Generation and Classification of Illicit Bitcoin Transactions

Pablo de Juan Fidalgo(✉), Carmen Cámara, and Pedro Peris-Lopez

Universidad Carlos III de Madrid, Madrid, Spain
pablodejuan99@gmail.com

Abstract. Financial fraud is an everyday problem that banking institutions have to face. With the disruption of Bitcoin as a new model which relies on decentralisation and anonymity, attackers have taken advantage of this monetary system. It allows them to obtain funds from illegal activities such as ransomware payments and hide them. At the same time, Law Enforcement Agencies use open-source data to apply network forensics to Blockchain data. The analysis is usually performed by using artificial intelligence. Unfortunately, the current situation shows a scarcity of high-quality data sets to train the detection algorithms. This work tries to overcome this barrier with significant contributions. With nearly 25,000 illicit transactions, we have increased the Elliptic Data Set –the most extensive labelled transaction data publicly available in any cryptocurrency. The former data set only contained 4,545 illicit transactions, resulting in a class imbalance of 9.8:90.2 illicit/licit ratio. Our work has changed that to a 41.2:58.8 illicit/licit ratio. Besides, to show that class imbalance datasets can also be beaten with artificial work, we have studied the use of generative adversarial networks (GAN) for creating synthetic samples. Finally, the last part of this work was dedicated to applying deep learning and, more particularly, long short-term memory networks (LSTM) for the binary classification problem. We show ideal results that can help change the current state-of-the-art trend, mainly focused on machine learning algorithms.

Keywords: Bitcoin · Anti-money laundering · Data imbalance · Deep learning · Generative adversarial networks · Long short-term memory networks

1 Introduction and Background

Since the origin of the monetary system, people have tried different tricks to evade the network's numerous controls. Banking entities are usually behind the endless amount of methods that attackers implement during their illegal activities. This is why we can observe a vast effort from these institutions against these practices. Although detecting illicit transactions is not easy, the way the network is implemented helps track these movements. A centralised system will always be a better approach considering that the governance depends only on one entity.

J. Bravo et al. (Eds.): UCAmI 2022, LNNS 594, pp. 1086–1097, 2023.
https://doi.org/10.1007/978-3-031-21333-5_108

In 2008, Satoshi Nakamoto [16] presented Bitcoin together with Blockchain. The white paper explained deeply the creation of a peer-to-peer (P2P) network that will support the different transactions of this new cryptocurrency. There were some key points behind this concept. The first one is that cryptography would be the central element, providing security to the network and to the nodes that participate along it. Apart from it, as we are dealing with a P2P network, there will not be any central authority in charge of monitoring the different transactions. These transactions will be stored in blocks that will be chained one after the other. The protocol proposes a consensus between the nodes to avoid attacks on the network. That is, the nodes can check through the computation of hashes if a block is correct or not. Bitcoin started to gain popularity, and more users were joining the network. Some of this attraction came because of the anonymity that masks the users. It is essential to mention that although Blockchain technology has some measures to avoid fraudulent behaviour, it cannot prevent fraud itself. An undetermined number of Bitcoin transactions is known to hide an illegal action. There are many illicit actions such as scams, malware, terrorist organisations, ransomware, Ponzi schemes, etc.

From a few years ago to the present day, researchers have been trying to develop new artificial intelligence algorithms that allow detecting Bitcoin addresses and transactions under these suspicious activities. Unfortunately, the community is facing some problems with this challenge. On the one hand, there are few labelled Bitcoin data sets available. Without data that can help us learn from these behaviours, we will not be able to detect if something is going wrong or not. On the other hand, due to the nature of the system, the vast majority of the transactions are benign. This leads to imbalanced data, with the minority class being critical.

This work aims to reduce the current gap that Blockchain's forensics researchers are dealing with by contributing nearly 25,000 new illicit transactions to one of the most extensive data sets on the Internet. Apart from that, a synthetic generation approach through Generative Adversarial Networks (GANs) is also studied. Finally, the current state of the art mainly focuses on classification through Machine Learning techniques. Instead, this paper tries to solve this problem with Deep Learning, particularly with Long Short-Term Memory Networks (LSTM).

