the structure of Bitcoin transaction networks and suitable community detection methods.

Bitcoin User Transaction Networks	Nodes	Edges
G_1 : 31 December 2010	584	667
G_2 : 31 December 2011	5976	7553
G_3 : 31 December 2012	17,594	29,730
G_4 : 31 December 2013	56,454	93,612
G_5 : 31 December 2014	81,490	127,796
G_6 : 31 December 2015	147,813	238,988
G_7 : 31 December 2016	285,391	421,900
G_8 : 31 December 2017	365,761	557,759
G_9 : 31 December 2018	300,135	418,657
G_{10} : 31 December 2019	346,181	507,036
G_{11} : 31 December 2020	516,398	666,037

Table 2. Bitcoin user transaction networks we used to detect illegal communities.

Bitcoin User Transaction Networks	Nodes	Edges
G_{12} : 25 January 2017	711,255	1,376,711
G_{13} : 26 January 2017	720,994	1,560,367
G_{14} : 27 January 2017	645,053	1,400,766

Table 3. Darknet markets and their bitcoin address count.

Darknet Market	Bitcoin Address Count
AlphaBayMarket	189,776
SilkRoad2Market	350,036
SilkRoadMarketplace	372,753
YABTCL.com	3243
AgoraMarket	498,001
SheepMarketplace	53,639
BlackBankMarket	50,878
PandoraOpenMarket	55,757
NucleusMarket	146,381
BlueSkyMarketplace	18,997
MiddleEarthMarketplace	34,149
AbraxasMarket	119,119
EvolutionMarket	420,615
Total	2,313,344

3.2. Data

We used the Blockchain Data API (https://www.blockchain.com/explorer/api/blockchain_api) to obtain blocks of transactions on different dates, then parse the blocks to construct transaction networks. The multi-input heuristic (Section 3.3) is used to construct user transaction networks, as shown in Tables 1 and 2. To detect illegal communities, we use networks constructed on dates where illegal activities were detected and reported. Based on the money laundering reported by the Federal Bureau of Investigation (FBI) and Homeland Security Investigations (HSI), as discussed in the Case 1:22-mj-00022-RMM [22], we selected blocks of transactions for the 25 January 2017, 26 January 2017 and 27 January 2017, in which money laundering took place. We also randomly selected a recent date 17 January 2023, then construct networks on the selected dates using multi-input heuristics.

Community detection alone does not provide meaningful information, as community nodes are just pseudonymous addresses. Using the Wallet Explorer API: <https://www.walletexplorer.com/> (accessed on 21 October 2022), the IKNAIO API (<https://www.ikna.io/>) and darknet markets, we selected 13 illegal darknet markets that were shut down and collect the associated 2,313,344 Bitcoin addresses, which we consider to be illegal addresses. Wallet Explorer maps entities to Bitcoin addresses used before 2018 [8,23], while IKNAIO is a CryptoAsset Analytics tool with an API that can be used to access its data. Both data sources have been used for data validation.

3.3. User Detection

Multi-input, peel chain and change address identification heuristics were previously proposed for user detection tasks [3,24]. A peel chain is a transaction pattern, which reduces the large amount of cryptocurrency associated with an address through a series of small transactions [3]. The change is the amount that goes back to the sender, who initiated the transaction, and different heuristics have been proposed to detect this [3,24,25]. The multi-input heuristic assumes that multiple input addresses to a transaction are controlled by the same user, as the user collects coins from different addresses to make a payment of a greater amount than that in each of the input addresses [3,24]. Multi-input heuristics are the most consistently used in the literature. We use this in this work to create user transaction networks.

3.4. Community Detection

Different community detection methods exist [14,26]. We investigate and group different methods into six categories based on the techniques used to detect communities. From each category, we select popular methods based on their performance on other social networks. We benchmark them on Bitcoin transaction networks to select a suitable method for illegal community detection.

3.4.1. Distance-Based Methods

K-means [27], DBSCAN [28] and hierarchical clustering [29] are popular distance-based clustering methods. Specifically, k-means and DBSCAN are the most used for cluster analysis in Bitcoin transaction networks [13,24]. We select k-means and DBSCAN from this category for our benchmark.

3.4.2. Spectral Clustering

Spectral clustering starts by computing the Laplacian matrix $L = D - A$ or its variants [30], where D is a degree matrix. The next step computes the eigenvectors of L , which can be achieved using eigenvalue decomposition, to find an approximation of L as $L \cong U\Lambda U^T$. The columns of U are eigenvectors of L , while Λ consists of eigenvalues on the diagonal. The third step extracts from U a matrix H , corresponding to the top k eigenvalues, where each row represents a vertex's representation in k dimensional space. Then, a clustering algorithm such as k-means is applied to H to group its vertices into k communities. The number of communities k is often estimated using eigengap heuristic, where k is related to the largest gap between consecutive eigenvalues λ_k and λ_{k+1} of L compared to gaps between $\lambda_1, \lambda_2 \dots \lambda_k$ [31].

