Tutorial: Graph-based Management and Mining of Blockchain Data

Arijit Khan and Cuneyt Gurcan Akcora

AALBORG UNIVERSITY

University of Manitoba

Tutorial Outline

1) **Introduction**
   1.1 Blockchain Components
   1.2 Blockchain Data Structures and Storage
   1.3 Blockchain Categories

2) **Data Extraction and Analysis Tools**

3) **Graphs Constructed**
   3.1 UTXO (unspent transaction output)-based
   3.2 Account-based

4) **Graph Machine Learning on Blockchain Graphs**
   4.1 Graph Analysis on Blockchain Graphs
   4.2 Topological Data Analysis on Blockchain Graphs
   4.3 Machine Learning on Blockchain Graphs

5) **Target Applications of Blockchain Data Analysis**

6) **Open Problems**

Core Blockchain

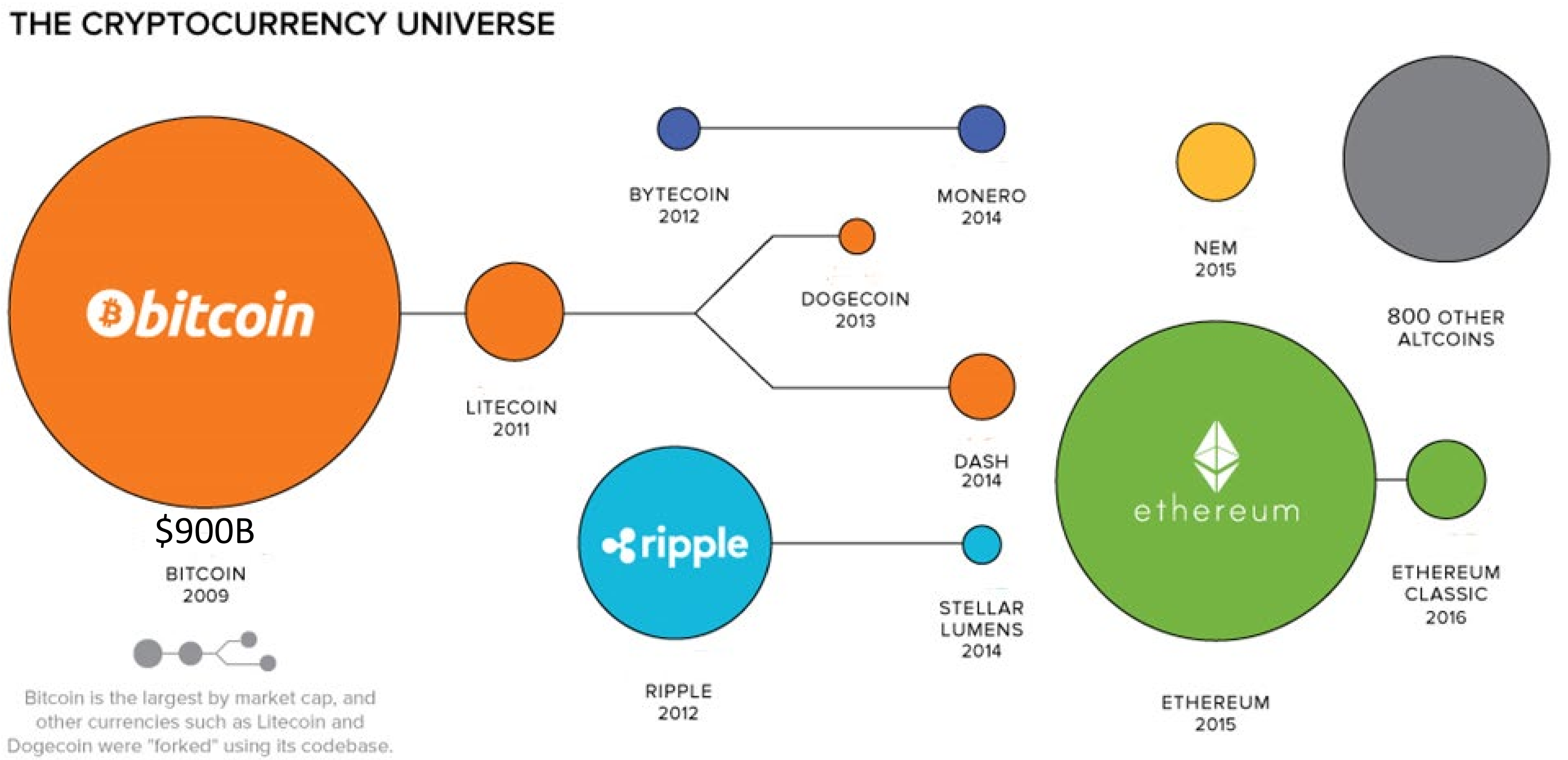
10/31/2008: Satoshi Nakamoto posted the Bitcoin white paper to a forum.

1/3/2009: The first data block in the Bitcoin.

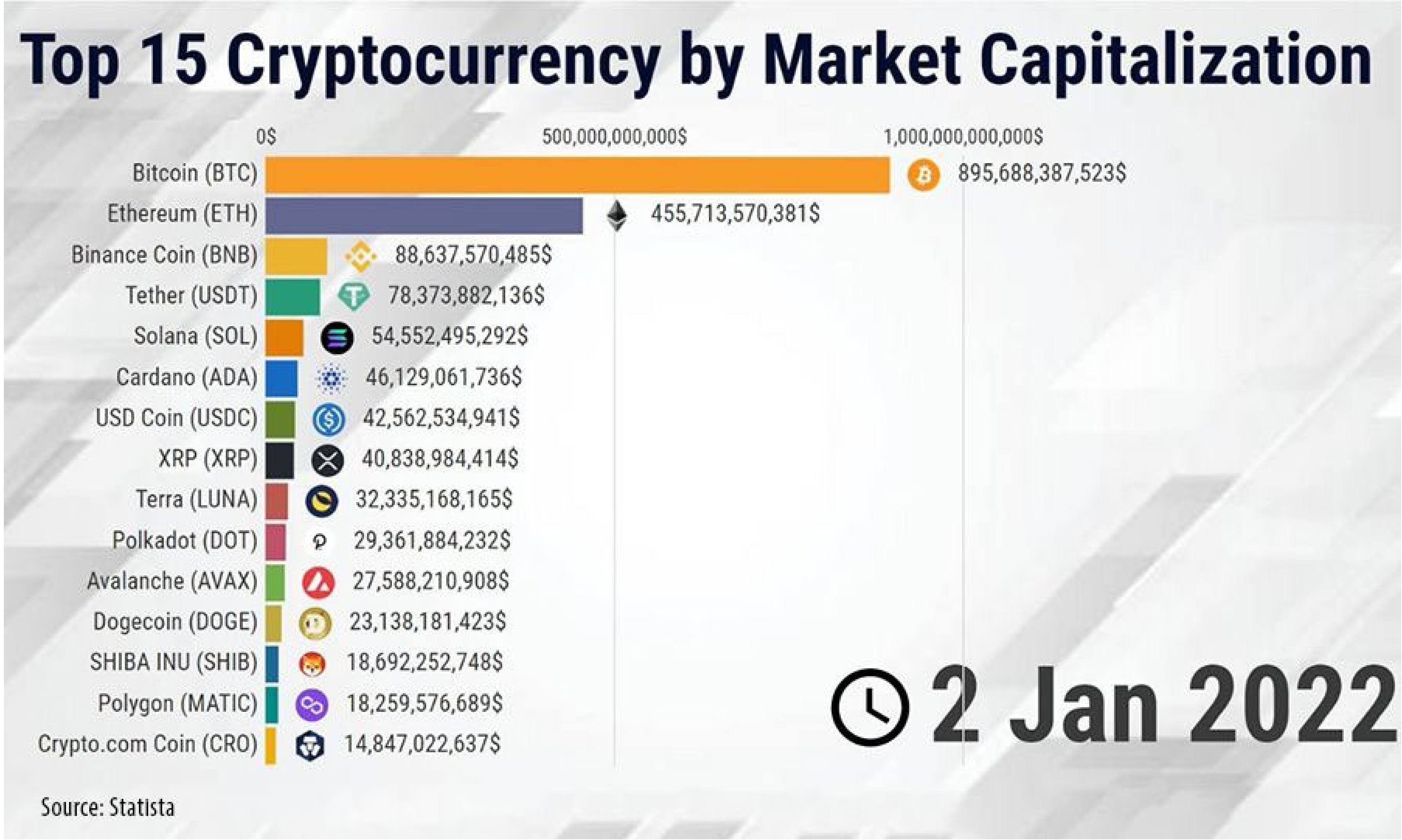
Bitcoin

Ethereum Litecoin Ripple Neo

Dash Iota Nem Zcash



* By JEFF DESJARDINS. Image retrieved from VisualCapitalist.com and updated.



Source: https://statisticsanddata.org/data/top-15-cryptocurrency-by-market-capitalization-and-price-2013-2021/

Summary of Features of top 5 Blockchain Platforms for Enterprises

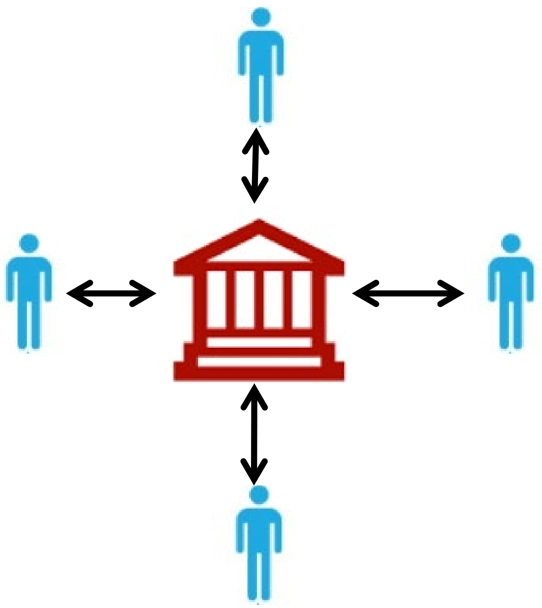
	Ethereum	Hyperledger Fabric	R3 Corda	Ripple	Quorum
Industry-focus	Cross-industry	Cross-industry	Financial Services	Financial Services	Cross-industry
Governance	Ethereum developers	Linux Foundation	R3 Consortium	Ripple Labs	Ethereum developers & JP Morgan Chase
Ledger type	Permissionless	Permissioned	Permissioned	Permissioned	Permissioned
Cryptocurrency	Ether (ETH)	None	None	Ripple (XRP)	None
% providers with experience[1]	93%	93%	60%	33%	27%
% share of engagements[2]	52%	12%	13%	4%	10%
Coin Market Cap[3]	$91.5 B (18%)	Not applicable	Not Applicable	$43.9 B (9%)	Not Applicable
Consensus algorithm	Proof of Work (PoW)	Pluggable framework	Pluggable framework	Probabilistic voting	Majority voting
Smart contract functionality	Yes	Yes	Yes	No	Yes

1. *Based on responses from 15 leading blockchain service providers*
2. *Based on a random sample of set of 50 enterprise blockchain engagements across multiple industries*
3. *Coinmarketcap.com as of Feb 20, 2018, 6:20 PM UTC*

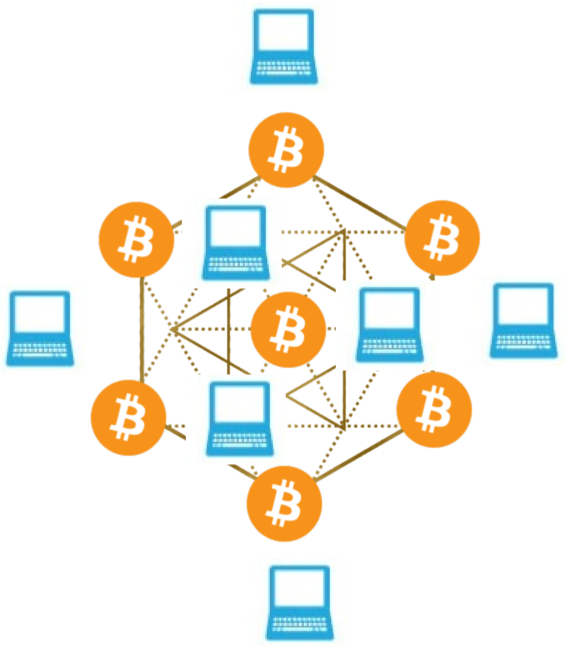
Source: HfS Research, 2018

Source: https://www.hfsresearch.com/blockchain/top-5-blockchain-platforms_031618/

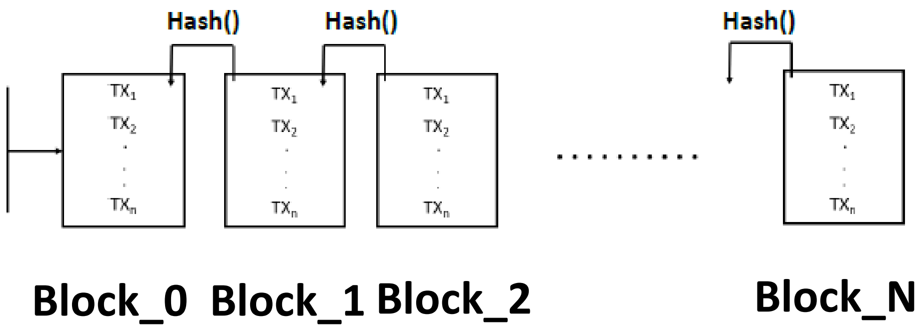
Blockchain: Introduction



Traditional approach: controlled by a central and trusted third-party, e.g., a bank.



