2019 IEEE Conference on Communications and Network Security (CNS): Workshops: DLoT: 2nd International Workshop on Distributed Ledger of Things

Deanonymizing Cryptocurrency with Graph Learning: The Promises and Challenges

Anil Gaihre* Santosh Pandey* Hang Liu
*: Equal contribution
University of Massachusetts Lowell

***Abstract*—The world economy is embracing the next generation currency, i.e., cryptocurrencies, which dates back to 2009 when Satoshi Nakamoto made Bitcoin publicly available. Rooted from the nature of decentralization and anonymity of blockchain, the cryptocurrencies have, unfortunately, been leveraged for illicit activities by the criminals. The good news is that typical cryptocurrencies, such as Bitcoin, have to publicly publish their transactions, known as a graph, to retain their ultimate goal of trustless and decentralized transaction verification, which lends law enforcement a means to deanonymizing cryptocurrencies. At meantime, graph learning is an extremely powerful tool to extract the latent features of each vertex in a graph to fulfill various tasks, such as, classifying graph vertices. In this work, we discuss the promises and challenges of exploiting graph learning to deanonymizing cryptocurrencies, which can aid the cyberfighters to circumvent cryptocurrency-based illicit activities.**

Cryptocurrencies [38] are the next generation currency that has already started disrupting mainstream financial systems. Such an emerging economy system, however, is well-suited for illicit activities, mainly stemming from two facts. First, the pseudonymous nature of the cryptocurrencies help hide the criminal users from their real world identifications. Second, not bounded by any international borders, as well as the lacking of typical regulations [48] offers tempting convenience for the criminals to conduct worldwide illegal transactions. As an evidence, a number of such activities [43], [12], [14], [20] has already been reported in Bitcoin and other cryptocurrencies.

The good news is typical cryptocurrencies are invented to support transactions *without* the trusted third parties (i.e. bank) and run in a *decentralized* manner, hence require to *publicly* publish all transactions in a chain of blocks (also referred to as blockchain). In this case, the cyberfighters can take this publicly available data and try to deanonymize the criminals through, mainly, the following four methods [18], such as, direct interacting with the users [37], crawling third party information [47], [4], customizing the Bitcoin client itself to identify the neighbors in the peer-to-peer (P2P) network [11], [29] or through analyzing of the transaction graphs [35], [44].

Given the former three attempts are either labor intensive or blockchain user dependent, graph analysis [18], [31], [33], [32], [30], [26], [19], [50], [25] become the most promising mechanism to deanonymizing cryptocurrencies. Before discussing this approach, we briefly describe how to construct the transaction graphs as follows. The publicly available hashes of the transaction and addresses can be mapped into vertices and edges with which several existing toolsets [18] can construct an address-transaction bipartite graph.

Traditional graph-based approaches mainly fall into two categories. First, shrinking the cryptocurrency address space in order to facilitate an easier process of deanonymization. Basically, Bitcoin and other cryptocurrencies (i.e., Altcoins) allow a user to possess multiple addresses to receive the currency from a transaction. This increases the searching space of identifying a real world user. To cope with this concern, various kinds of heuristics [17], [35] have been used to merge multiple addresses to the same user. For instance, the multi-input [35] heuristics assumes the addresses that are used as input to the transaction belong to the single user[1].Second, clustering unknown addresses to tagged ones for deanonymization [16]. Toward that end, existing work [18], [23], [36], [42], [24], [6] exploits the following traditional graph features for vertex classification.

1) Degree (i.e. in-degree or out-degree) of each vertex.
2) Currency amount accumulated in the address.
3) Holding time of the currency amount.
4) The series of active usage timestamps.
5) Influx and efflux of the currencies of an address.

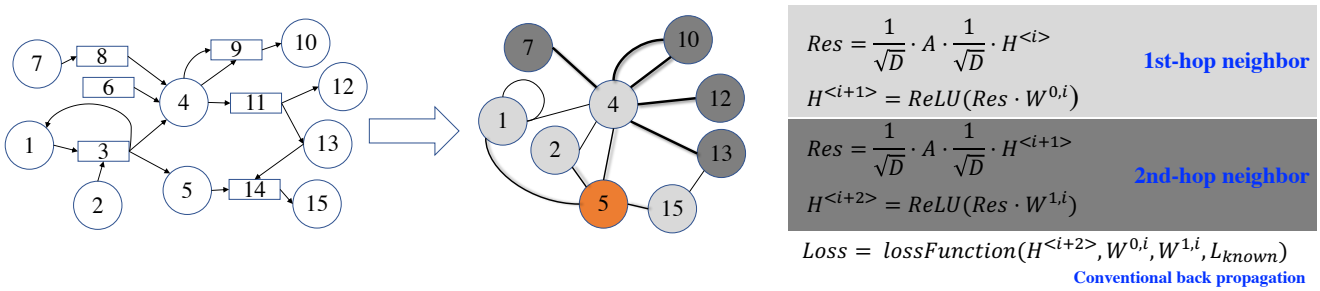
For instance, an address with moderate degree but with very high accumulation of Bitcoin is likely to be cold wallet address of big organizations [18]. Further, [36] uses K-means clustering detect fraudulent activities in Bitcoin. Likewise, other methods of learning like Support Vector Machine (SVM) and Mahalanobis Distance [42], [24] can be used to detect anomaly in Bitcoin graph. [23] works on training its model from already existing clusters from chain analysis tool [6] and detects newer clusters which are not yet identified using Gradient Boosting Classifier.