3.4.3. Community Quality Optimization

This category consists of methods that find communities that maximize the modularity [20], minimize the conductance [32], or optimize any other community quality measure. Modularity maximization is the most popular. This method aims to split the network into communities that maximize the modularity measure in Equation (2). Leiden [33] and Louvain [34] are recent popular algorithms based on the modularity maximization, while the first modularity maximization method was introduced by Newman and Girvan [20]. Louvain combines the label propagation method (Section 3.4.4) with the modularity optimization for a fast community detection that is scalable to large networks. We selected Leiden and Louvain for global community detection, and Personalized PageRank (PPR) and heat kernel (HK) [32] to optimize the conductance in Equation (1) for local community detection.

3.4.4. Label Propagation

Label propagation (LP) assumes that a vertex v chooses to be in a community to which most of its neighbors belong [35]. LP starts by ranking all vertices of the input network in a certain order, and each vertex is assigned a unique label. From various iterations, labels propagate through the network until each vertex has a community label, carried by the maximum number of neighbors. As all labels are unique at the first iteration, the first vertex takes a random label from its neighbors and, in the subsequent iterations, ties are broken uniformly and randomly. The adapted label propagation methods have been proposed to handle community detection in dynamic networks [36], to improve the speed [37] and detect overlapping communities [38]. The LP [39] was selected for disjoint community detection, while DEMON [40] was selected for overlapping community detection.

3.4.5. Clique Percolation Method (CPM)

CPM finds communities based on adjacent cliques of size k [41]. A clique is a set of vertices that are completely connected to one another. Two cliques of size k are adjacent if they share $k - 1$ vertices. As the detected communities depend on the clique size k being used as input, these communities are known as k -communities. For example, one may find communities based on adjacent 3-cliques or 5-cliques. We used a 2-cliques approach in our benchmark to capture communities consisting of two users who often transact between themselves.

3.4.6. Network Representation Learning

This category consists of matrix-decomposition-based methods: PCA, SVD [42], NMF [43]; graph-diffusion-based methods: LINE [44], DeepWalk [45], Node2Vec [46]; Autoencoder [47], and graph neural networks [48]. These methods aim to learn a representation of the network that captures similarities between nodes better than the adjacency matrix. Apart from NMF, which can output communities without requiring an additional clustering method [49], other methods in this category require that the obtained embeddings are clustered using distance-based clustering methods. Some methods, such as ClusterGCN include a clustering layer, but require that the number of clusters is specified [50]. We selected Node2Vec due to its popularity for generating embeddings, and DBSCAN to cluster the embeddings into disjoint communities. We also selected BigClam [51], a popular matrix decomposition-based method for overlapping community detection.

4. Experiments and Results

From the different community detection methods that exist, this section aims to investigate the most suitable methods for Bitcoin transaction networks. We also investigate the presence and interactions of 2,313,344 known Bitcoin illegal addresses in daily Bitcoin transactions.

We compared methods that detect communities of the same category using community quality measures and execution time: (1) Leiden, Louvain, label propagation, Node2Vec followed by DBSCAN, and spectral clustering with k -means are compared in terms of their ability to detect disjoint communities in the entire network; (2) clique percolation method (CPM), DEMON and BigClam are compared for their ability to detect overlapping communities; (3) heat kernel (HK) and personalized PageRank (PPR) for their ability to detect local communities to which a seed node of interest belongs. For local community detection, we randomly selected 100 seed nodes and averaged the obtained scores. The modularity measure is only computed on global communities, as it is a global measure, and the average conductance is used to measure the quality of both global and local communities. In the benchmarked algorithms, Node2Vec with DBSCAN denotes that DBSCAN was used to cluster the embeddings generated by Node2Vec. Spectral clustering with k -means denotes that k -means was used to cluster the eigenvectors obtained from the Laplacian matrix. Eigengap heuristic was used to estimate k . Default parameters were used, except for CPM, where 2-cliques were selected to capture the communities consisting of two isolated users who only transacted between themselves.

4.1. Results

The results show that the structure of Bitcoin transaction networks consist of disconnected components as shown in Figure 1.

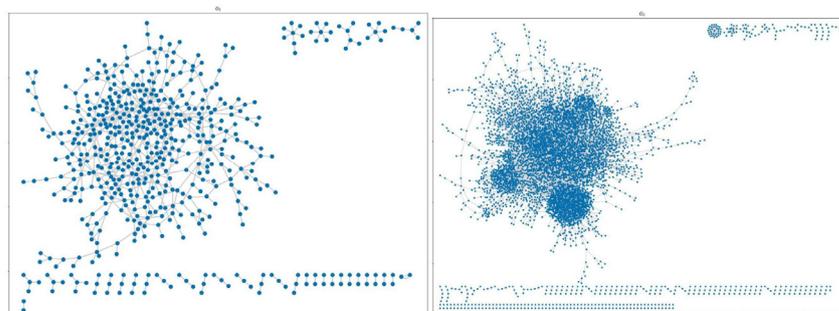


Figure 1. Structure of Bitcoin transaction networks (G_1 and G_2). The networks consist of disconnected components and this pattern remains the same in all analyzed Bitcoin transaction networks.

4.1.1. Benchmark Results

The benchmark of different community detection methods shows that distance-based methods struggle to detect communities on Bitcoin transaction networks that consist of disconnected components. Network representation learning methods also suffer from their dependence on distance-based methods in the clustering phase.