Blockchain approach: each participant in a peer-to-peer network has a copy of the database, ensuring immutability.



Blockchain: A distributed, digital ledger of records (transactions) stored in a sequential order.

Each block contains the hash of the previous block.

The blocks are shared openly among its participants to create an immutable sequence of transactions.

Blockchain is updated by consensus among its users (open or controlled set).

Blockchain Consensus: Proof-of-Work

- Proof-of-work is done by miners, who compete to create new blocks with the latest transactions.
- The work (i.e., the computation) is reasonably hard (yet feasible) for the prover (miner), but is easy to check for the verifier (other users).
- The competition is won by the one whose computer can solve a math puzzle in proof-of-work the fastest -- this generates the cryptographic link between the current block and the previous block.
- The winning miner shares the new block with the rest of the network and earns some reward (newly minted cryptocurrency).
- Miners join the longest chain to resolve forks in blockchain.

Proof-of-X

- Proof-of-Work is energy expensive, difficult to scale.

- With trust in participants (permissioned setting), consensus costs can be reduced.

- Proof-of-X is an umbrella term that covers Proof-of-Work alternatives in block mining.

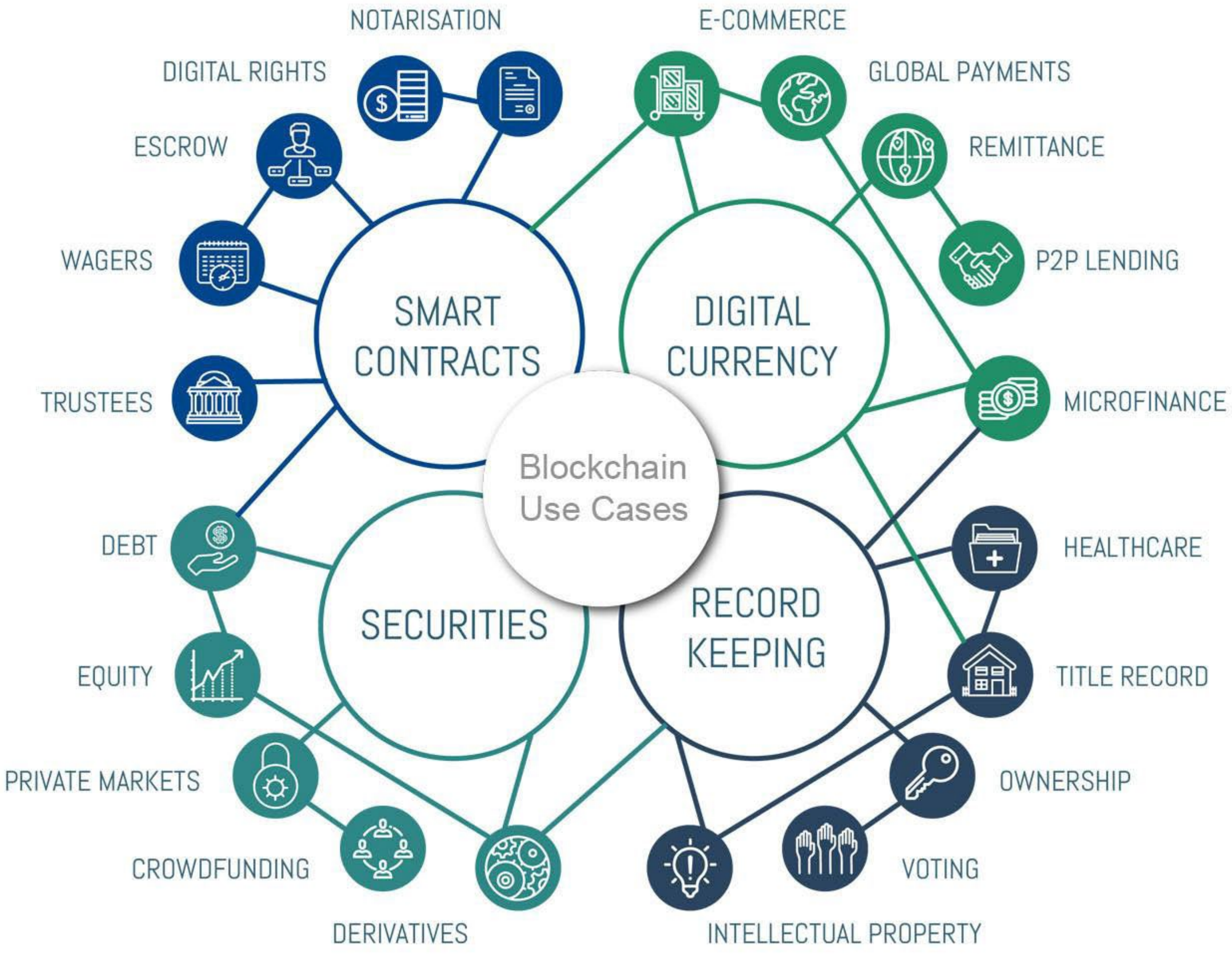
- Each alternative scheme expects miners to show a proof that they have done enough work or spent enough wealth before creating the block.

- **Proof-of-Stake**: Stake = Coin × Age. The miner with the highest stake becomes the next miner in the chain. Once coins are used, their age becomes zero.

- In September 2022, Ethereum made the transition from a proof-of-work system to a proof-of-stake system.

- Proof-of-Burn, Delegated-Proof-of-Stake, Memory-hard Proof-of-Work, Proof-of-Ownership, Proof-of-Publication, …

Applications of Blockchains



Survey: H. Huang, W. Kong, S. Zhou, Z. Zheng, and S. Guo. 2021. **A Survey of state-of-the-art on blockchains: theories, modelings, and tools**. ACM Computing Surveys 54, 2 (2021), 44:1–44:42.

Source: https://hellosergio.medium.com/6-emerging-categories-for-blockchain-use-cases-4650f824d130

Public, Private, and Permissioned Blockchains

- **Public blockchains** are open to any user to join and participate.

- **Private blockchains** have a central controlling authority, usually the company behind the blockchain. Participants are chosen by the authority with protected access modes.

- **Permissioned, or consortium, blockchains** are one or more entities, e.g., a group of companies that can be in charge of the access control. These "administrator" nodes grant different access modes to participating nodes, depending on business requirements.

Our Focus: Public, Permissionless Blockchains

- Public permissionless blockchains allow access to trusted, transparent, comprehensive, and granular datasets of digital economic behaviors.

- Blockchain data analytics, also called the distributed ledger analytics (DLA), is an emerging field of research (**Financial data mining**).

Blockchain Data Analytics

- Data stored in a public blockchain can be considered as **big data.**

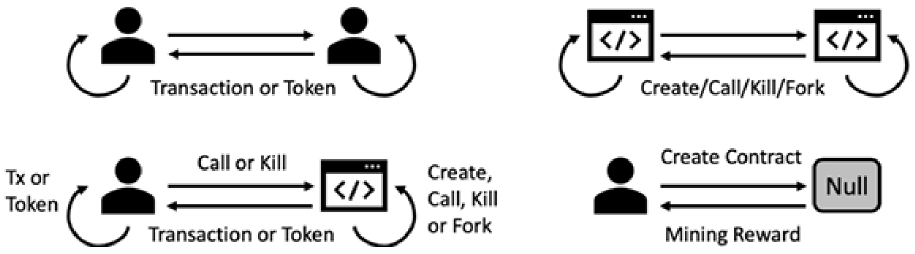
- **Volume:** Ethereum archive nodes that store a complete snapshot of the Ethereum blockchain, including all the transaction records, take up to **4TB of space.**

  https://decrypt.co/24779/ethereum-archive-nodes-now-take-up-4-terabytes-of-space

- **Velocity:** Ethereum blockchain has processed more than **1.1 million transactions per day** in July 2021.

  https://www.statista.com/statistics/730838/number-of-daily-cryptocurrency-transactions-by-type/

- **Veracity:** Ethereum contains a vast number of **heterogeneous interactions**, e.g., user-to-user, user-to-contract, contract-to-user, and contract-to-contract across multiple layers via external and internal transactions, ether, tokens, dAapps, etc.



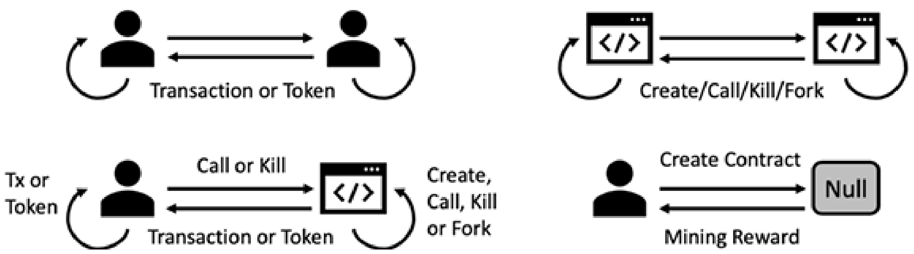
Interactions in the Ethereum Blockchain Network

Graph-based Blockchain Data Analytics

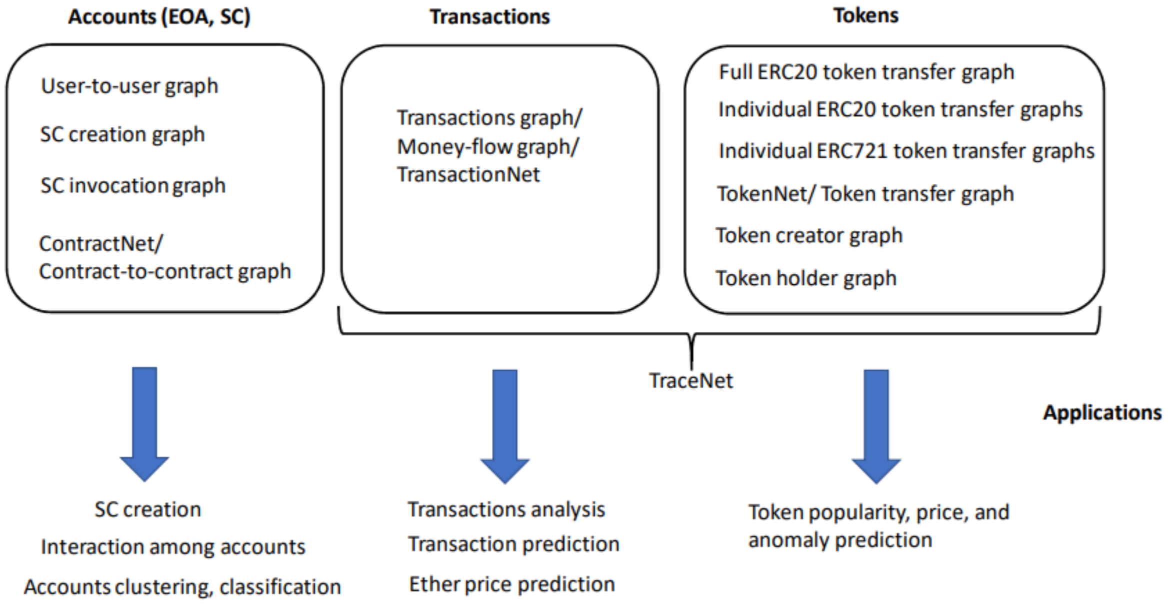
- Data stored in a public blockchain such as in Ethereum can be considered as **big data.**

- **Data analytic methods** can be applied to extract knowledge hidden in the blockchain.

- Several recent research works performed **graph analysis** on the publicly available blockchain data to reveal insights into its transactions and for important downstream tasks, e.g., **cryptocurrency price prediction**, **address clustering**, **phishing scams**, and **counterfeit tokens detection**.



Interactions in the Ethereum Blockchain Network



Various graphs created from interactions between accounts, transactions, token transfers; as well as their common applications

This Tutorial Is NOT About …

- **Applications of blockchains.**

  Related survey: H. Huang, W. Kong, S. Zhou, Z. Zheng, and S. Guo. 2021. **A survey of state-of-the-art on blockchains: theories, modelings, and tools.** ACM Comput. Surv. 54, 2 (2021), 44:1–44:42.

- **Distributed databases aspects of blockchains** , e.g., consensus protocols, confidentiality, fault-tolerance, scalability, blockchain systems, and production deployment.

  Related tutorials/ articles:

  M. J. Amiri, D. Agrawal, and A. E. Abbadi. 2021. **Permissioned blockchains: properties, techniques and applications**. In SIGMOD.