Bitcoin network has two characteristics that usually do not match together. The nodes that work under the protocol are anonymous. Their identifier consists of an alphanumerical string of 26 to 32 characters long. This address by itself cannot be linked to your personal identity. If we take this premise, we can observe why criminal groups like Bitcoin's anonymity. Despite this network feature, something that comes along with Bitcoin is that it is open and public to the whole Internet. This means that anyone can inspect the latest blocks uploaded to the network, the different transactions inside each block, or how the funds have been distributed through the addresses involved.

Although Bitcoin is a P2P network, many users with bitcoins in their wallets join Bitcoin thanks to exchanges, which act as centralised points in the architecture. These exchanges work in the same way as traditional exchanges that help you interchange money from one currency to another, e.g., US dollars to Euros. The only difference is that, in this case, we are interchanging fiduciary money that a central banking entity has distributed to a cryptocurrency such as bitcoins, ether (from the Ethereum network), etc.

These entities work under Know-Your-Customer (KYC) policies. This means that before you can exchange your fiduciary money, you must fill in some data that identifies yourself. Of course, this is not the only way to enter Bitcoin's network, as there is also a parallel option with the P2P market. In fact, this is the one that most criminals follow.

1.1 Dealing with Scarcity of Labelled Data

Artificial Intelligence is the primary tool involved in the detection of abnormal behaviours. It can be seen in different fields, such as in bio-medics for skin cancer detection or cybersecurity for developing Intrusion Detection Systems (IDS). The critical part of the process is having a high-quality data set with labelled data, particularly the class that wants to be detected, in our case, illicit transactions.

After reviewing different research papers, like [4,9,19,21], we collected more than 13,500 Bitcoin addresses from the published datasets linked to illicit activities such as ransomware campaigns and Ponzi schemes. The downside is that we still lacked a proper dataset with features.

1.2 Elliptic Data Set

Thankfully, Elliptic Data Set [22] appeared along the way. This resulted from research that gathered professionals from IBM, MIT and Elliptic. Regarding Elliptic [1], it is a cryptocurrency intelligence company focused on safeguarding crypto-currency ecosystems from criminal activity.

The data set provides more than 200,000 transactions that belong to licit categories (exchanges, wallet providers, miners, licit services, etc.) versus illicit ones (scams, malware, terrorist organisations, ransomware, Ponzi schemes, etc.). They are located in a period of 49 weeks. The transactions are classified into three different categories. Two percent (4,545) are labelled as class1 (illicit), twenty-one percent (42,019) are labelled class2 (licit). The rest are not labelled concerning licit versus illicit but have other features. This imbalance in the labelled data was one of the greatest challenges to overcome.

Regarding the features linked to each transaction, we can observe that there are 166 features. The first 94 features represent local information about the transaction. The remaining 72 features, called aggregated features, are obtained by aggregating transaction information one-hop back-ward/forward from the centre node. Despite having many features, a detailed description of them cannot be provided due to Elliptic's intellectual property policies. This lack of information made it more challenging to analyse the problem.

2 Balancing the Data Set

2.1 Natural Generation of Data

Working with a heavily unbalanced data set can create a strong bias in our results. Considering that during our initial steps to find a suitable dataset, we collected more than 13,500 illicit Bitcoin addresses, we launched an effort to combine both pieces of information. In parallel, it was found that a user from Kaggle deanonymised 99.5 per cent of Elliptic transactions [5]. The transaction ID from the Elliptic dataset was directly linked to the corresponding Bitcoin hash. The reader can consult several works [6,20] for further information in deanonymization analysis. Using this information, we could observe that the 49-time steps corresponded to an interval of two weeks (i.e., from the 1st of January, 2016, to the 2nd of October, 2017). With this restriction, a Python script[1] was developed to retrieve the transactions from the Bitcoin addresses in that period. The script consisted of a piece of code that made queries through Blockstream API [7] to the Bitcoin blockchain and only stored the hash of the transactions of those wallets if they were located in that date interval. After the execution of the script, we ended up with 24,947 transactions. Then we compared the gathered transactions to those already in the Elliptic data set and luckily, any of them was repeated. This means that our research provides five times more transactions than the ones that were before. Apart from giving more labelled data, this significant contribution[2] aims to reduce the gap between the anomaly class and the normal one, which is always a critical part. At this moment, labelled data is split in the following way:

- Illicit transactions correspond to 41.24 percent (29,492 out of 71,511 labelled data)
- Licit transactions correspond to 58.76 percent (42,01 out of 71,511 labelled data)

2.2 Synthetic Generation of Data

Another common approach when dealing with unbalanced data is to apply some transformations to the data set. In particular, this is usually used with data corresponding to images. Some data augmentation techniques are flipping, rotation, scaling, translation, cropping or applying Gaussian noise. After combining these techniques with a small dataset, we would end up with a large data set.

There are different approaches to fixing the imbalance problem. The first one is undersampling which consists in removing some samples from the majority class. On the other hand, we have oversampling, which increases the minority class with artificial samples. Some works such as [11,23,26] propose different techniques which are more or less sophisticated. In our case, as we are dealing with time-series data, we need high-quality algorithms because of the nature of

[1] https://github.com/PabDJ/IllicitBitcoinTransactions.

[2] https://www.kaggle.com/datasets/pablodejuanfidalgo/augmented-elliptic-data-set.

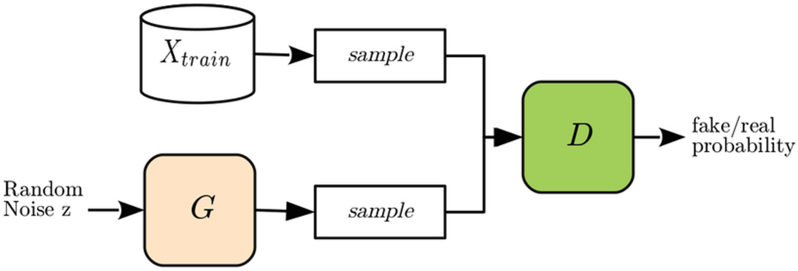

Fig. 1. Generative Adversarial Network Model Architecture.

the dataset. After considering the different options, we thought the best approach was to use generative adversarial networks.

Generative Adversarial Networks. The concept of Generative Adversarial Networks was first introduced in 2014 [13]. It consists of two deep generative models that work under a competition (see Fig. 1). On the one hand, we have a generative model, Generator, which generates new samples according to the input data distribution. On the other hand, we have a discriminative model, Discriminator. It analyses the data created by the Generator and, depending on different parameters, decides whether it is original or fake. As this network is connected, the feedback from one model is learnt by the other and vice-versa. This competition between neural networks leads to the production of high-quality data.

TGAN represents the architecture that was studied in [24]. It stands for Tabular Generative Adversarial Network and has shown optimal results when working with data sets similar to the Elliptic Data Set. In particular, the work [10] presented data augmentation with the well-known Credit Card Fraud Detection Data Set. This specific data set was composed of two classes: the majority was 99.83 percent of the whole dataset, while the minority class was only a poor 0.17 percent. After training the network, the generated data could be relied on as much as the original data for future experiments.

In our experiments, the number of samples generated was 24,947. This number corresponds to the same amount of illicit transactions collected in Subsect. 2.1. Therefore we can perform a fair comparison between original and artificial transactions (see Sect. 4).

3 Classification of Illicit Transactions

The task of deciding whether a transaction is licit or illicit is critical. But suppose we dive deeper into the consequences of an incorrect decision. In that case, we can agree that a false negative, i.e., to label an illicit transaction as licit, is worse

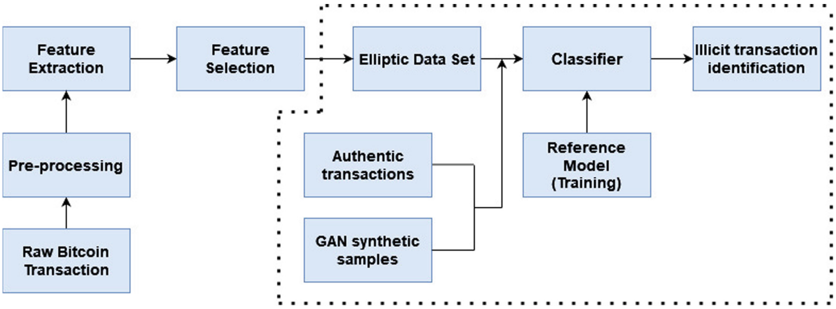

Fig. 2. Classical machine learning workflow.

than a false positive. This way, our model aims to reduce the number of false negatives as much as possible without significantly increasing the number of false positives.