Algorithms that return communities with the highest modularity (Figure 2) also have the lowest conductance (Figures 3 and 4), which indicates that the detected communities are dense, with few or no connections to the rest of the network. This can occur when a dense component is detected as a community. Leiden, Louvain, and label propagation outperformed spectral clustering with k-means and Node2Vec with DBSCAN in detecting densely connected disjoint communities based on modularity and average conductance scores. For global overlapping community detection, the clique percolation method (CPM) outperformed other algorithms from G_1 to G_7 but did not scale beyond networks with more than 300,000 nodes and its results from G_8 to G_{11} were empty (Figure 2).

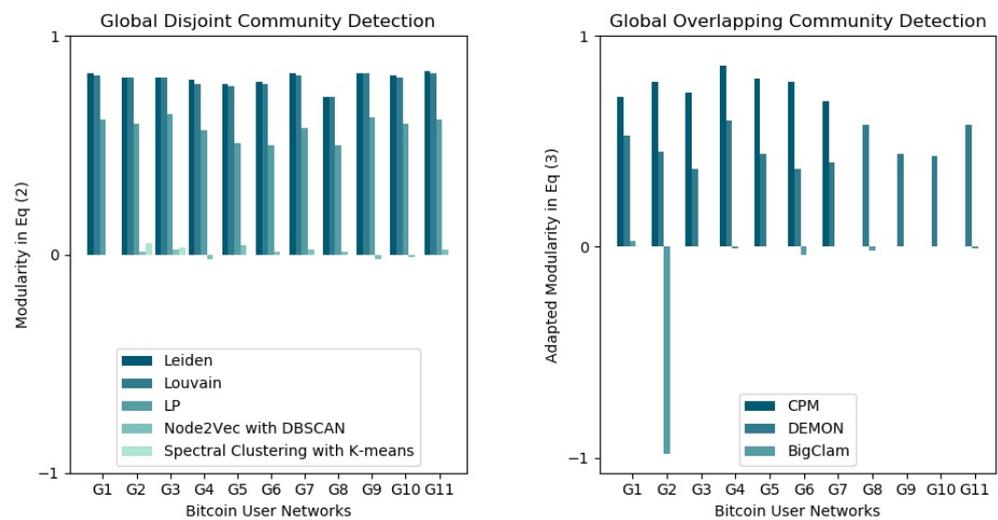


Figure 2. Modularity results. Modularity in Equation (2) was used to evaluate the quality of global disjoint communities, while the modularity in Equation (3) was used for global overlapping communities.

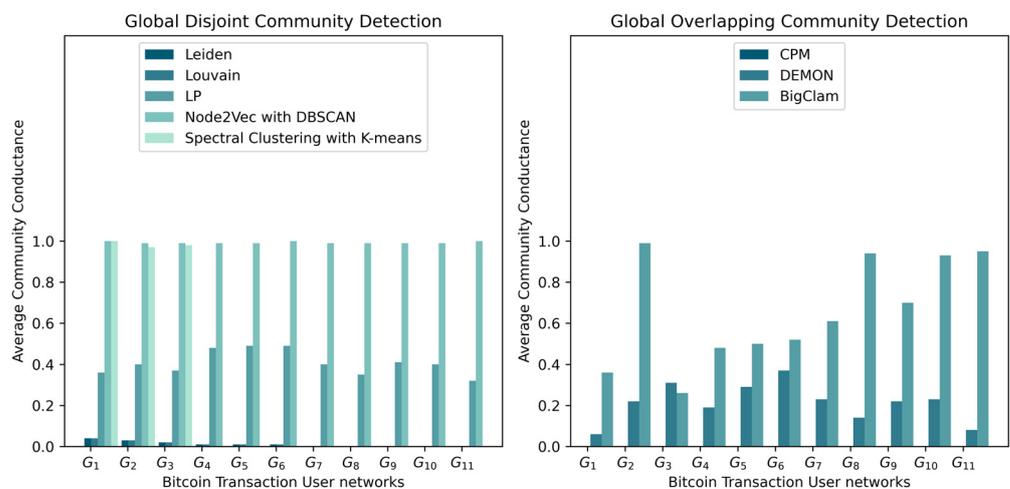


Figure 3. As the conductance measures a fraction of community edges that points outside the community, the conductance consisting of zeros for CPM indicates that all the detected communities have very few or no edges pointing outside the community.

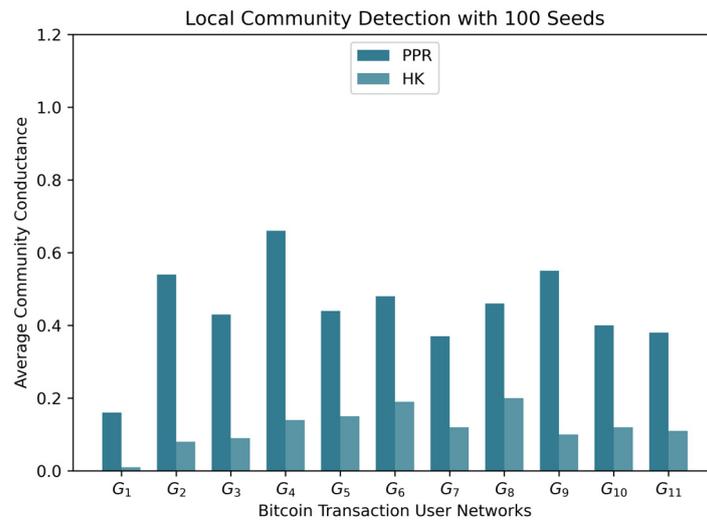


Figure 4. Average conductance results computed on the communities of 100 Bitcoin users. Heat kernel (HK) outperformed personalized PageRank (PPR) in detecting communities that do not have many edges pointing outside.