  S. Gupta, J. Hellings, S. Rahnama, and M. Sadoghi. 2020. **Building high throughput permissioned blockchain fabrics: challenges and opportunities.** PVLDB 13, 12 (2020), 3441–3444.

  S. Maiyya, V. Zakhary, M. J. Amiri, D. Agrawal, and A. E. Abbadi. 2019. **Database and distributed computing foundations of blockchains**. In SIGMOD.

  C. Mohan. 2019. **State of public and private blockchains: myths and reality**. In SIGMOD.

- **Security and privacy on blockchains.**

  Related survey: R. Zhang, R. Xue, and L. Liu: **Security and privacy on blockchain**. ACM Comput. Surv. 52(3): 51:1-51:34 (2019).

Relevant Tutorials

- C. Akcora, M. Kantarcioglu, Y. R. Gel. **Data science on blockchains**. KDD 2021
- C. Akcora, M. Kantarcioglu, Y. R. Gel. **Data science on blockchains**. SDM 2021
- C. Akcora, M. Kantarcioglu, Y. R. Gel. **Data science on blockchains**. ICDE 2020
- C. Akcora, M. Kantarcioglu, Y. R. Gel. **Blockchain data analytics**. ICDM 2018

These tutorials covered fundamental building blocks of blockchains and data structures of UTXO and account blockchains.

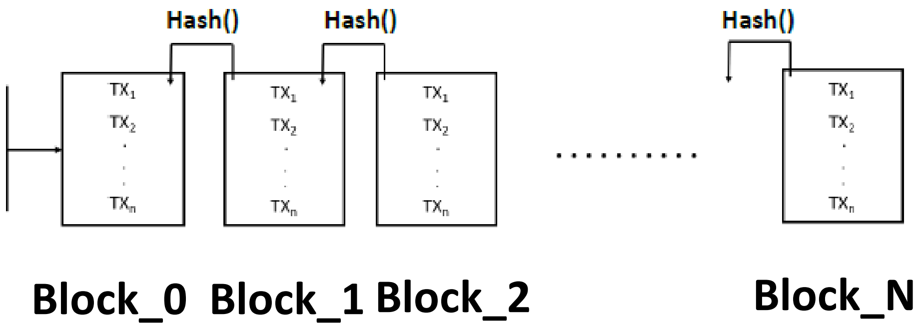
Unlike ours, these tutorials do not cover blockchain graph models, data extraction and analysis, state-of-the-art in graph analysis, topological data analysis, and graph machine learning for blockchain data.

Relevant Surveys

- Jiajing Wu, Jieli Liu, Yijing Zhao, Zibin Zheng. **Analysis of cryptocurrency transactions from a network perspective: an overview**. J. Netw. Comput. Appl. 190: 103139 (2021).
- F. Victor, P. Ruppel, A. Küpper. **A taxonomy for distributed ledger analytics**. Computer 54(2): 30-38 (2021).
- A. Kamišalić and R. Kramberger and I. Fister. **Synergy of blockchain technology and data mining techniques for anomaly detection**. Appl. Sciences 11:17 (2021).
- C. Akcora, Y. R. Gel, and M. Kantarcioglu**. Blockchain networks: data structures of Bitcoin, Monero, Zcash, Ethereum, Ripple, and Iota**. WIREs Data Mining Knowl. Discov. 12, 1 (2022).
- A. Khan. **Graph Analysis of the Ethereum blockchain data: a survey of datasets, techniques, and future direction**. In IEEE International Conference on Blockchain 2022.

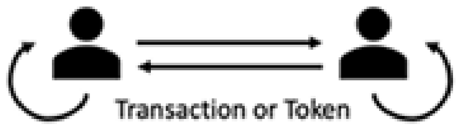
Blockchain Components

- **Ledger:** A ledger is a series (or chain) of blocks on which transaction details are recorded after suitable authentication and verification by the designated network participants.



- **Cryptocurrencies:** A cryptocurrency is a medium of exchange, that is digital and uses encryption techniques to control the creation of monetary units and to verify the transfer of funds.

- **Transactions:** . A transaction is a transfer of assets (e.g., cryptocurrencies, tokens) from one address to another.



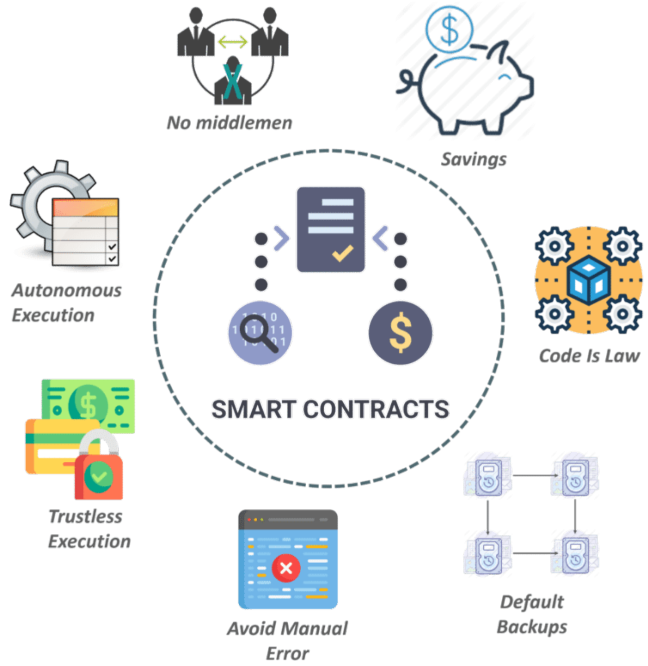
Blockchain Components

- **Smart Contracts:** A smart contract is deployed to a specific address on the blockchain and constitutes a collection of code (for multiple functions) and data (its state). Smart contracts can define rules and automatically enforce them via the code. User accounts interact with a smart contract by transactions that execute a function defined on the contract. Smart contracts can also call (or, kill) each other, even itself, if processing a transaction requires some functionality within the other or in the same contract.

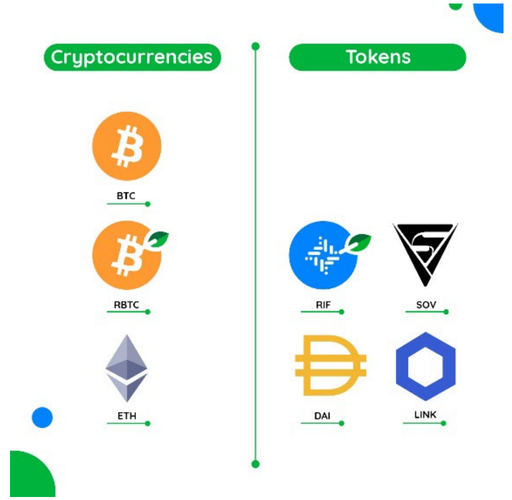
  - Smart contracts were first proposed in 1994 by Nick Szabo, who coined the term, referring to "a set of promises, specified in digital form, including protocols within which the parties perform on these promises".
  - Ethereum implemented a Turing-complete language on its blockchain, supporting smart contracts (2015).
  - Smart contracts introduced by Ethereum are fundamental building blocks for decentralized finance (DeFi) and NFT applications.

- **Tokens:** Tokens are digital assets or access rights provided by their issuers, managed by smart contracts and the blockchain platform. A token's smart contract specifies meta-attributes about the token, including its symbol, total supply, decimals, etc.

  - Two most popular token standards on Ethereum are: **(1) ERC20,** a standard interface for fungible (interchangeable) tokens, such as voting tokens, staking tokens, or virtual currencies, -- widely used in initial coin offering (ICO); and **(2) ERC721,** a standard interface for non-fungible tokens (NFTs), e.g., a deed for a song or an artwork.



Source: https://www.edureka.co/blog/smart-contracts/



Source: https://developers.rsk.co/guides/get-crypto-on-rsk/cryptocurrency-vs-token/

Blockchain Components

- **dApps:** A decentralized application (dapp) is built on a decentralized peer-to-peer network that combines smart contract(s) as backend and a frontend user interface, generally implemented via HTML5, CSS, and web3.js.
  - In Ethereum, about 70% dapps have only one smart contract, and 90% dapps have less than three smart contracts, while there are also some dapps having more than 100 smart contracts.
  - A dapp author may even include a smart contract written by others.
  - Exchanges, wallet, and gamesare the most popular dApp categories.

- **DeFi:** DeFi, or decentralized finance, are dApps for financial products and services, e.g., loans, savings, insurance, exchanges, liquidity, lenders, and trading, powered by decentralized blockchain technologies such as Ethereum. DeFi protocols are autonomous programs (i.e., smart contracts) that constitute a collection of rules similar to physical financial institutions.

- **Stablecoins:** Stablecoins are cryptocurrencies, whose value is pegged, or tied, to that of another currency, commodity or financial instrument, e.g., Tether (USDT) and TrueUSD (TUSD) are popular stablecoins backed by U.S. dollar, TerraUSD (UST) algorithmic stablecoin.

K. Wu, **An empirical study of blockchain-based decentralized applications**, ArXiV, 2019.
C. R. Harvey, A. Ramachandran, and J. Santoro, **DeFi and the future of finance**. John Wiley & Sons, 2021.
S. Kitzler, F. Victor, P. Saggese, and B. Haslhofer, **Disentangling decentralized finance (DeFi) Compositions**, ArXiV, 2021.

Blockchains: Data Structures, Storage and Categories

Private and Public Blockchains

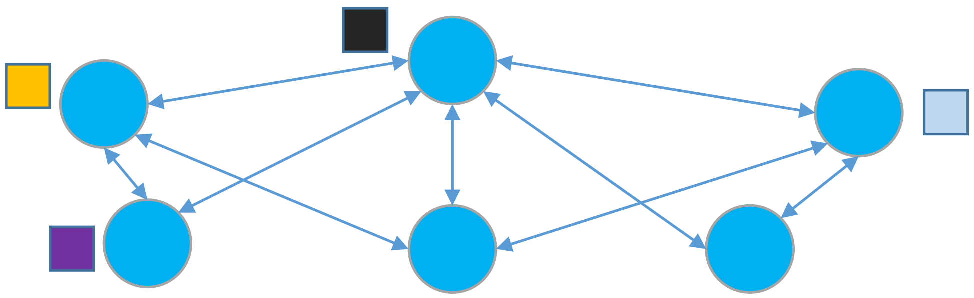
Permissionless (public) blockchains

Bitcoin, Litecoin, Ethereum

Permissioned (private) blockchains

Hyperledger, R3

- By definition any user can join a public blockchain (e.g., Bitcoin).
- For corporate settings, the transparency means that rivals can learn company finances and buy/sale relationships.
- The permissioned blockchains were created for industrial settings.
- Permissioned: Less power consumption, more secure, privacy aware, but for all purposes a gated community.



Data can be more:

- Notary Documents
- Pictures
- Identity Documents
- Shipping logs
- Manufacturing logs
- IOT data

1- On-chain storage

2- Off-chain storage:

✓ Store hashes of data (as proof)

✓ Store the address of data (Our data resides as IP: 145.178.14.29)

UTXO vs Account-Based Blockchains

- Bitcoin and many cryptocurrencies use a construct called an output.
- An output stores a set of addresses and the amount of coins these addresses receive (note that there may be many addresses in a tx).
- Each transaction (except for the coinbase transaction) consumes one or more outputs and creates one or more outputs.
- These blockchains are known as unspent transaction output based (UTXO) blockchains.

UTXO vs Account-Based Blockchains

- A few newer blockchains, such as Ethereum, do not use UTXOs.
- Instead, each address holds an account, and each transaction contains one input and one output address.
- These blockchains are known as account-based blockchains.
- UTXO-account distinction is important because it changes the generated transaction data (and consequently how we model data).

Blockchain – Beyond Cryptocurrencies

Vitalik Buterin

Vitalik Buterin, 2016

Born	January 31, 1994 (age 26) Kolomna, Russia
Nationality	Russian-Canadian
Alma mater	University of Waterloo (dropped out)
Known for	Ethereum, *Bitcoin Magazine*
Awards	Thiel Fellowship
Scientific career	
Fields	Digital contracts, digital currencies, game theory
Website	vitalik.ca

- Butterin created Ethereum to store data and software code on a blockchain.
- Similar to Bitcoin, Ethereum has a currency: Ether.
- The code (a smart contract) is written in a coding language, such as Solidity, which is then compiled to bytecode and executed on the Ethereum Virtual Machine.
- An analogy is the MYSQL snippets stored on a database.

Solidity

First Layer vs Second Layer

- Over time, blockchains started to run into scalability.
- Initial solutions, such as Segregated Witness, were developed to leave some of the encryption signatures and other non-transactional data out of blocks.
- Scalability efforts have culminated in second layer solutions, such as the Lightning Network, where most of the transactions are executed off the blockchain.
- The first layer (i.e., the blockchain itself) only stores a summary of transactions that occur on the second layer.

Lightning Network – 2nd layer solution

- Lightning Network creates another layer on top of the blockchain.
- Users transact with each other offline, without paying transaction fees for each transaction.
- Only the first and last transactions are written to the blockchain.
- LN was designed for repeated low value (micro) transactions, but it can be used for large transactions as well.
- The offline nature implies that we cannot see each transaction individually; only the aggregate information is published to the blockchain at the end.
- Good for transaction privacy, but not for the identity privacy!