Binary classification is one of the ancient tasks carried out in Artificial Intelligence. This action needs the existence of a training data set that allows the model to map the input data into specific labels. Regarding Artificial Intelligence, there are different ways of approaching the classification problem. The most common is machine learning. This statement can be proved by looking at the enormous amount of scientific literature that performs machine learning experiments [3,14,15,17] for binary classification. Nevertheless, some new innovative researches rely on deep learning, such as [25]. In this case, a Recurrent Neural Network (RNN) learning model for hunting cryptocurrency malware threats is proposed.

Machine learning is the art of designing computers with the ability to learn from statistics without explicitly programming them. On the other hand, considering deep learning as a subset of machine learning, the main difference is that deep learning models try to replicate human brain structure as much as possible. This is a layered structure of algorithms called Artificial Neural Networks (ANN). The building of this system can be divided into input, hidden, and output layers. Lastly, machine learning usually requires less computing power and deep learning typically needs less ongoing human intervention.

Another critical topic is deciding how the learning process is performed. Supervised machine learning is the technique that consumes labelled data. The model is fed with data, and with the help of the previously labelled samples, the system produces an outcome. On the opposite side, we can find unsupervised machine learning, which does not require labels.

Figure 2 shows the typical structure of a machine learning problem. The area surrounded by the dotted lines specifies the work of this paper.

Once we got a good overview of the landscape, we decided to test machine learning methods and afterwards perform an experiment with deep learning for a comparison between both techniques.

3.1 Machine Learning for Classification

Machine Learning is a branch of artificial intelligence which is exponentially growing. The focal point behind the concept relies on using algorithms modelled to make classification or prediction tasks. It has some dependence on human beings because machine learning needs a set of features that allows it to know the commonalities and differences inside the samples being analysed.

The learning process can be divided into three parts: decision, error function and model optimisation. We can find a wide variety of machine learning algorithms in the literature. Our research has focused on two of them, random forest and logistic regression.

Random forest (RF) is one of the most popular machine learning algorithms. The algorithm is made of a collection of trees which act individually. Each of them returns a prediction, which is voted and the one that gets most of the votes becomes the model's prediction. The power of this mechanism is that the ability of the trees to work together as a group outperforms the work of each individual tree. Logistic Regression (LR) is also a widely used classification technique in machine learning. It uses a logistic function to model the dichotomous dependent variable. In our experiments, there are only two possible classes (licit o illicit transactions). We urge the reader to consult [13] for additional details on these algorithms.

3.2 Deep Learning for Classification

As previously introduced, deep learning stands out by the ability to replicate the structure and functionality of the human brain through artificial neural networks. One key difference between machine learning and deep learning is that we can train the network with more and more samples and still increase the performance; meanwhile, in the former one, we usually reach a plateau. Another notable difference is that machine learning needs a supervised process in which the features are extracted (feature extraction algorithm). In contrast, the network implicitly does this process in deep learning.

Regarding the network structure, we can observe that different layers of neurons form it. The first and last layers are named visible layers; the rest are hidden layers. The computation can follow different paths. If the layer in charge of processing the data only feeds the next layer, we call forward propagation. In the case that the previous layers are also affected, such as recalculating the weights to adjust the model, we call it backpropagation. Mixing both techniques is what allows the algorithm to reach better performances. There are also some learning methods such as learning rate decay, transfer learning or dropout. The last one is crucial to fighting against overfitting.

We have used Long Short-Term Memory (LSTM) network in this work. LSTM networks are a type of Recurrent Neural Network (RNN). One of the critical points of RNN is that they have loops that enable the learning process to persist over time instead of only maintaining information from the last iterations. LSTMs overcome the vanishing gradient problem through blocks of memory that are connected and that performs better than the typical neurons. Thanks to LSTM architecture, in particular, due to the mechanism of gates that forms the network, we can obtain great results when working with time-series data, as in this case. The interested reader can consult [12] for a detailed introduction to LSTMs.