By analyzing the speed of the algorithms (Figure 5), algorithms that learn the embedding of the network or estimate the number of communities are the slowest. Leiden, Louvain, and label propagation (LP) are the fastest methods and do not use any of those steps. Given their good performance results in detecting dense communities, they are the most suitable for Bitcoin transaction networks.

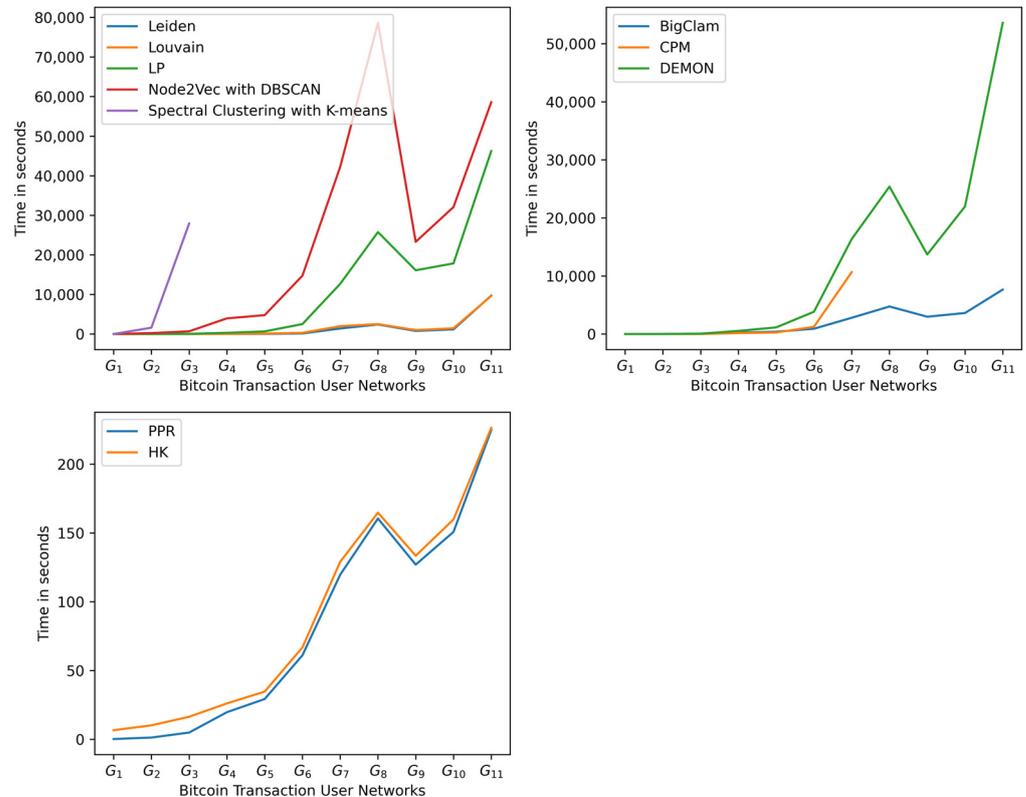


Figure 5. Most of the algorithms that perform well in the detection of good community structures are the fastest, except for heat kernel (HK).

4.1.2. Illegal Community Detection Results

Based on the performance of Louvain in the first experiment, we used it to detect communities that were labeled using known illegal addresses (Table 3) to detect illegal communities. The results show that 0.06% of the detected communities are illegal, with one or more illegal addresses. Figure 6 shows a community from the transactions of 25 January 2017 (G_{12}) containing 19 users associated with 69 illegal addresses.

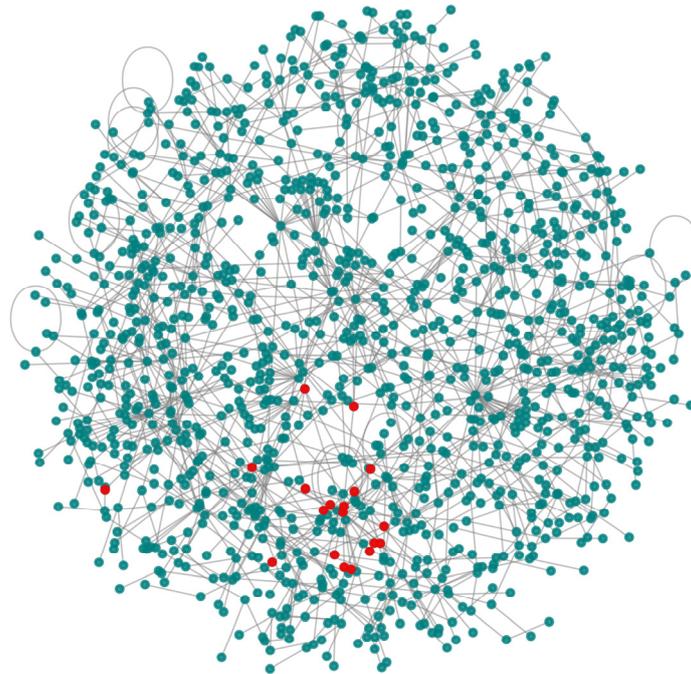


Figure 6. An illegal community consisting of 19 illegal users. Most of these users interact between themselves, as shown by red nodes.