Privacy Coins

- Bitcoin's pseudonymous nature poses privacy problems.

- New cryptocurrencies have been developed to break the mapping between input-output addresses, and even hide the transaction amounts.

ZCASH MONERO

DASH

Monero

- **Monero** (April 2014) uses ring signatures and allows users to mix other transaction outputs as (fake) inputs, so that the mapping between inputs and outputs are blurred.

- Transaction structure is transaction output based (TXO), amounts could be visible or hidden. Alphabay adopted Monero in 2016.

Confidential Transaction – amounts are not disclosed.

Inputs (3)

	Amount	Key Image
+	0.008000000000	d582442d895e2bea7a3c605dab0ab2fdc89dc509829087e29ca9cd2fceb5431f
+	0.000000000000	7c2874b22e49428ed77546fb8b9e56aa8624cc201718acc1ca1845466d13bc88
+	0.010000000000	572e2ac6a50c01b51f3eb12a030eb0c556eb1669b0fe73f030ade5d471b0831d

Outputs (2)

Amount	Public Key
0.000000000000	95c16aef66d1eaf1b3db676b9e3f68579b329c39f327be39fc627a2325a6e1bf
0.000000000000	8201c43798760afe6ab42f7b4083bcb1d7f9f50c1b9b2d564fa66875ecd9d185

ZCash

- **Zcash** (October 2016) transactions can be transparent and similar to bitcoin transactions in which case they are controlled by a *t-address*.

- or can be a type of zero-knowledge proof called zk-SNARKs; the transactions are then said to be shielded and are controlled by a *z-address.*

- Newly generated coins are required to pass through the shielded pool.

- Zcash can hide both transaction amounts and user entities, however less than 10% of all transactions were done by using z-addresses.

Kappos, G., Yousaf, H., Maller, M. and Meiklejohn, S., 2018. **An empirical analysis of anonymity in Zcash**. In *27th USENIX Security Symposium (USENIX Security 18)* (pp. 463-477).

Data Extraction and Analysis Tools

Data Extraction Methods

- **Run a full-node on the blockchain to collect all historic transactions – e.g., Bitcoin-Core, Geth, and Parity.**
  - Massive-storage and hardware requirement; more than a week to fully synchronize entire data at a newly connected node.
  - Not good for ad-hoc queries.

- **Web3 services and APIs for data extraction – e.g., Infura, SoChain, and Quicknode.**
  - high costs if users want to extract large amounts of data; paid and slow APIs.
  - Blockchain data is stored at clients in heterogeneous, complex data structures, in binary or in encrypted format, which cannot be directly used for exploration, mining, or visualization.

- **Well-processed blockchain datasets – e.g.,**
  - **Google Big Query** (https://cloud.google.com/blog/products/data-analytics/introducing-six-new-cryptocurrencies-in-bigquery-public-datasets-and-how-to-analyze-them)
  - **https://xblock.pro/#/** (Sun Yat-sen University and others)
  - ETL (extract-transform-load) can still be an issue.

How to Parse the Data

https://github.com/alecalve/python-bitcoin-blockchain-parser

README.md

bitcoin-blockchain-parser build passing coverage 96%

This Python 3 library provides a parser for the raw data stored by bitcoind.

https://github.com/bitcoinj/bitcoinj

README.md

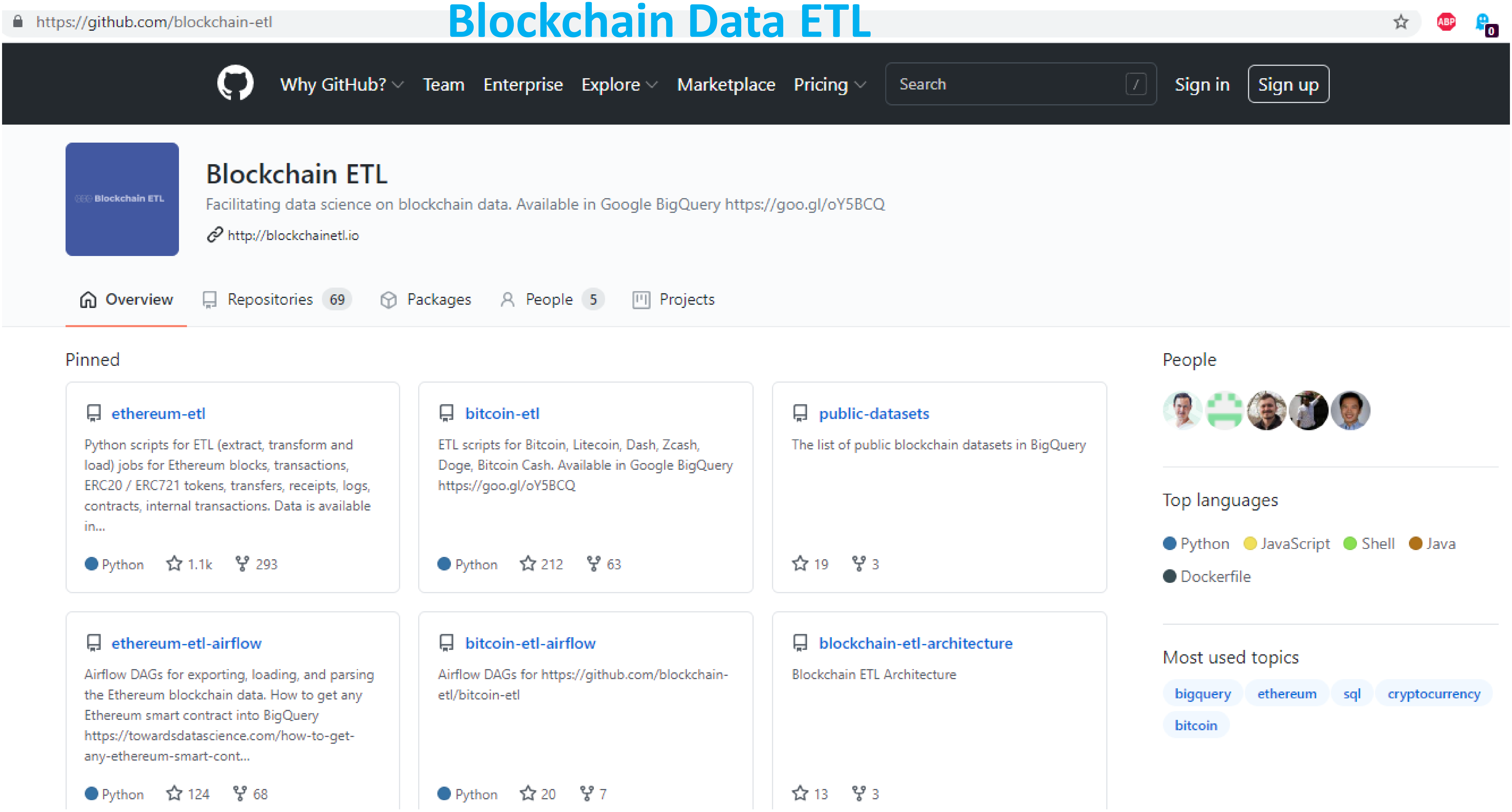
Java CI failing build passing pipeline passed coverage 67%

IRC #bitcoinj

Welcome to bitcoinj

The bitcoinj library is a Java implementation of the Bitcoin protocol, which allows it to maintain a wallet and send/receive transactions without needing a local copy of Bitcoin Core. It comes with full documentation and some example apps showing how to use it.

Blockchain Data ETL



Source of Truth – Google BigQuery

table_id	utc_created_date	utc_modified_date	rows_millions	size_gb
blocks	2019-01-15 13:30:29.658	2021-05-06 05:29:23.607	**11.72**	**12.07**
token_transfers	2019-01-15 13:28:07.793	2021-05-06 05:31:55.894	**595.69**	**171.88**
traces	2019-01-15 13:55:23.777	2021-05-06 05:22:25.641	**2775.28**	**1626.74**
transactions	2019-01-15 13:29:49.289	2021-05-06 05:28:48.798	**985.76**	**455.64**

These four tables from Google BigQuery are the most important sets of data from the Ethereum blockchain in terms of the primary **"interaction networks" between User and Contract accounts**.

Voon Hou Su, Sourav Sen Gupta, Arijit Khan. **Automating ETL and mining of ethereum blockchain network**, WSDM 2022.

Problem to Solve

Convert this

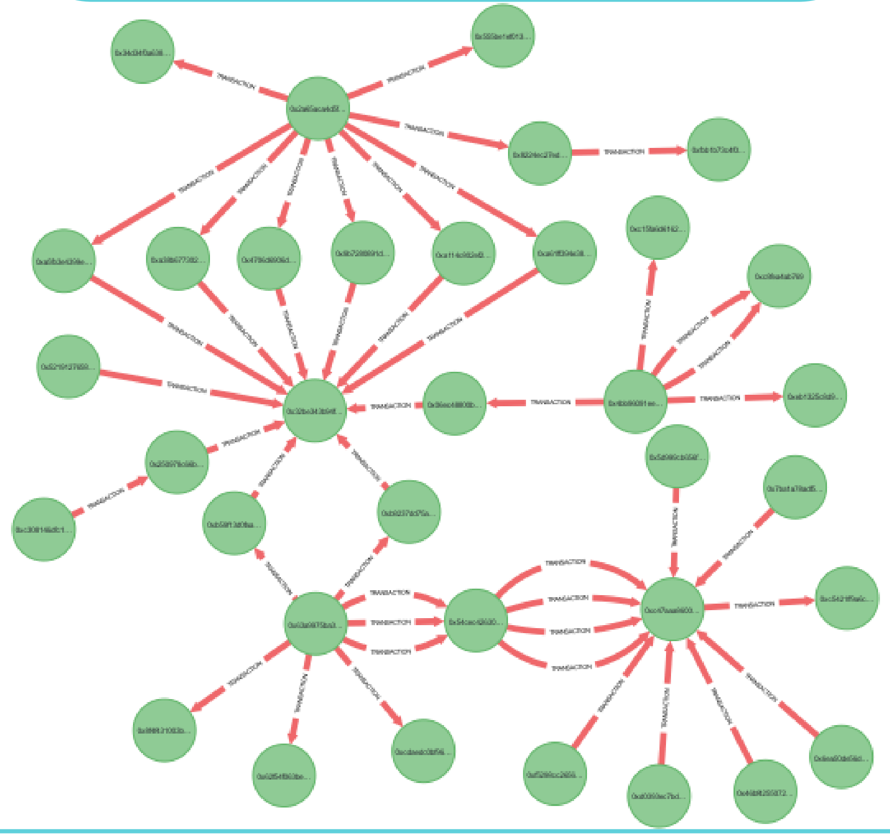
Tabular Representation

How to perform this step?

To this

Graph Representation

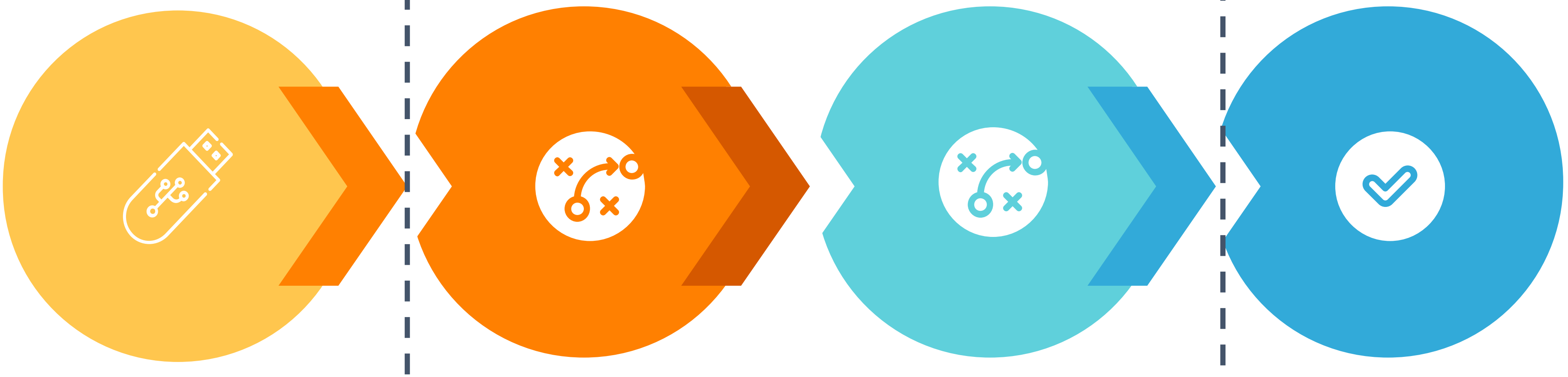
from_address	to_address	edge_data	block_number
0xd3b1fad...	0x1625a9f...	...	0
0x4bc3c20...	0xfe611a3...		1
0x40af81b...	0x5716678...		2
0x9786a24...	0xa25a8dc...		3



Voon Hou Su, Sourav Sen Gupta, Arijit Khan. **Automating ETL and mining of Ethereum blockchain network**, WSDM 2022.