4 Experiments

This section will show the different experiments carried out and compare them to the results obtained in the original implementation. First, we will start with machine learning algorithms such as Random Forest and Logistic Regression; then, we will move to deep learning with LSTM.

Although the original paper divides the implementation of these algorithms depending on the features that the model consumed, we only tested the case in which all features are employed. That is, we use both local features and aggregated features linked to the neighbour transactions. The data set was not preprocessed because it was already done. We tried to replicate the configuration given by the original implementation as much as possible. The split between training and test data followed a 70:30 ratio. Regarding the classification models, Logistic Regression was employed with default parameters. Secondly, Random Forest was implemented with the following parameters: n_estimators = 50, max_depth = 100 and max_features = 50.

For the implementation of LSTM, we used Keras [2], which is the API from TensorFlow for building and training deep learning models. The network consisted of an LSTM layer with 166 neurons (one per feature). After that, a Dropout layer was added with the parameter set to 0.2, and finally, a Dense layer with sigmoid function as activation. The model was compiled with binary cross-entropy as the loss function, and the optimiser was set to Adam. The number of epochs was 1,000, and the batch size was 32. Precision, Recall and F1 Score are the metrics used to assess performance.

4.1 Results

In this section, we will show and discuss the results linked to the different experiments that have been carried out.

Table 1 shows the ones related to the machine learning experiments. Elliptic RF (Random Forest) and Elliptic LR (Logistic Regression) rows collect the results from the original paper. They have been included in the table for a better visual comparison. Then, for each machine learning algorithm, we performed two experiments. We wanted to analyse the performance of the two models with the augmented data. Firstly, “natural tx” represents the transactions gathered by researching the different data sets of Bitcoin addresses. Secondly, “synthetic tx” is the synthetic transactions obtained with the TGAN experiment. We can observe that both experiments outperform the original ones in all the metrics. These results point out the necessity of balanced data sets for machine learning.

Table 1. Machine learning results.

Method	Precision	Recall	F1
Elliptic RF [22]	0.956	0.670	0.788
RF with natural tx	0.985	0.962	0.974
RF with synthetic tx	0.999	0.983	0.991
Elliptic LR	0.404	0.593	0.481
LR with natural tx	0.784	0.824	0.804
LR with synthetic tx	0.951	0.961	0.956

Table 2. Deep learning results.

Method	Precision	Recall	F1
Elliptic GCN	0.812	0.512	0.628
Elliptic Skip-GCN	0.812	0.623	0.705
Elliptic EvolveGCN	0.850	0.624	0.720
LSTM with Elliptic data	0.908	0.855	0.868
LSTM with natural tx	0.947	0.927	0.934
LSTM with synthetic tx	0.991	0.981	0.985

Between natural and artificial generation, we can observe that the algorithm fed with synthetic data obtains better results. The explanation can be in the data quality, as the features from those artificial samples might not differ too much between them, provoking an overfitting problem. In the end, the most realistic scenario is the one that works with ground-truth data. Looking at both algorithms, we can appreciate that Random Forest does not show such a big difference as the observed between the two experiments using Logistic Regression.

Table 2 is focused on deep learning. The first three rows were obtained from the original implementation and are included for comparison purposes. We want to show that Long Short-Term Networks outperform the different variants of Graph Convolutional Networks (GCN). More precisely, the weakest experiment (LSTM with Elliptic data), done with the unbalanced data set, outperformed the whole set of GCN. A performance improvement can be observed when LSTM works with a more balanced data set. These are the cases of LSTM with natural tx and LSTM with synthetic tx. As mentioned before, the samples generated with the aid of TGAN create better results than the ones with natural transactions; the difference is not as significant as the one established between LSTM and GCN.

This section wants to expose two main strengths. The first one shows the importance of balanced data sets. We hope our work helps to reduce the current gap with a new dataset and the possibility of generating ourselves one. The results show the impact on the performance of all the algorithms tested. The second one presents deep learning as an already working alternative. Machine learning, in particular Random Forest, has been chosen for an endless list of binary classification problems, achieving excellent performance. Still, we have shown how LSTM networks can achieve excellent results with time-series data.