Figure 7 shows ranked communities by size, with associated conductance (left) and size (right). Red markers represented illegal communities. We identified some illegal communities that were isolated from the rest of the network, as indicated by their zero conductance.

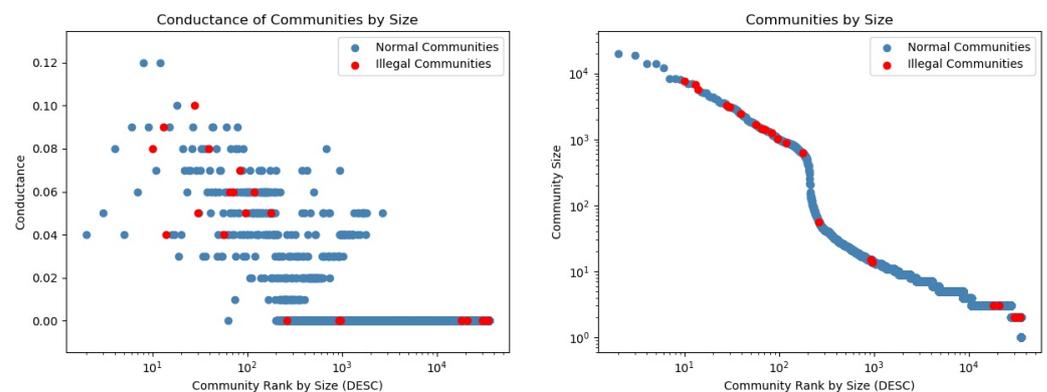


Figure 7. Conductance and size of the detected communities in G_{12} . Red markers indicate illegal communities. We identify large communities with a low conductance (<0.5), which is not usually the case in other social networks. Regarding the distribution of community sizes (right), the steep drop after 10^2 on the x-axis and close to 10^3 on the y-axis indicates that few communities (around 100) are very big, and each of the remaining communities consist of less than 1000 members (the network consisted of 35,296 communities in total).

4.2. Discussion

As community detection with spectral clustering and Node2Vec consists of two phases, embedding and clustering, we investigated the phase that contributed to the poor community detection results. We identified that the embedding phase could learn representations with visible clusters (Figure 8) and the clustering phase was the issue (Figure 9). Figure 8 shows the embedded representation of G_2 presented in Figure 1 (right section) with visible clusters while Figure 9 shows poor performance of k-means with different values of k .

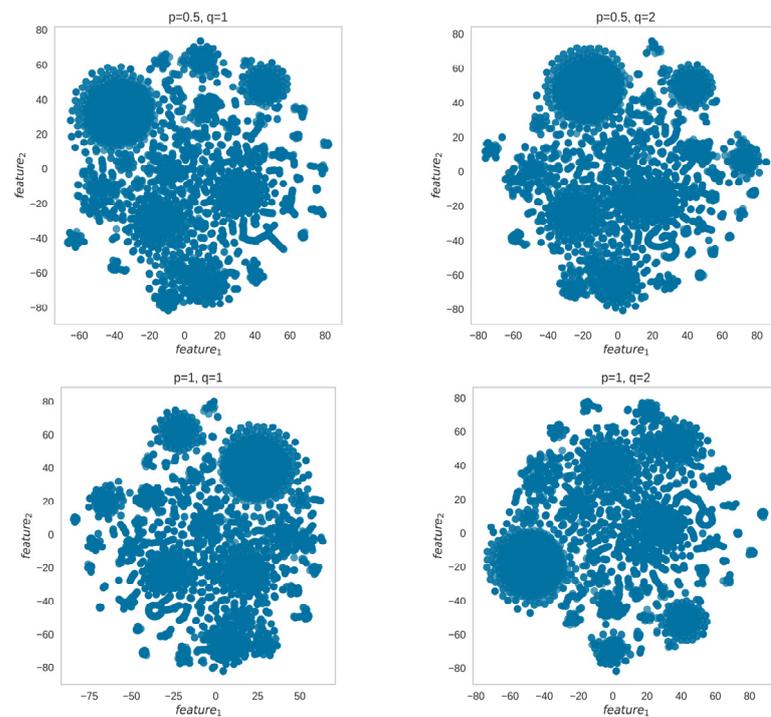


Figure 8. The embeddings learned by Node2Vec on the network shown in Figure 1 contain visible clusters for different parameters. However, the clustering phase with k-means or DBSCAN could not detect clusters with good modularity or average conductance, as shown by the low modularity and high conductance results in Figures 2 and 3 on G_2 .

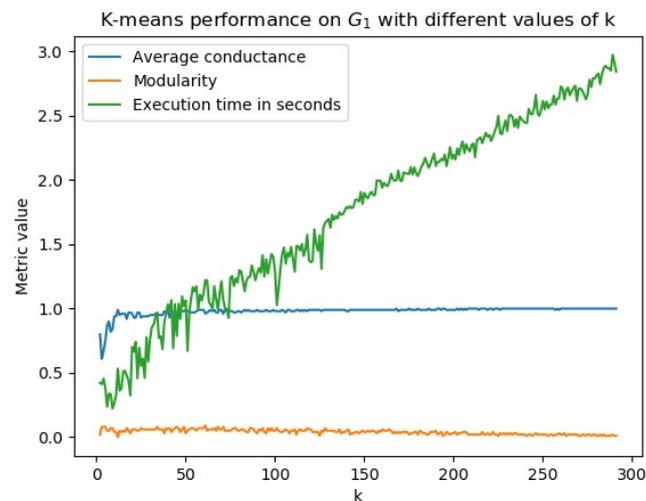


Figure 9. Iterating from $k = 2$ to $k = 292$ (half of the network's total number of nodes of G_1), clusters detected have a modularity close to zero and an average conductance close to 1, which indicates a bad community structure.