Existing Solution

Download Data	Preprocess Data - 1	Preprocess Data - 2	Data is Ready



Manually download the data of interest from BigQuery	Write a script to extract distinct addresses and hash them	Write a script to build interactions using hash ids obtained from previous step	Data is processed and is ready for analysis

Voon Hou Su, Sourav Sen Gupta, Arijit Khan. **Automating ETL and mining of Ethereum blockchain network**, WSDM 2022.

Existing Solution - Issues

01
No Automation

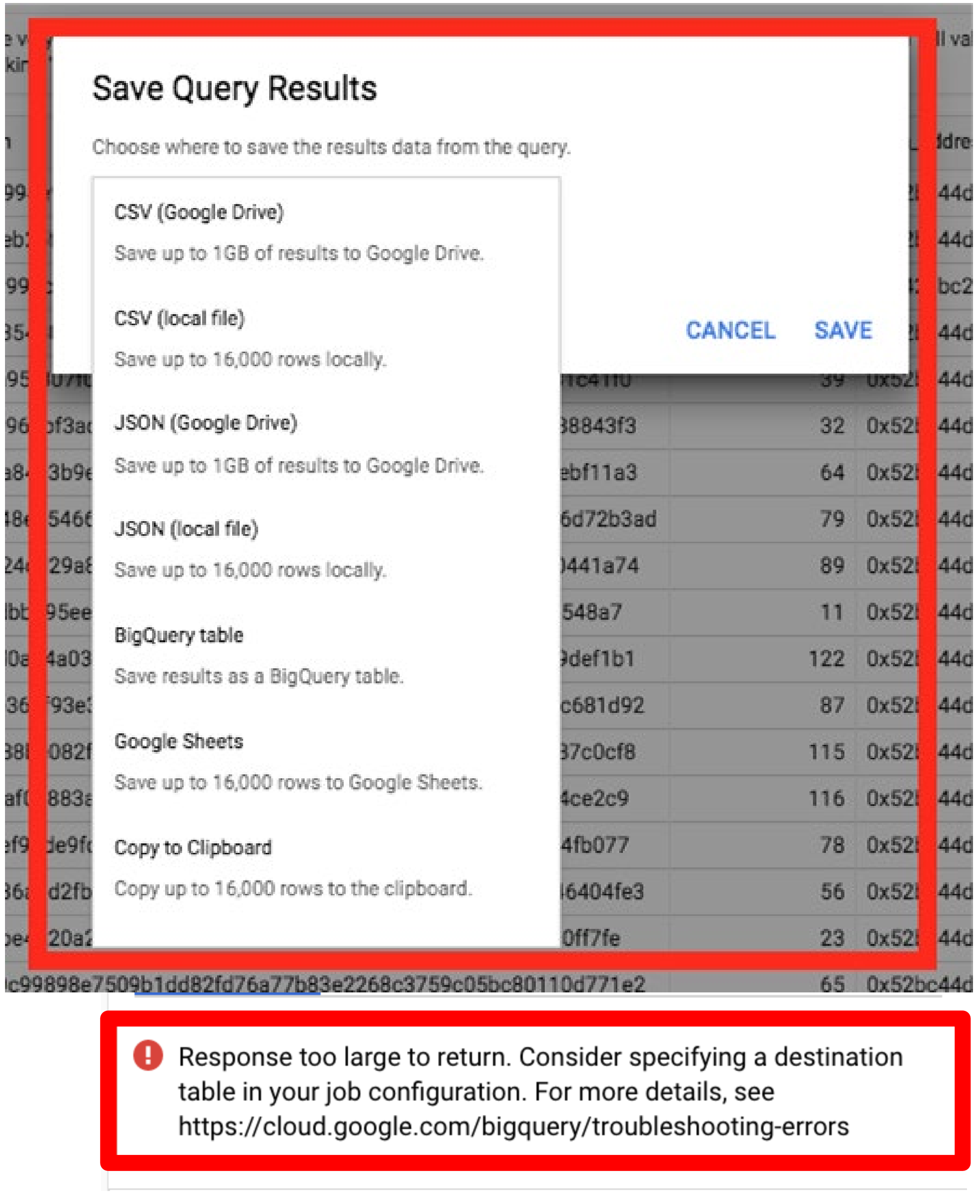
The process is not automated. Users have to write their own BigQuery queries and preprocessing scripts.

02
Not Intuitive

Output format is still in tabular form. Interactions cannot be easily visualised in an intuitive manner.

03
BigQuery Limitations

Difficult to extract BigQuery results that are more than 1GB/16,000 rows in size.



V. H. Su, S. S. Gupta, A. Khan. **Automating ETL and mining of Ethereum blockchain network**, WSDM 2022.

Existing Solution - Issues

01
No Automation

The process is not automated. Users have to write their own BigQuery queries and preprocessing scripts.

02
Not Intuitive

Output format is still in tabular form. Interactions cannot be easily visualised in an intuitive manner.

03
BigQuery Limitations

Difficult to extract BigQuery results that are more than 1GB/16,000 rows in size.

04
No Ability for Data Reuse

Data downloaded from BigQuery is not catalogue and is stored in the filesystem as flat files.

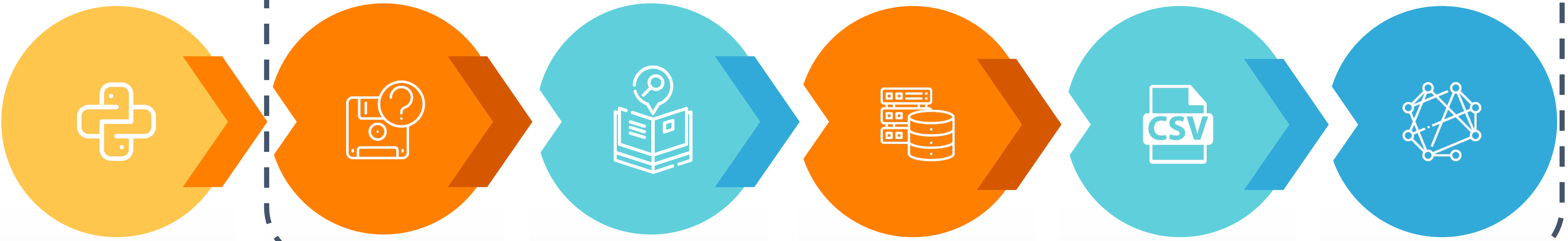
05
Interaction Metadata Lost

Interaction data like amount of tokens and what type of tokens exchanged is not stored/represented.

V. H. Su, S. S. Gupta, A. Khan. **Automating ETL and mining of Ethereum blockchain network**, WSDM 2022.

Proposed Solution - Workflow

Automated Steps



01 User uses the Python SDK to submit an ETL request

02 Missing data in local Hive database is identified

03 Missing data is downloaded from Google BigQuery

04 Missing data is added to local Hive database

05 Required data to construct network/graph is exported to CSV files

06 Network/graph is constructed in Neo4j with CSV files exported in previous step

Voon Hou Su, Sourav Sen Gupta, Arijit Khan. **Automating ETL and mining of Ethereum blockchain network**, WSDM 2022

Proposed Solution - Benefits

EtherNet

Fully Automated

Entire ETL workflow is fully automated

Consistent Entry

Consistent access layer to ETL workflows via EtherNet Python SDK

Data Preserved

Interaction/edge data is preserved in the result

Enables Discovery

Users can see which graphs already exist

Intuitive

Data is stored as graph in a graph database instead of flat files

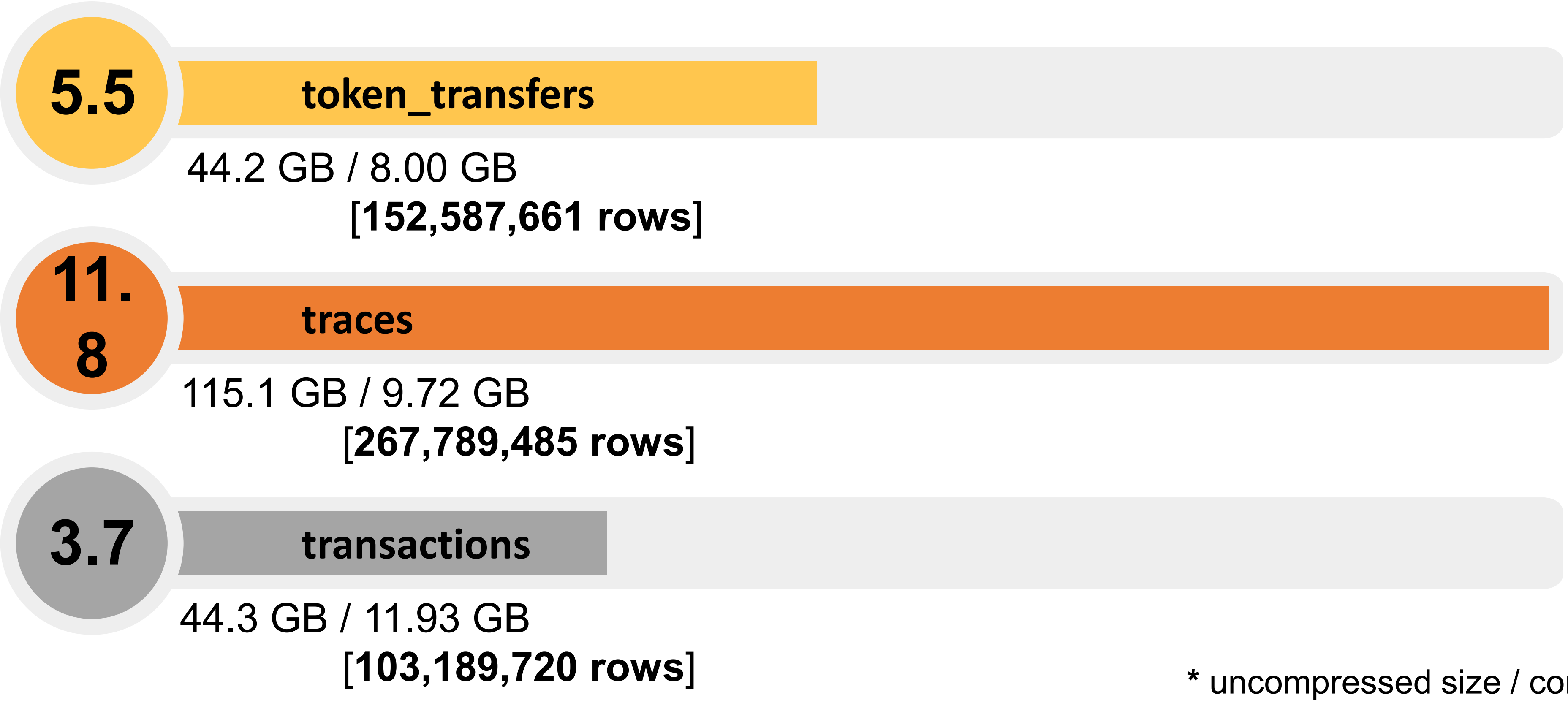
Scalable & Efficient

Data properly indexed and compressed in Hive

Voon Hou Su, Sourav Sen Gupta, Arijit Khan. **Automating ETL and mining of Ethereum blockchain network**, WSDM 2022

Proposed Solution - Efficiency

Data Compression Ratios* in Hive Tables



* uncompressed size / compressed size

Voon Hou Su, Sourav Sen Gupta, Arijit Khan. **Automating ETL and mining of Ethereum blockchain network**, WSDM 2022

Proposed Solution – Future Direction

01
Not too easy to deploy

Many components in the tool, requiring a considerably large overhead in deployment.

02
Domain knowledge required

In order to maintain and optimise EtherNet, domain knowledge on Hadoop and HDFS is required.