5 Conclusions and Future Work

In summary, novel techniques for the generation and classification of illicit Bitcoin transactions have been proposed in this paper. First of all, after detecting the scarcity of ground truth Bitcoin data sets, which are essential for feeding supervised machine learning algorithms, we found nearly 25,000 illicit transactions linked to more than 13,500 Bitcoin addresses related to ransomware payments and Ponzi schemes. Our mission was to solve this class imbalance problem which usually ends up with poor classification results. This contribution means that Elliptic Data Set changed from 4,545 illicit transactions to 29,492 illicit transactions and that class imbalance went from 9.8:90.2 to 41.2:58.8 illicit/licit ratio.

Apart from that, we also have studied how Generative Adversarial Networks can lead the generation of synthetic samples accurately. This approach can reduce the gap between unbalanced data sets, especially when dealing with time-series data, as in our case. The results achieved in this work after the generation of new samples outperform the ones published by the original paper, which were used as a baseline.

To classify the transactions into the two classes, we observed that state-of-the-art continues developing machine learning models for binary classification. Regarding the nature of our data set, we present LSTM Networks as an example of how deep learning, particularly RNNs, can also be an excellent option for this kind of task. It offers the same performance as the Random Forest, which is close to ideal.

In future work, there are additional tasks to enhance the results presented in this paper. The first one is related to the dataset itself. Although we can work without knowing what is represented behind each future, as a black box, we understand that reversing each of them, or at least the ones related to the transaction itself, i.e., the first 94 features, would give a better approach for analysing the problem. The second one is related to the hyperparameters. Although they were chosen looking for the best results, we were limited by the machine's computational power. A better environment, linked to a hyperparameter tuning would give even better results. Finally, a GAN architecture has been studied, such as TGAN. We also considered implementing WGAN, which uses "critic" as an alternative to the discriminator. This approach has shown good performance in tabular data such as the Credit Card Fraud Data Set [8,18]. We would be able to generate more artificial data with this new model and compare the performance of the different classifiers with more instances.

Acknowledgment. This work was supported by the Spanish Ministry of Science, Innovation and Universities grant PID2019-111429RBC21(ODIO); and by the Comunidad de Madrid (Spain) under the projects PUCFA (PUCFA-CM-UC3M) and

CYNAMON (P2018/TCS-4566)–cofinanced by European Structural Funds (ESF and FEDER).

We want to thank Claudio Bellei, Head of Data Science at Elliptic, for providing us the features related to the illicit transactions that we collected.