Despite those aforementioned features are easy to retrieve, they might fall short for cyberfighters who require high accuracy in anomaly detection. Graph embedding is proven to be an increasingly popular paradigm toward more accurate feature extraction for graph vertices [13]. Mathematically, an embedding for a graph G is a representation of G on a compact, connected 2-manifold surface [7].

To the best of our knowledge, there exist three graph embedding construction methods, i.e., matrix factorization, random walk and Graph Convolutional Network (GCN) [21]. In particular, matrix factorization [28], [9], [45], [10], [39]

[1]It should be noted that this particular heuristic may not work for some transactions like mixing.

978-1-5386-7117-7/19/$31.00 ©2019 IEEE



(a). Bitcoin bi-partite graph (b). Bitcoin address interaction graph (c). Graph Convolutional Network

Fig. 1: Applying GCN [27] to Bitcoin graph, particularly, for vertex 5 in (b). In (c), A, D, H and W stand for the graph in (b), degree of each vertex, the embedding of each vertex and the weight of the fully connected neural network. L_{known} represent the known label (such as criminal address) of a set of known vertices.

represents the graph in the form of an adjacency matrix and use factorization to extract the embeddings of that graph. When we can only partially observe a graph or if the graph is too large, random walk can be used for generating the embeddings. In particular, this option [34], [40], [22], [15], [41] exploits a stochastic process which defines a path consisting of successive space in the graph. Third, deep learning based graph embedding computation is recently gaining popularity. Briefly, GCN [27] proposes a graph defined neural network. In particular, it defines a convolution operator on the graph which iteratively aggregates the embeddings of neighbors for a node which is scalable and more efficient. It was evident that a randomly initiated 2-layer GCN is able to produce useful features of nodes in a graph [27].

Figure 1 briefly explains how GCN works on a Bitcoin transaction graph. For simplicity, we adopt the traditional convention [18] to convert the Bitcoin graph as shown in Figure 1(a) (where circle and rectangle represent address and transaction vertices, respectively) into an undirected graph as shown in the Figure 1(b). Note, graph learning can also accommodate directed graphs. The essence of GCN falls into two steps: First, it gathers the information of the first hop neighbors to compute the embedding for the vertex of interest. To involving more topological information, one can gather and compute the embedding with multiple hops of neighbors, as shown in Figure 1(c). Second, GCN exploits the labels of known vertices to update the weight of neural network (often known as back propagation). Figure 1(b) and 1(c) illustrate a 2-hop neighbor based embedding construction. In particular, at first iteration, node 5 gathers and computes the embedding based upon 5's 1st-hop neighbors, i.e.,{1, 2, 4, 15}. To involve more topological information, this example also considers the 2nd-hop of neighbors, i.e., {7, 10, 12, 13}.

Considering GCN is more efficient than matrix factorization and random walk based approaches [21], as well as coming with a more popular community support and ecosystem, we choose the GCN based embedding construction for crytocurrency denonymization. However, the cryptocurrencies, such as, Bitcoin graphs, do possess the following unique challenges that ask for particular treatment.

1) Large and *extremely* skewed graph: The cryptocurrency graph consists of a large volume of vertices and presents absurdly skewed degree distributions – often more skewed than social networks. Without address clustering [18] reports the number of vertices in Bitcoin graph to be more than 720 million vertices with 1.6 billion edges. With the majority of the edges binds to very few exchange center addresses, gaming addresses and miner addresses, the graph is highly skewed from degree ranging from 1 to 13 million of address nodes on June 2018 [8]. Both the high volume of vertices and skewed degree distribution add challenges for state-of-the-art graph learning systems.
2) Dynamically *increasing* graph: The cryptocurrency transaction graph are increasing and append-only in nature. This would likely cause the continuous increase in the size of the graph demanding more computational and memory resources. Also, one should be able to exploit the temporal information to avoid repeated computations.
3) Semantic graph [49], [46]: A user might want to use the cryptocurrency as a distributed storage or encode smart contract in the transactions. These kind of activities retrofit semantic to the cryptocurrencies and immediately complicate the process of graph learning.