03
Lack of cross-connectivity

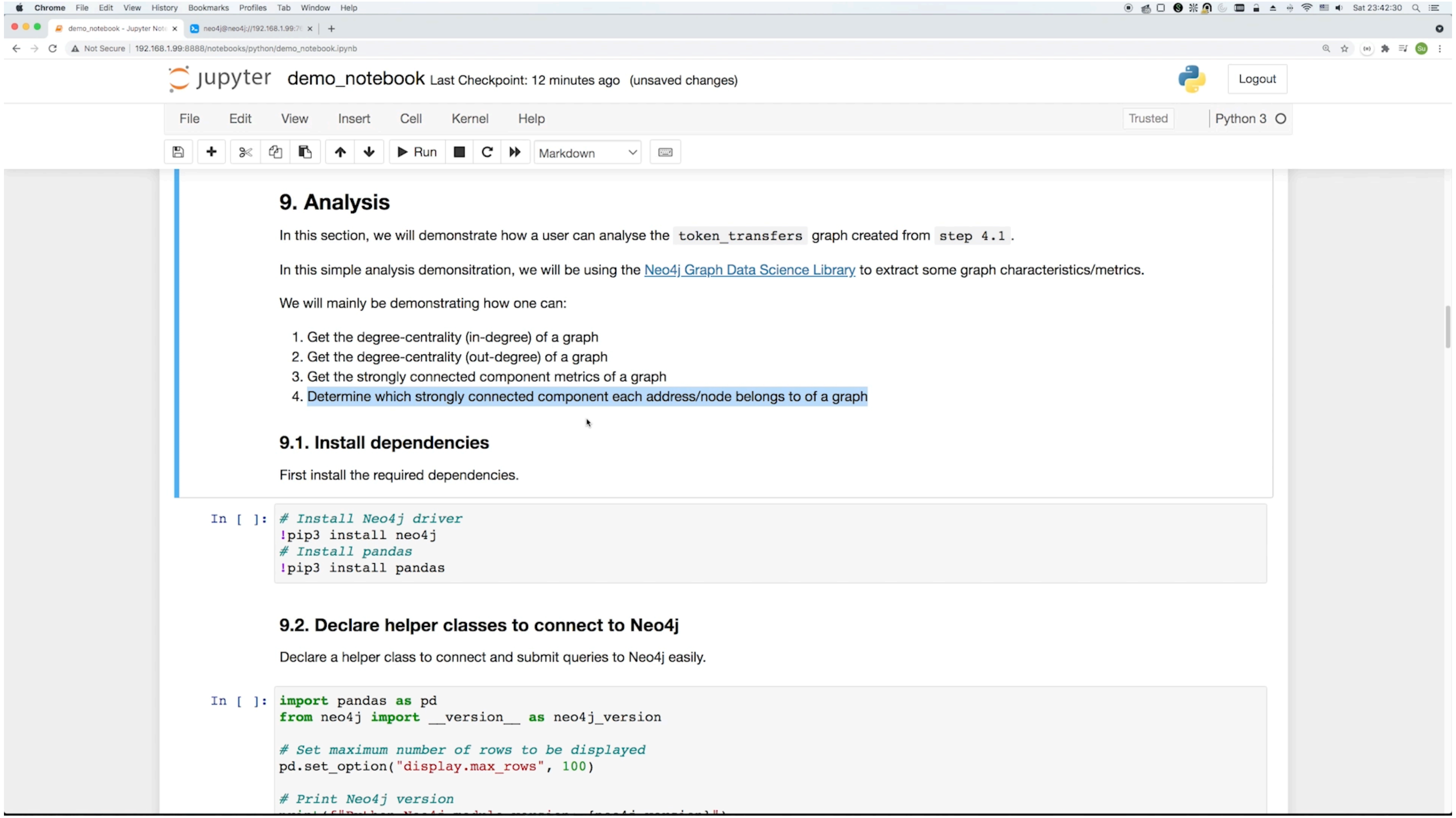
Lack of support with other tools used for network analysis like NetworkX – potential future work.

V. H. Su, S. S. Gupta, A. Khan. **Automating ETL and mining of Ethereum blockchain network**, WSDM 2022.

Check out the toolbox – open-sourced at:

https://github.com/voonhousntu/ethernet

Demonstration – Notebook Interface



V. H. Su, S. S. Gupta, A. Khan. **Automating ETL and mining of Ethereum blockchain network**, WSDM 2022.

Blockchain Query Models

Ethereum Query Language (EQL) is a query language that allows users to retrieve information from the blockchain by writing SQL-like queries.

Not able to search inside contract attributes when querying.

Listing 3: EQL Block Query Example

```
SELECT block.parent.number, block.hash,
    block.timestamp, block.number,
    block.amountOfTransactions
FROM ethereum.blocks AS block
WHERE block.timestamp BETWEEN date('2016-01-01')
    AND now() AND block.transactions.size >10
ORDER BY block.transactions.size
LIMIT 100;
```

Santiago Bragagnolo, Henrique Rocha, Marcus Denker, Stéphane Ducasse. ***Ethereum query language***. *Proceedings of the 1st International Workshop on Emerging Trends in Software Engineering for Blockchain*. 2018.

Blockchain Data Analytic Tools

- **Bartoletti et al.** developed a Scala framework for blockchain data analytics. This can integrate relevant blockchain data with data from other sources, and organize them in a database, either SQL or NoSQL.

- **GraphSense** is an open-source platform for analyzing cryptocurrency transactions.

- **BlockSci** loads the parsed data as an in-memory database, which the user can either query directly or through a Jupyter notebook interface.

- **Industry:** https://santiment.net/ , https://www.nansen.ai/ , https://www.blockchain.com/ , https://www.chainalysis.com/ etc.

M. Bartoletti, S. Lande, L. Pompianu, A. Bracciali. **A general framework for blockchain analytics**. SERIAL@Middleware 2017.
B. Haslhofer, R. Stütz, M. Romiti, R. King. *GraphSense:* **A general-purpose cryptoasset analytics platform**. CoRR 2021.
H. A. Kalodner, M. Möser, K. Lee, S. Goldfeder, M. Plattner, A. Chator, A. Narayanan. **BlockSci: design and applications of a blockchain analysis platform**. USENIX Security Symposium 2020.

Blockchain Data Analytic Tools

○ **Information on User Accounts:** https://etherscan.io/, https://cryptoscamdb.org/, https://tutela.xyz/ - fraud detection and classifying accounts.

○ **Static code analysis, machine learning on smart contracts** are popular for code reuse checking, contract classification, and ponzi schemes detection.

○ **LATTE** provides a novel visual smart contract construction system. This will benefit non-programmers to easily construct a contract by manipulating visual objects and without writing Solidity code.

○ **BiVA** is a graph mining tool for the bitcoin network visualization and analysis and transaction pattern analysis.

F. Victor. **Address clustering heuristics for Ethereum**. Financial Cryptography, 2020.
W. Chen, Z. Zheng, J. Cui, E. C. H. Ngai, P. Zheng, Y. Zhou. **Detecting Ponzi schemes on Ethereum: towards healthier blockchain technology**, WWW, 2018.
T. Hu, X. Liu, T. Chen, X. Zhang, X. Huang, W. Niu, J. Lu, K. Zhou, Y. Liu. **Transaction-based classification and detection approach for Ethereum smart contract**. Inf. Process. Manag. 58(2): 102462 (2021).
S. Tikhomirov, E. Voskresenskaya, I. Ivanitskiy, R. Takhaviev, E. Marchenko, Y. Alexandrov. **SmartCheck: static analysis of Ethereum smart contracts**. WETSEB@ICSE 2018.
S. Ducasse, H. Rocha, S. Bragagnolo, M. Denker, C. Francomme. **SmartAnvil: open-source tool suite for smart contract analysis**. Blockchain and Web 3.0: Social, Economic, and Technological Challenges. 2019.
S. Tan and S. S. Bhowmick and H.-E. Chua and X. Xiao. **LATTE: visual construction of smart contracts**, SIGMOD, 2020.
F. E. Oggier, A. Datta, and S. Phetsouvanh. **An ego network analysis of sextortionists**. Soc. Netw. Anal. Min., 10(1), 2020.

Blockchain Data Analytic Tools

○ **Visualization of blockchain data:** BitConeView, BitConduite, Bitcoinrain, Ethviewer, …

Survey: N. Tovanich, N. Heulot, J.-D. Fekete, P. Isenberg.
Visualization of Blockchain data: a systematic review. IEEE Trans. Vis. Comput.
Graph. 27(7): 3135-3152 (2021)

○ **Natural language processing and sentiment analysis** using tweets, online articles, cryptocurrency prices and charts, Google Trends about blockchain.

- ➢ O. Kraaijeveld and J. D. Smedt. **The predictive power of public Twitter sentiment for forecasting cryptocurrency prices**, 2020, Journal of International Financial Markets, Institutions and Money, 65.

- ➢ A.-D. Vo and Q.-P. Nguyen and C.-Y. Ock, **Sentiment analysis of news for effective cryptocurrency price prediction**, International Journal of Knowledge Engineering, 5(2), 2019.

- ➢ Abraham and D. Higdon and J. Nelson and J. Ibarra. **Cryptocurrency price prediction using tweet volumes and sentiment analysis**, SMU Data Science Review, 2018.

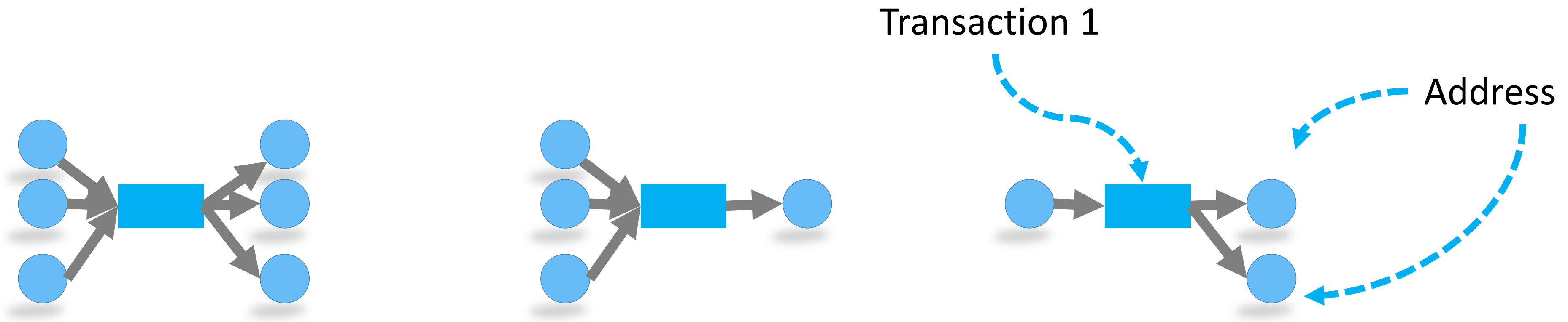
Blockchain Graphs: UTXO, Account Networks

UTXO Graphs: Bitcoin, Litecoin, Monero, ZCash

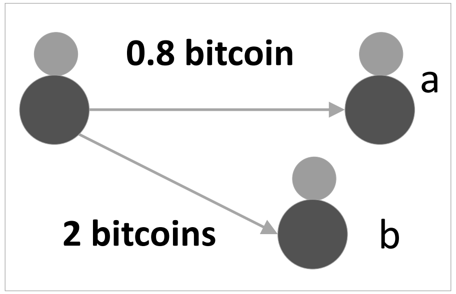
What does a UTXO transaction Look Like?

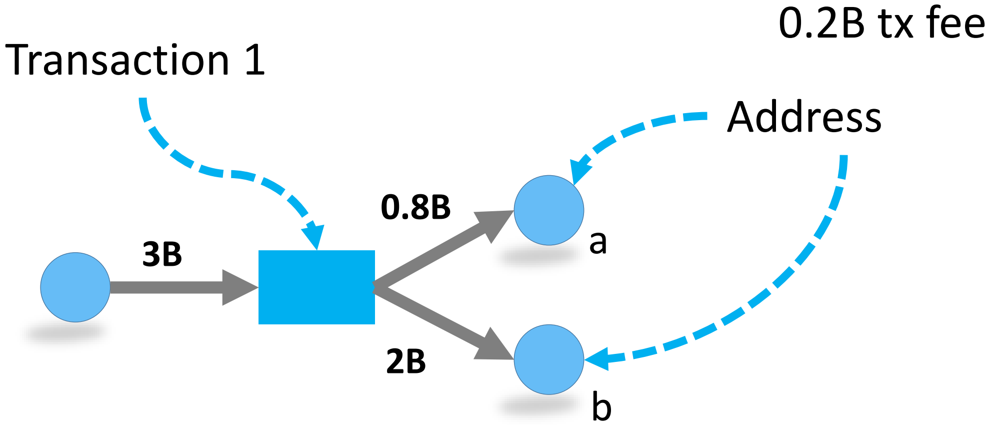
- A UTXO transaction can have $i > 0$ inputs and $o > 0$ outputs. Usually $i = 1$ and $o = 2$ (57% of all transactions in Bitcoin).

- i and o can be arbitrarily large, as long as the transaction size is less than the block size (1MB in Bitcoin).



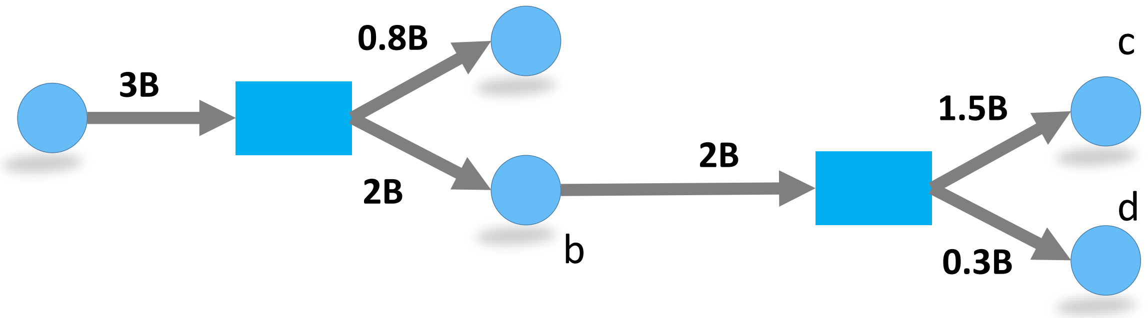
Transaction Output (TXO) Based Blockchains





Next, if address b wants to spend its received 2B, it needs to show proof of funds:

“Use the 2B I received from Block 1, transaction 1 and to pay 1.5B to c and 0.3B to d”.