The same problem was identified with DBSCAN. By using a grid search that aimed to find the best combinations of its parameters, we could not find any combination that could generate good clusters on bitcoin transaction networks, as shown in Figure 10.

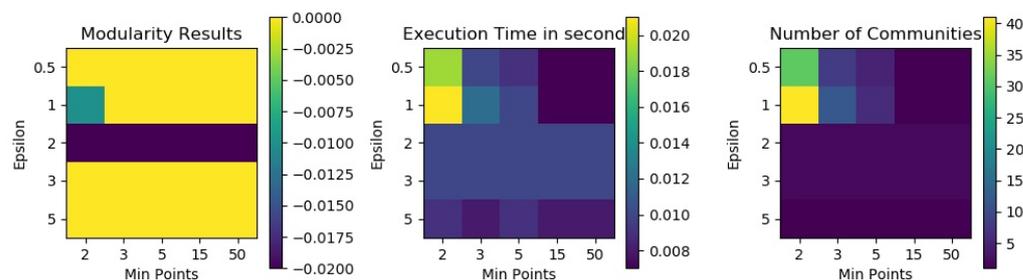


Figure 10. DBSCAN results in terms of modularity, number of detected communities and execution time for different combinations of epsilon and min points on G_1 . All modularity values are very low (zero or less), which indicates a bad community structure.

5. Conclusions

Community detection can be used in blockchain transaction networks to detect illegal communities containing users associated with illegal activities. It becomes easy to identify other users in the community that are likely to be involved in illegal activities based on their interactions in the community. However, the lack of illegal ground-truth communities makes the illegal community detection task challenging in terms of validating the detected communities. We addressed the challenge by using known illegal addresses to label the detected communities. By using 2,313,344 Bitcoin addresses associated with illegal darknet markets and a community detection method selected based on the results of the benchmark, 0.06% of the detected communities on daily transaction networks contain one or more illegal addresses involved in illegal transactions. Illegal transactions include the purchase of illegal products and money laundering.

The results also showed that distance-based clustering methods, specifically k-means and DBSCAN, did not yield good results in our benchmark of different community detection methods on Bitcoin transaction networks. Spectral clustering and network representation learning-based methods are affected depending on the k-means or DBSCAN in their clustering phase. Adapting distance-based clustering methods to improve their performance on disconnected networks would improve their performance on blockchain transaction networks. Obtaining a larger dataset of illegal addresses could help to identify more illegal communities from daily transactions.

Author Contributions: D.K. reviewed the literature, performed the experiments, and wrote the initial draft of the manuscript. C.J.T. conceptualized the study, provided critical feedback, and proofread the manuscript throughout the writing process. M.C. proofread the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Sichuan Science and Technology Program (Grant No. 2023YFH0057), Foreign Talents Program of Ministry of China (Grant No. G2022186003L).

Data Availability Statement: Bitcoin transaction data is publicly available. We used the Blockchain Data API (https://www.blockchain.com/explorer/api/blockchain_api) to get blocks of transactions for a particular date, then parsed the blocks to construct transaction networks. Bitcoin addresses associated with darknet markets were obtained using the Wallet Explorer API (<https://www.walletexplorer.com/>) and validated using data from IKNAIO.