While the aforementioned hardships are challenging, there also exist properties that are friendly to GCN:

1) Inactive/zero balance addresses: Many of the addresses in Bitcoin are inactive. Identifying and removing those addresses will help reduce the problem size. As a consequence, GCN will experience reduced workload, subsequently, faster embedding computations.
2) Publicly available address labels: Thanks to the popularity of cryptocurrencies, an array of websites [5], [2], [1], [3] are hosting the real world identities of the addresses. These labels could be used to facilitate faster GCN learning and training.

In summary, graph learning is a promising tool for deanonymizing cryptocurrencies but with mounting challenges in the computing horizon. Toward societal benefits, future cyber law enforcements and researchers shall invest in this direction.

REFERENCES

[1] Bitcoin discussion forum. Available at https://forum.bitcoin.com/bitcoin-discussion//.

[2] Bitcoin forum. Available at https://bitcointalk.org/index.php?board=1.0.

[3] Bitcoin investing forum. Available at https://www.investing.com/crypto/bitcoin/chat/.

[4] Bitiodine github. Available at https://github.com/mikispag/bitiodine.

[5] Blockchain.com. Available at https://www.blockchain.com/charts/market-price?timespan=all&showDataPoints=true.

[6] Chain analysis. Available at https://www.chainalysis.com/. Accessed 21 Mar. 2019.

[7] En.wikipedia.org *Graph embedding*. Available at: https://en.wikipedia.org/wiki/Graph_embedding. Accessed 21 Mar. 2019.

[8] Rusty blockparser github repository. Available at https://github.com/JoinMarket-Org/joinmarket.

[9] AHMED, A., SHERVASHIDZE, N., NARAYANAMURTHY, S., JOSIFOVSKI, V., AND SMOLA, A. J. Distributed large-scale natural graph factorization. In *Proceedings of the 22nd International World Wide Web Conference (WWW 2013)* (2013).

[10] BELKIN, M., AND NIYOGI, P. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic* (Cambridge, MA, USA, 2001), NIPS'01, MIT Press, pp. 585–591.

[11] BIRYUKOV, A., KHOVRATOVICH, D., AND PUSTOGAROV, I. Deanonymisation of clients in bitcoin p2p network. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security* (2014), ACM, pp. 15–29.

[12] BRYANS, D. Bitcoin and money laundering: mining for an effective solution. *Ind. LJ 89* (2014), 441.

[13] CAI, H., ZHENG, V. W., AND CHANG, K. C.-C. A comprehensive survey of graph embedding: Problems, techniques, and applications. *IEEE Transactions on Knowledge and Data Engineering 30* (2018), 1616–1637.

[14] CARNEGIE, AND CYLAB. Traveling the silk road : A measurement analysis of a large anonymous online marketplace.

[15] CHEN, H., PEROZZI, B., HU, Y., AND SKIENA, S. Harp: Hierarchical representation learning for networks. In *AAAI* (2018).

[16] EBERLE, W., AND HOLDER, L. Anomaly detection in data represented as graphs. *Intelligent Data Analysis 11*, 6 (2007), 663–689.

[17] ERMILOV, D., PANOV, M., AND YANOVICH, Y. Automatic bitcoin address clustering. In *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)* (2017), IEEE, pp. 461–466.

[18] GAIHRE, A., LUO, Y., AND LIU, H. Do bitcoin users really care about anonymity? an analysis of the bitcoin transaction graph. In *2018 IEEE International Conference on Big Data (Big Data)* (2018), IEEE, pp. 1198–1207.

[19] GAIHRE, A., WU, Z., YAO, F., AND LIU, H. Exploring runtime optimizations to accelerate breadth-first search on gpus. In *Proceedings of the 28th International Symposium on High-Performance Parallel and Distributed Computing* (2019), ACM.

[20] GIBBS, S. Child abuse imagery found within bitcoin's blockchain. https://www.theguardian.com/technology/2018/mar/20/child-abuse-imagery-bitcoin-blockchain-illegal-content.

[21] GOYAL, P., AND FERRARA, E. Graph embedding techniques, applications, and performance: A survey. *Knowledge-Based Systems 151* (2018), 78–94.

[22] GROVER, A., AND LESKOVEC, J. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining* (2016), ACM, pp. 855–864.

[23] HARLEV, M. A., SUN YIN, H., LANGENHELDT, K. C., MUKKAMALA, R., AND VATRAPU, R. 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* (2018).

[24] HIRSHMAN, J., HUANG, Y., AND MACKE, S. Unsupervised approaches to detecting anomalous behavior in the bitcoin transaction network. *3rd ed. Technical report, Stanford University* (2013).