References

1. Elliptic: blockchain analytics amp; crypto compliance solutions. https://www.elliptic.co/
2. Implementation of the keras API, the high-level API of tensorflow. https://www.tensorflow.org/api_docs/python/tf/keras
3. Alarab, I., Prakoonwit, S., Nacer, M.I.: Comparative analysis using supervised learning methods for anti-money laundering in bitcoin. In: Proceedings of the 2020 5th International Conference on Machine Learning Technologies, pp. 11-17. ICMLT 2020, Association for Computing Machinery, New York, NY, USA (2020). https://doi.org/10.1145/3409073.3409078
4. Bartoletti, M., Pes, B., Serusi, S.: Data mining for detecting bitcoin Ponzi schemes. In: 2018 Crypto Valley Conference on Blockchain Technology (CVCBT), pp. 75–84 (2018). https://doi.org/10.1109/CVCBT.2018.00014
5. Benzik: Deanonymized 99.5 PCT of elliptic transactions (2019). https://www.kaggle.com/datasets/alexbenzik/deanonymized-995-pct-of-elliptic-transactions
6. Biryukov, A., Tikhomirov, S.: Deanonymization and linkability of cryptocurrency transactions based on network analysis. In: 2019 IEEE European Symposium on Security and Privacy (EuroSP), pp. 172–184 (2019). https://doi.org/10.1109/EuroSP.2019.00022
7. Blockstream: esplora HTTP API. https://github.com/Blockstream/esplora/blob/master/API.md
8. Clemente, F.: How to generate synthetic tabular data? Wasserstein loss for generative adversarial networks (2020). https://towardsdatascience.com/how-to-generate-synthetic-tabular-data-bcde7c28038a
9. Conti, M., Gangwal, A., Ruj, S.: On the economic significance of ransomware campaigns: a bitcoin transactions perspective. Comput. Secur. **79**, 162–189 (2018). https://doi.org/10.1016/j.cose.2018.08.008
10. Dutta, G.: Fixing imbalance dataset using tGAN (2021). https://www.kaggle.com/code/gauravduttakiit/fixing-imbalance-dataset-using-tgan
11. Feldman, E.V., Ruchay, A.N., Matveeva, V.K., Samsonova, V.D.: Bitcoin abnormal transaction detection based on machine learning. In: van der Aalst, W.M.P., et al. (eds.) Recent Trends in Analysis of Images, Social Networks and Texts, pp. 205–215. Springer International Publishing, Cham (2021)
12. Foster, D.: Generative Deep Learning. O'Reilly Media, Sebastopol (2019)
13. Rebala, G., Ravi, A., Churiwala, S.: An Introduction to Machine Learning. Springer, Cham (2019)
14. Lorenz, J., Silva, M.I., Aparício, D., Ascensão, J.T., Bizarro, P.: Machine learning methods to detect money laundering in the bitcoin blockchain in the presence of label scarcity. In: Proceedings of the First ACM International Conference on AI in Finance, pp. 1–8 (2020)
15. Monamo, P.M., Marivate, V., Twala, B.: A multifaceted approach to bitcoin fraud detection: global and local outliers. In: 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA), pp. 188–194 (2016). https://doi.org/10.1109/ICMLA.2016.0039

16. Nakamoto, S.: Bitcoin: A peer-to-peer electronic cash system. http://bitcoin.org/bitcoin.pdf (2008)
17. Nerurkar, P., Bhirud, S., Ludinard, R., Busnel, Y., Kumari, S.: Supervised learning model for identifying illegal activities in bitcoin. Appl. Intell. **51**, 1–20 (2021). https://doi.org/10.1007/s10489-020-02048-w
18. Pandey, A., Bhatt, D.L., Bhowmik, T.: Limitations and applicability of GANs in banking domain. In: ADGN@ECAI (2020)
19. Paquet-Clouston, M., Haslhofer, B., Dupont, B.: Ransomware payments in the bitcoin ecosystem. J. Cybersecur. **5**(1), tyz003 (2019)
20. Ron, D., Shamir, A.: Quantitative analysis of the full bitcoin transaction graph. In: Sadeghi, A.R. (ed.) Financial Cryptography and Data Security, pp. 6–24. Springer, Berlin Heidelberg, Berlin, Heidelberg (2013)
21. van de Voort, J., Coneys, S.: Classifying bitcoin ponzi schemes with machine learning (2018). https://github.com/seanconeys/Bitcoin_Ponzi_ml/blob/master/FinalPaper_PonziClassification.pdf
22. Weber, M., et al.: anti-money laundering in bitcoin: experimenting with graph convolutional networks for financial forensics. arXiv preprint arXiv:1908.02591 (2019)
23. Wen, Q., et al.: Time series data augmentation for deep learning: a survey. arXiv preprint arXiv:2002.12478 (2020)
24. Xu, L., Veeramachaneni, K.: Synthesizing tabular data using generative adversarial networks. arXiv preprint arXiv:1811.11264 (2018)
25. Yazdinejad, A., HaddadPajouh, H., Dehghantanha, A., Parizi, R.M., Srivastava, G., Chen, M.Y.: Cryptocurrency malware hunting: a deep recurrent neural network approach. Appl. Soft. Comput. **96**, 106630 (2020) https://doi.org/10.1016/j.asoc.2020.106630, https://www.sciencedirect.com/science/article/pii/S1568494620305688
26. Zola, F., Segurola-Gil, L., Bruse, J., Galar, M., Orduna-Urrutia, R.: Attacking bitcoin anonymity: generative adversarial networks for improving bitcoin entity classification. Appl. Intell. (2022). https://doi.org/10.1007/s10489-022-03378-7