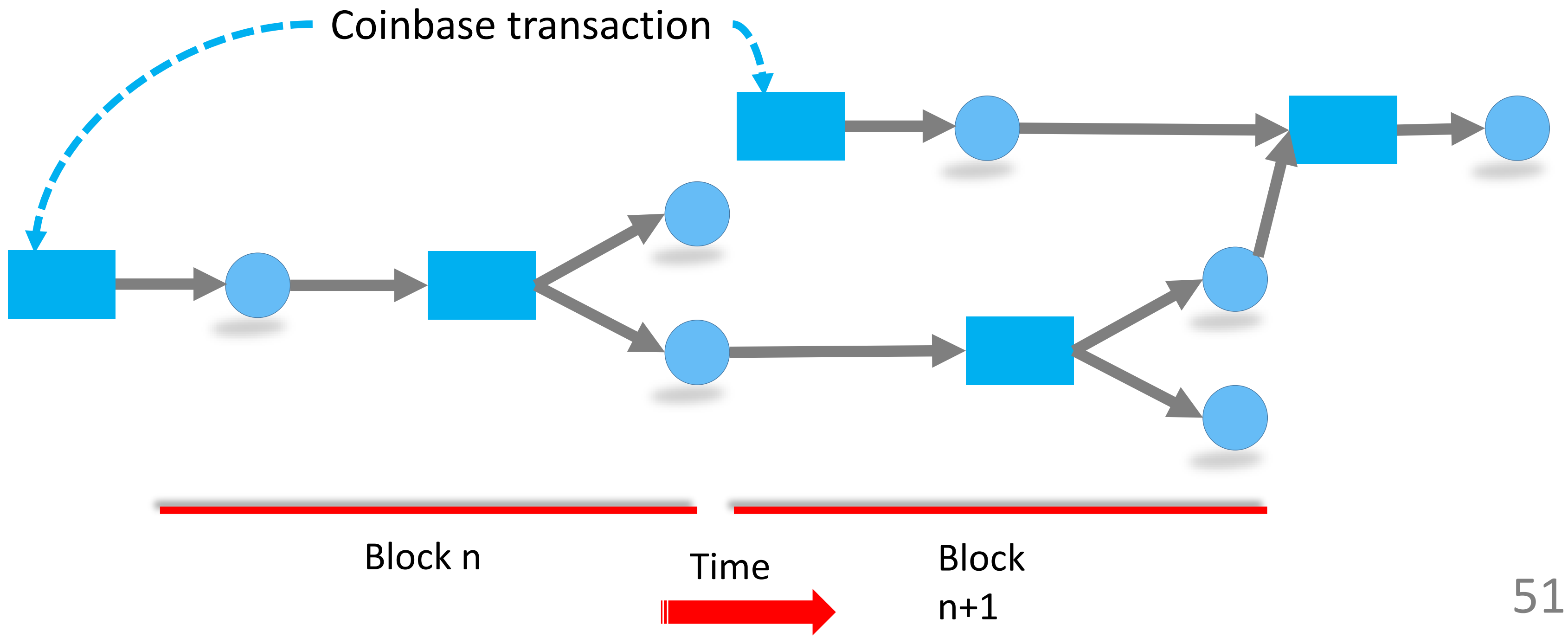


Transaction Output (TXO) Based Blockchains

- Genesis block 0: The first block, created by Nakamoto.

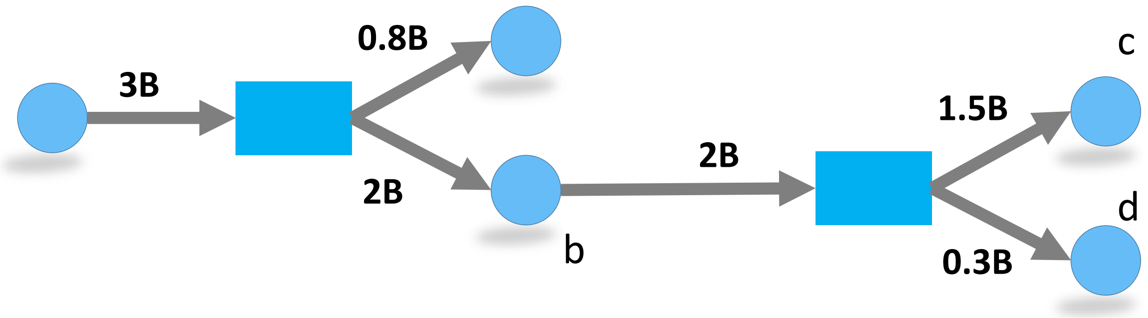
- Every block has one coinbase transaction that creates bitcoins (sum of block reward + transaction fees).

- All other payments must show proof of funds (previous outputs).



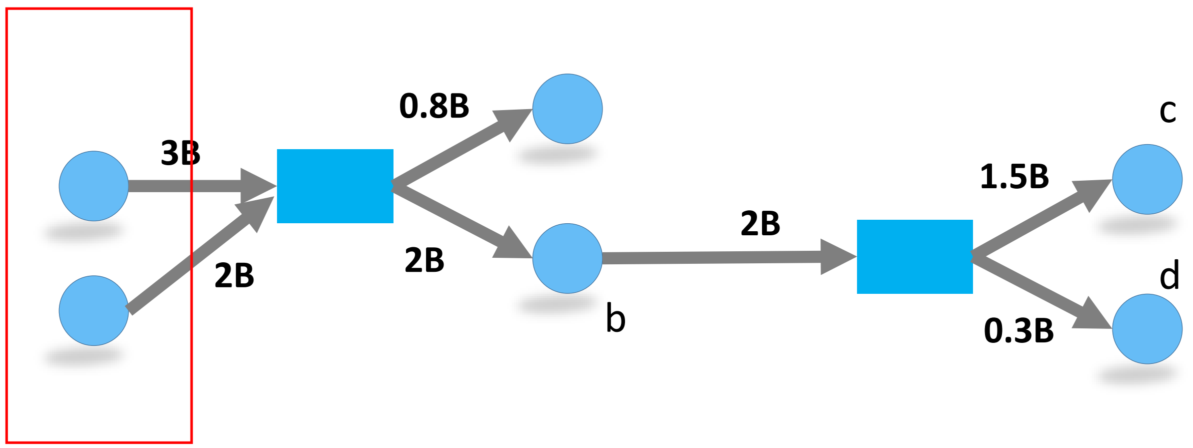
A Few Notes on the Physical Word

- Bitcoin uses addresses to represent accounts. If you want to “open an account”, you need to create a bitcoin address (easily).
- An address is a short string of text that is created by using private/public key cryptography.
- If you know the address of someone, you can send bitcoins to the address. You do not need to know anything else (i.e., owner’s name, zip code, etc.) about the address.
- This means that multiple output addresses in a transaction can belong to two unrelated people.



A Few Notes on the Physical Word

What about input addresses?

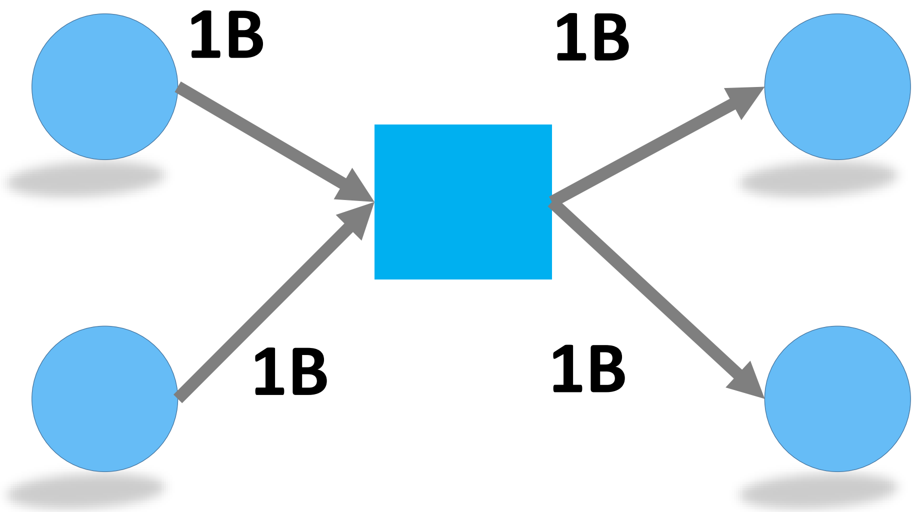


They probably know each other, or they are the same person.
Because they need to sign the transaction by using private keys.

Three Graph Rules for TXO

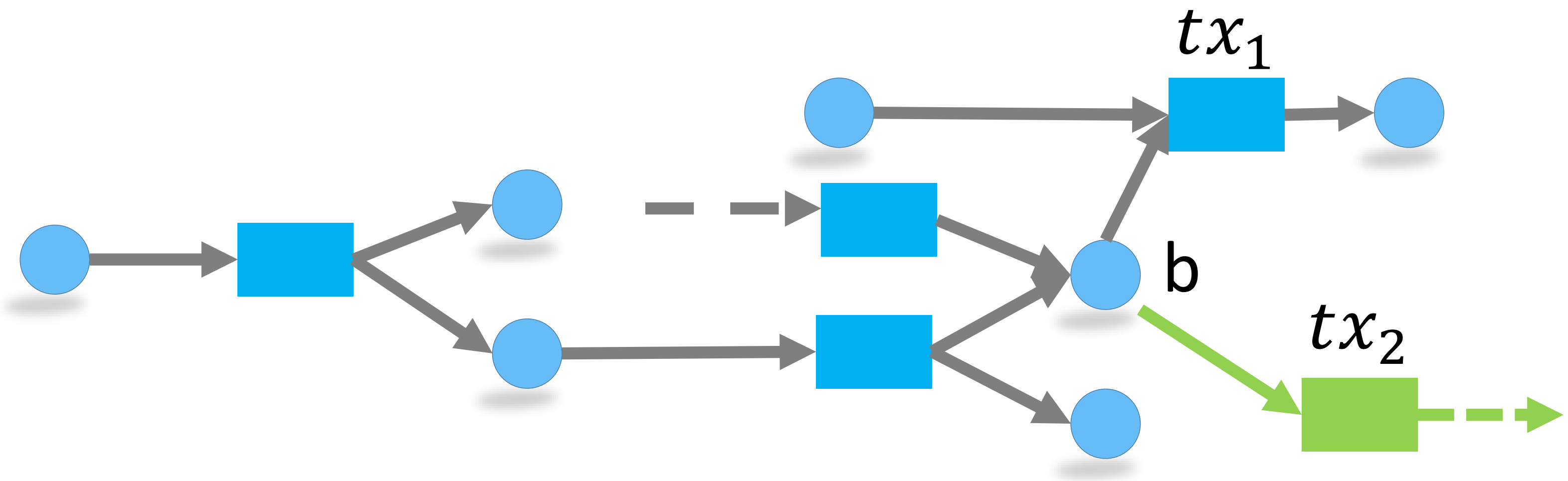
1 – **Mapping Rule**: Multiple inputs can be signed separately and merged, but the input-output address mappings are not recorded.

A transaction can be considered a lake with incoming rivers, and outgoing emissaries. Coins mix in this lake.



Three Graph Rules for TXO

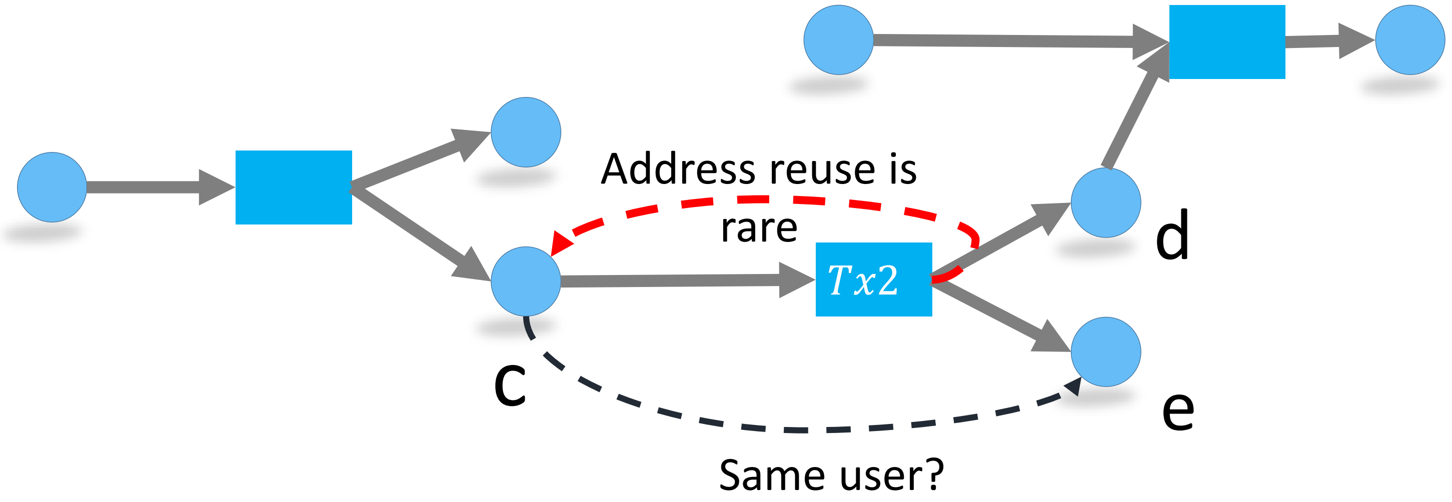
2- **Source Rule**: Coins can be gained from multiple transactions. These can be spent at once or separately (dashed edges connect to unspecified addresses).