[25] HU, Y., LIU, H., AND HUANG, H. H. High-performance triangle counting on gpus. In *2018 IEEE High Performance extreme Computing Conference (HPEC)* (2018), IEEE, pp. 1–5.

[26] JI, Y., LIU, H., AND HUANG, H. H. ispan: Parallel identification of strongly connected components with spanning trees. In *SC18: International Conference for High Performance Computing, Networking, Storage and Analysis* (2018), IEEE, pp. 731–742.

[27] KIPF, T. N., AND WELLING, M. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).

[28] KOREN, Y., BELL, R., AND VOLINSKY, C. Matrix factorization techniques for recommender systems. *Computer*, 8 (2009), 30–37.

[29] KOSHY, P. Coinseer: A telescope into bitcoin.

[30] LIU, H., HU, Y., AND HUANG, H. H. Tricore: Scalable triangle counting on gpus. In *High Performance Computing, Networking, Storage and Analysis, 2015 SC-International Conference for* (2018).

[31] LIU, H., AND HUANG, H. H. Enterprise: breadth-first graph traversal on gpus. In *High Performance Computing, Networking, Storage and Analysis, 2015 SC-International Conference for* (2015), IEEE, pp. 1–12.

[32] LIU, H., AND HUANG, H. H. Graphene: fine-grained io management for graph computing. In *Proceedings of the 15th Usenix Conference on File and Storage Technologies* (2017), USENIX Association, pp. 285–299.

[33] LIU, H., HUANG, H. H., AND HU, Y. ibfs: Concurrent breadth-first search on gpus. In *Proceedings of the 2016 International Conference on Management of Data* (2016), ACM, pp. 403–416.

[34] LOVSZ, L. Random walks on graphs: A survey, 1993.

[35] MEIKLEJOHN, S., POMAROLE, M., JORDAN, G., LEVCHENKO, K., MCCOY, D., VOELKER, G. M., AND SAVAGE, S. A fistful of bitcoins: characterizing payments among men with no names. In *Proceedings of the 2013 conference on Internet measurement conference* (2013), ACM, pp. 127–140.

[36] MONAMO, P., MARIVATE, V., AND TWALA, B. Unsupervised learning for robust bitcoin fraud detection. In *2016 Information Security for South Africa (ISSA)* (2016), IEEE, pp. 129–134.

[37] MOSER, M., BOHME, R., AND BREUKER, D. An inquiry into money laundering tools in the bitcoin ecosystem. In *eCrime Researchers Summit (eCRS), 2013* (2013), IEEE, pp. 1–14.

[38] NAKAMOTO, S. Bitcoin: A peer-to-peer electronic cash system.

[39] OU, M., CUI, P., PEI, J., ZHANG, Z., AND ZHU, W. Asymmetric transitivity preserving graph embedding. In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, NY, USA, 2016), KDD '16, ACM, pp. 1105–1114.

[40] PEROZZI, B., AL-RFOU, R., AND SKIENA, S. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, NY, USA, 2014), KDD '14, ACM, pp. 701–710.

[41] PEROZZI, B., KULKARNI, V., AND SKIENA, S. Walklets: Multi-scale graph embeddings for interpretable network classification. *CoRR abs/1605.02115* (2016).

[42] PHAM, T., AND LEE, S. Anomaly detection in bitcoin network using unsupervised learning methods. *arXiv preprint arXiv:1611.03941* (2016).

[43] PORTNOFF, R. S., HUANG, D. Y., DOERFLER, P., AFROZ, S., AND MCCOY, D. Backpage and bitcoin: Uncovering human traffickers. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2017), ACM, pp. 1595–1604.

[44] REID, F., AND HARRIGAN, M. An analysis of anonymity in the bitcoin system. In *Security and privacy in social networks*. Springer, 2013, pp. 197–223.

[45] ROWEIS, S. T., AND SAUL, L. K. Nonlinear dimensionality reduction by locally linear embedding. *Science 290*, 5500 (2000), 2323–2326.

[46] SHIRRIFF, K. Hidden surprises in the bitcoin blockchain and how they are stored. http://www.righto.com/2014/02/ascii-bernanke-wikileaks-photographs.html.

[47] SPAGNUOLO, M., MAGGI, F., AND ZANERO, S. Bitiodine: Extracting intelligence from the bitcoin network. In *International Conference on Financial Cryptography and Data Security* (2014), Springer, pp. 457–468.

[48] STANKOVIC, S. An introductory guide to cryptocurrency regulation. Available at https://unblock.net/cryptocurrency-regulation/. Accessed 21 Mar. 2019.

[49] SWARD, A., VECNA, I., AND STONEDAHL, F. Data insertion in bitcoin's blockchain. *Ledger 3* (2018).

[50] YAN, D., AND LIU, H. Parallel graph processing. *Encyclopedia of Big Data Technologies* (2018), 1–8.