Acknowledgments: We acknowledge the Swiss Government for sponsoring the first author through the Swiss Government Excellence scholarships program. We acknowledge IKNAIO for providing data about Bitcoin addresses in Table 3.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Polge, J.; Robert, J.; Le Traon, Y. Permissioned blockchain frameworks in the industry: A comparison. *ICT Express* **2021**, *7*, 229–233. [CrossRef]
2. Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. Available online: <https://bitcoin.org/bitcoin.pdf> (accessed on 1 October 2022).
3. Kalodner, H.; Möser, M.; Lee, K.; Goldfeder, S.; Plattner, M.; Chator, A.; Narayanan, A. BlockSci: Design and Applications of a Blockchain Analysis Platform. In *Proceedings of the 29th USENIX Security Symposium*; USENIX Association: Berkeley, CA, USA, 2020; pp. 2721–2738.
4. Vallarano, N.; Tessone, C.J.; Squartini, T. Bitcoin Transaction Networks: An Overview of Recent Results. *Front. Phys.* **2020**, *8*, 286. [CrossRef]
5. De Collibus, F.M.; Partida, A.; Piškorec, M.; Tessone, C.J. Heterogeneous Preferential Attachment in Key Ethereum-Based Cryptoassets. *Front. Phys.* **2021**, *9*, 568. [CrossRef]
6. De Collibus, F.M.; Piškorec, M.; Partida, A.; Tessone, C.J. The Structural Role of Smart Contracts and Exchanges in the Centralisation of Ethereum-Based Cryptoassets. *Entropy* **2022**, *24*, 1048. [CrossRef] [PubMed]
7. Meiklejohn, S.; Pomarole, M.; Jordan, G.; Levchenko, K.; McCoy, D.; Voelker, G.M.; Savage, S. A fistful of Bitcoins: Characterizing payments among men with no names. *Commun. ACM* **2016**, *59*, 86–93. [CrossRef]
8. Jourdan, M.; Blandin, S.; Wynter, L.; Deshpande, P. Characterizing Entities in the Bitcoin Blockchain. In *Proceedings of the 2018 IEEE International Conference on Data Mining Workshops (ICDMW)*, Singapore, 17–20 November 2018; pp. 55–62.
9. Zhao, K.; Dong, G.; Bian, D. Detection of Illegal Transactions of Cryptocurrency Based on Mutual Information. *Electronics* **2023**, *12*, 1542. [CrossRef]
10. Turner, A.; Irwin, A.S.M. Bitcoin transactions: A digital discovery of illicit activity on the blockchain. *J. Financial Crime* **2018**, *25*, 109–130. [CrossRef]
11. Nerurkar, P. Illegal activity detection on bitcoin transaction using deep learning. *Soft Comput.* **2023**, *27*, 5503–5520. [CrossRef]
12. Qin, F.; Wu, Y.; Tao, F.; Liu, L.; Shi, L.; Miller, A.J. Multi-input address incremental clustering for the Bitcoin blockchain based on Petri net model analysis. *Digit. Commun. Netw.* **2022**, *8*, 680–686. [CrossRef]
13. Wu, S.X.; Wu, Z.; Chen, S.; Li, G.; Zhang, S. Community Detection in Blockchain Social Networks. *J. Commun. Inf. Netw.* **2021**, *6*, 59–71. [CrossRef]
14. Fortunato, S.; Hric, D. Community detection in networks: A user guide. *Phys. Rep.* **2016**, *659*, 1–44. [CrossRef]
15. Su, X.; Xue, S.; Liu, F.; Wu, J.; Yang, J.; Zhou, C.; Hu, W.; Paris, C.; Nepal, S.; Jin, D.; et al. A Comprehensive Survey on Community Detection With Deep Learning. *IEEE Trans. Neural Networks Learn. Syst.* **2022**. [CrossRef]
16. Chen, Y.; Guo, Y.; Wang, M.; Xu, E.; Xie, H.; Bie, R. Securing IOTA Blockchain Against Tangle Vulnerability by Using Large Deviation Theory. *IEEE Internet Things J.* **2023**. [CrossRef]
17. Chen, Y.; Guo, Y.; Wang, Y.; Bie, R. Toward Prevention of Parasite Chain Attack in IOTA Blockchain Networks by Using Evolutionary Game Model. *Mathematics* **2022**, *10*, 1108. [CrossRef]
18. Girvan, M.; Newman, M.E.J. Community structure in social and biological networks. *Proc. Natl. Acad. Sci. USA* **2002**, *99*, 7821–7826. [CrossRef]
19. Kannan, R.; Vempala, S.; Vetta, A. On clusterings: Good, bad and spectral. *J. ACM* **2004**, *51*, 497–515. [CrossRef]
20. Newman, M.E.J.; Girvan, M. Finding and evaluating community structure in networks. *Phys. Rev. E* **2004**, *69*, 026113. [CrossRef] [PubMed]
21. Lázár, A.; Ábel, D.; Vicsek, T. Modularity measure of networks with overlapping communities. *Europhys. Lett.* **2010**, *90*, 18001. [CrossRef]
22. Janczewski, C. *Two Arrested for Alleged Conspiracy to Launder \$4.5 Billion in Stolen Cryptocurrency*; U.S. Department of Justice: Washington, DC, USA, 2022.
23. Jourdan, M.; Blandin, S.; Wynter, L.; Deshpande, P. A probabilistic model of the bitcoin blockchain. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, Long Beach, CA, USA, 16–17 June 2019.
24. Fischer, J.A.; Palechor, A.; Dell’aglio, D.; Bernstein, A.; Tessone, C.J. The Complex Community Structure of the Bitcoin Address Correspondence Network. *Front. Phys.* **2021**, *9*, 1–16. [CrossRef]
25. Möser, M.; Narayanan, A. Resurrecting Address Clustering in Bitcoin. In *Proceedings of the International Conference on Financial Cryptography and Data Security*, Grand Anse, Grenada, 2–6 May 2022; pp. 386–403. [CrossRef]
26. Javed, M.A.; Younis, M.S.; Latif, S.; Qadir, J.; Baig, A. Community detection in networks: A multidisciplinary review. *J. Netw. Comput. Appl.* **2018**, *108*, 87–111. [CrossRef]
27. MacQueen, J.B. Some Methods for Classification and Analysis of Multivariate Observations. In *Berkeley Symposium on Mathematical Statistics and Probability*; University of California Press: Berkeley, CA, USA, 1967; pp. 281–297.
28. Ester, M.; Kriegel, H.P.; Sander, J.; Xu, X. A Density-based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, Portland, Oregon, USA, 2–4 August 1996; pp. 226–231.
29. Johnson, S.C. Hierarchical clustering schemes. *Psychometrika* **1967**, *32*, 241–254. [CrossRef]
30. He, K.; Sun, Y.; Bindel, D.; Hopcroft, J.; Li, Y. Detecting Overlapping Communities from Local Spectral Subspaces. In *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, Atlantic City, NJ, USA, 14–17 November 2015; pp. 769–774.

31. von Luxburg, U. A tutorial on spectral clustering. *Stat. Comput.* **2007**, *17*, 395–416. [[CrossRef](#)]
32. Andersen, R.; Chung, F.; Lang, K. Local Graph Partitioning using PageRank Vectors. In Proceedings of the IEEE Symposium on Foundations of Computer Science (FOCS), Berkeley, CA, USA, 21–24 October 2006; ACM: New York, NY, USA, 2006; pp. 475–486.
33. Traag, V.A.; Waltman, L.; Van Eck, N.J. From Louvain to Leiden: Guaranteeing well-connected communities. *Sci. Rep.* **2019**, *9*, 5233. [[CrossRef](#)]
34. Blondel, V.D.; Guillaume, J.L.; Lambiotte, R.; Lefebvre, E. Fast Unfolding of Communities in Large Networks. *J. Stat. Mech. Theory Exp.* **2008**, *2008*, P10008. [[CrossRef](#)]
35. Raghavan, U.N.; Albert, R.; Kumara, S. Near linear time algorithm to detect community structures in large-scale networks. *Phys. Rev. E* **2007**, *76*, 036106. [[CrossRef](#)] [[PubMed](#)]
36. Han, J.; Li, W.; Zhao, L.; Su, Z.; Zou, Y.; Deng, W. Community detection in dynamic networks via adaptive label propagation. *PLoS ONE* **2017**, *12*, e0188655. [[CrossRef](#)]
37. Traag, V.A.; Šubelj, L. Large network community detection by fast label propagation. *Sci. Rep.* **2023**, *13*, 2701. [[CrossRef](#)] [[PubMed](#)]
38. Xie, J.; Szymanski, B.K.; Liu, X. SLPA: Uncovering Overlapping Communities in Social Networks via a Speaker-Listener Interaction Dynamic Process. In Proceedings of the 2011 IEEE 11th International Conference on Data Mining Workshops, Vancouver, BC, Canada, 11 December 2011; pp. 344–349.
39. Cordasco, G.; Gargano, L. Community detection via semi-synchronous label propagation algorithms. In Proceedings of the 2010 IEEE International Workshop on Business Applications of Social Network Analysis, Bangalore, India, 15 December 2010.
40. Coscia, M.; Rossetti, G.; Giannotti, F.; Pedreschi, D. DEMON: A Local-First Discovery Method for Overlapping Communities. In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), Beijing China, 12–16 August 2012; pp. 615–623.
41. Palla, G.; Derényi, I.; Farkas, I.; Vicsek, T. Uncovering the overlapping community structure of complex networks in nature and society. *Nature* **2005**, *435*, 814–818. [[CrossRef](#)]
42. Golub, G.; Kahan, W. Calculating the Singular Values and Pseudo-Inverse of a Matrix. *J. Soc. Ind. Appl. Math. Ser. B Numer. Anal.* **1965**, *2*, 205–224. [[CrossRef](#)]
43. Binesh, N.; Rezghi, M. Fuzzy clustering in community detection based on nonnegative matrix factorization with two novel evaluation criteria. *Appl. Soft Comput.* **2018**, *69*, 689–703. [[CrossRef](#)]
44. Tang, J.; Qu, M.; Wang, M.; Zhang, M.; Yan, J.; Mei, Q. LINE: Large-scale Information Network Embedding. In Proceedings of the World Wide Web (WWW), Florence, Italy, 18–22 May 2015; pp. 1067–1077.
45. Perozzi, B.; Al-Rfou, R.; Skiena, S. DeepWalk: Online Learning of Social Representations. In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), New York, NY, USA, 24–27 August 2014; pp. 701–710.
46. Grover, A.; Leskovec, J. Node2Vec: Scalable Feature Learning for Networks. In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), San Francisco, CA, USA, 13–17 August 2016; pp. 855–864.
47. Yang, L.; Cao, X.; He, D.; Wang, C.; Wang, X.; Zhang, W. Modularity Based Community Detection with Deep Learning. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), New York, NY, USA, 9–15 July 2016; pp. 2252–2258.
48. Hamilton, W.L.; Ying, R.; Leskovec, J. Representation Learning on Graphs: Methods and Applications. *IEEE Data Eng. Bull.* **2017**, *40*, 52–74.
49. Kamuhanda, D.; Wang, M.; He, K. Sparse Nonnegative Matrix Factorization for Multiple-Local-Community Detection. *IEEE Trans. Comput. Soc. Syst.* **2020**, *7*, 1220–1233. [[CrossRef](#)]
50. Chiang, W.L.; Liu, X.; Si, S.; Li, Y.; Bengio, S.; Hsieh, C.J. Cluster-GCN: An Efficient Algorithm for Training Deep and Large Graph Convolutional Networks. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Anchorage, AK, USA, 4–8 August 2019; ACM: New York, NY, USA, 2019; pp. 257–266.
51. Yang, J.; Leskovec, J. Overlapping community Detection at Scale: A Nonnegative Matrix Factorization Approach. In Proceedings of the ACM International Conference on Web Search and Data Mining (WSDM), Rome, Italy, 4–8 February 2013; pp. 587–596.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.