Address b can spend bitcoins at tx_1(once), or at tx_1 and tx_2.

Three Graph Rules for TXO

3- **Balance Rule**: All coins gained from a transaction must be spent in a single transaction. Addresses cannot keep change, must forward it.

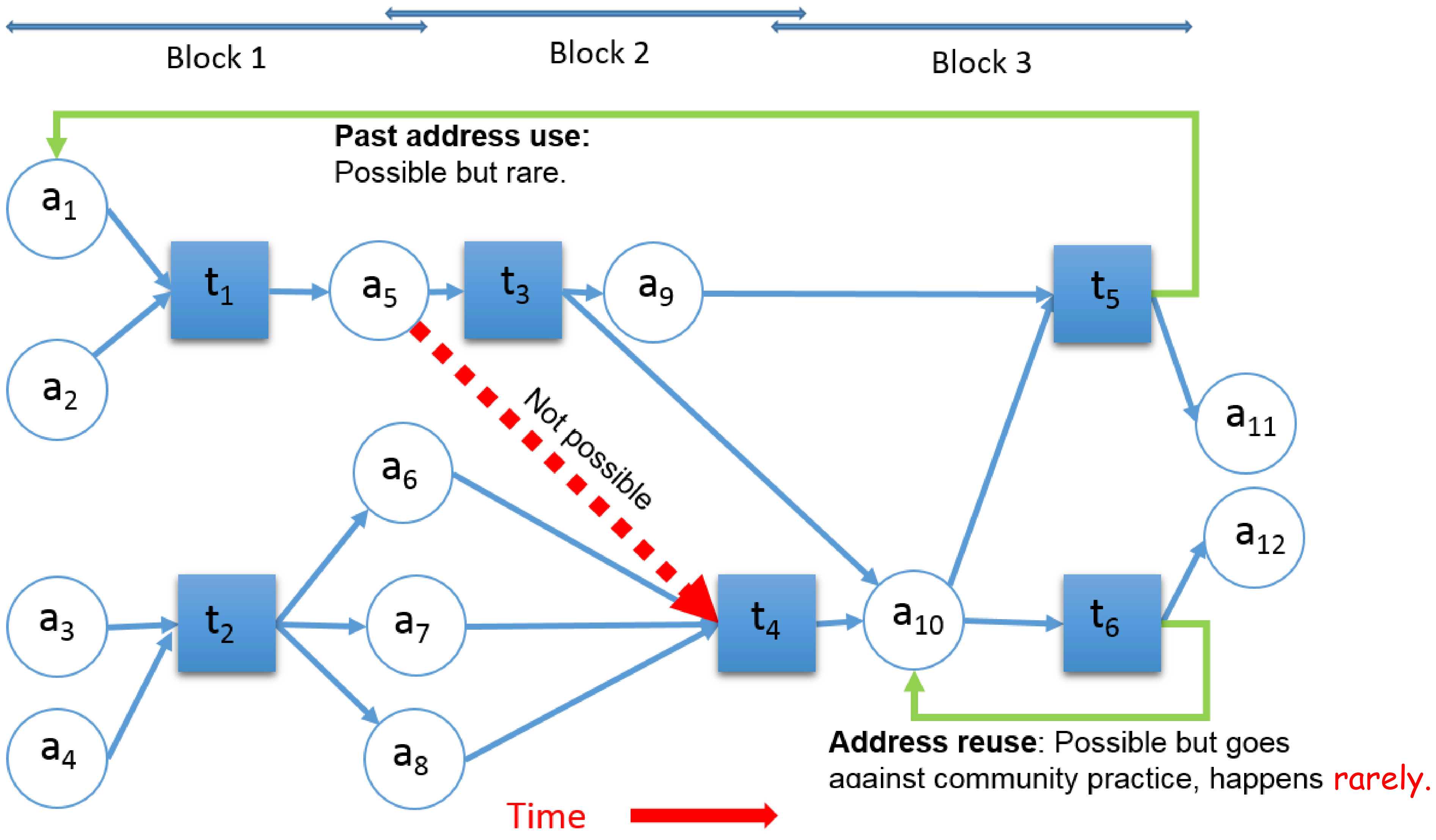


Two cases:

i - c sold all its coins: c, d and e all belong to different people, or
ii - c paid to d, and forwarded the change to its new address e.

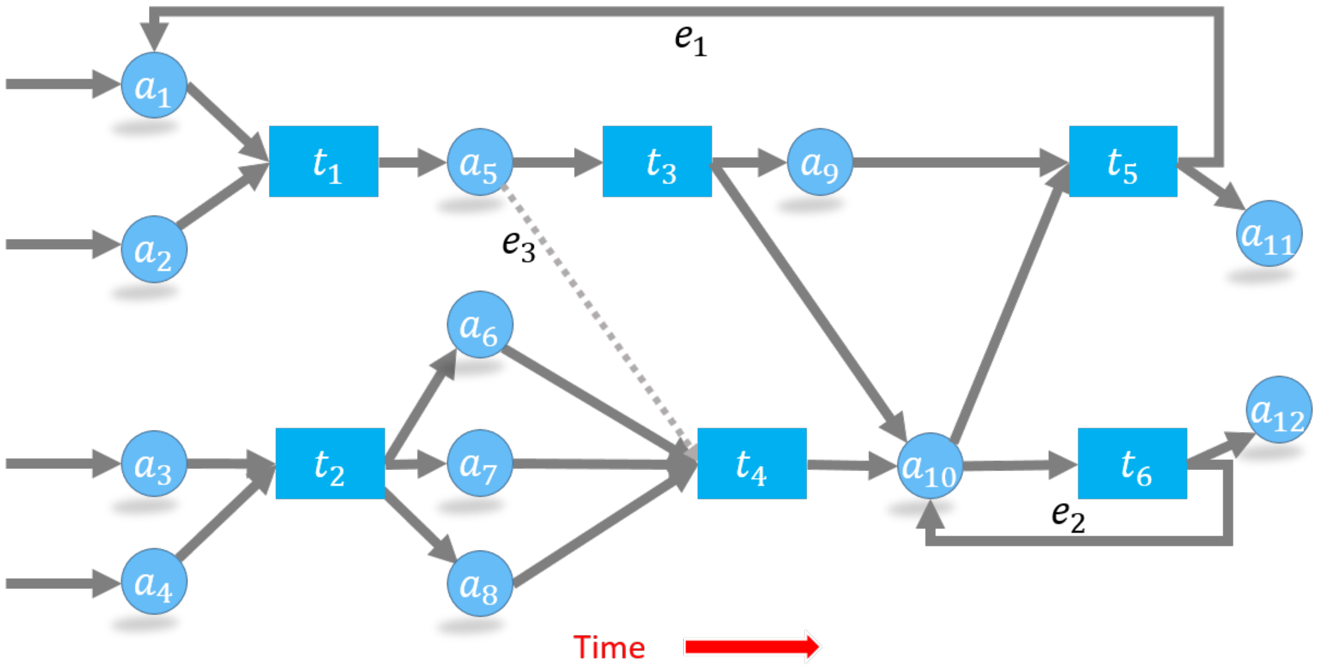
In many scenarios, we have to learn which addresses belong to the same entity.

A Toy TXO Graph

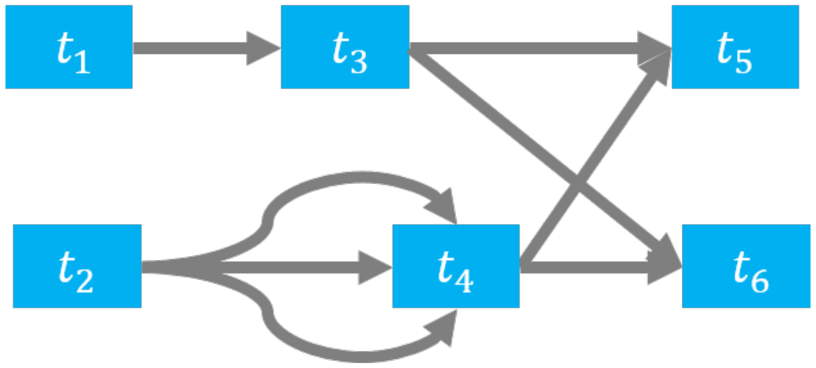


Transaction Graph

- Transaction graphs omit address nodes from the transaction network and create edges among transactions only.



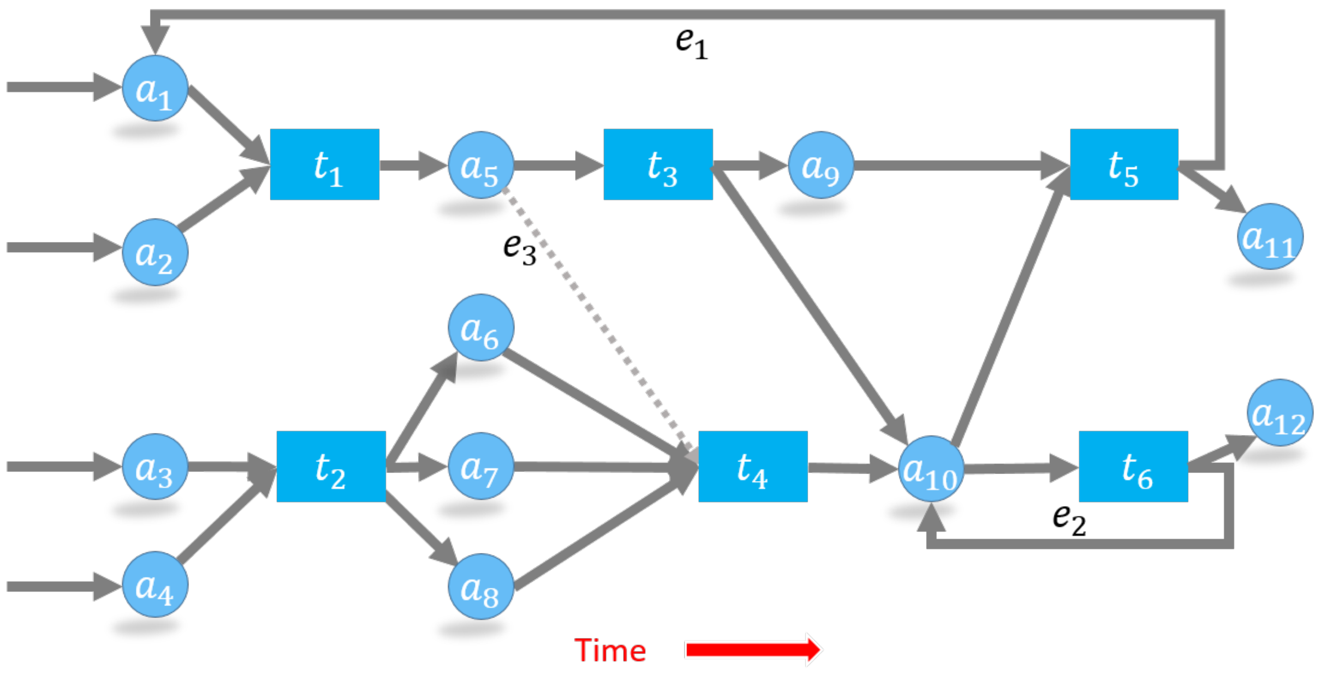
Heterogeneous graph

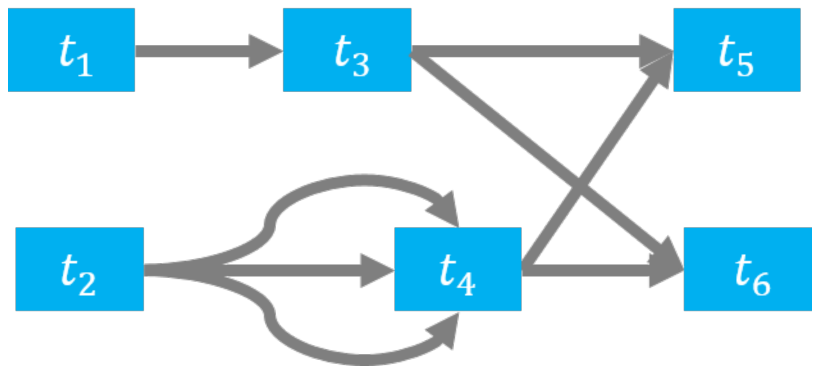


Transaction graph

Disadvantages

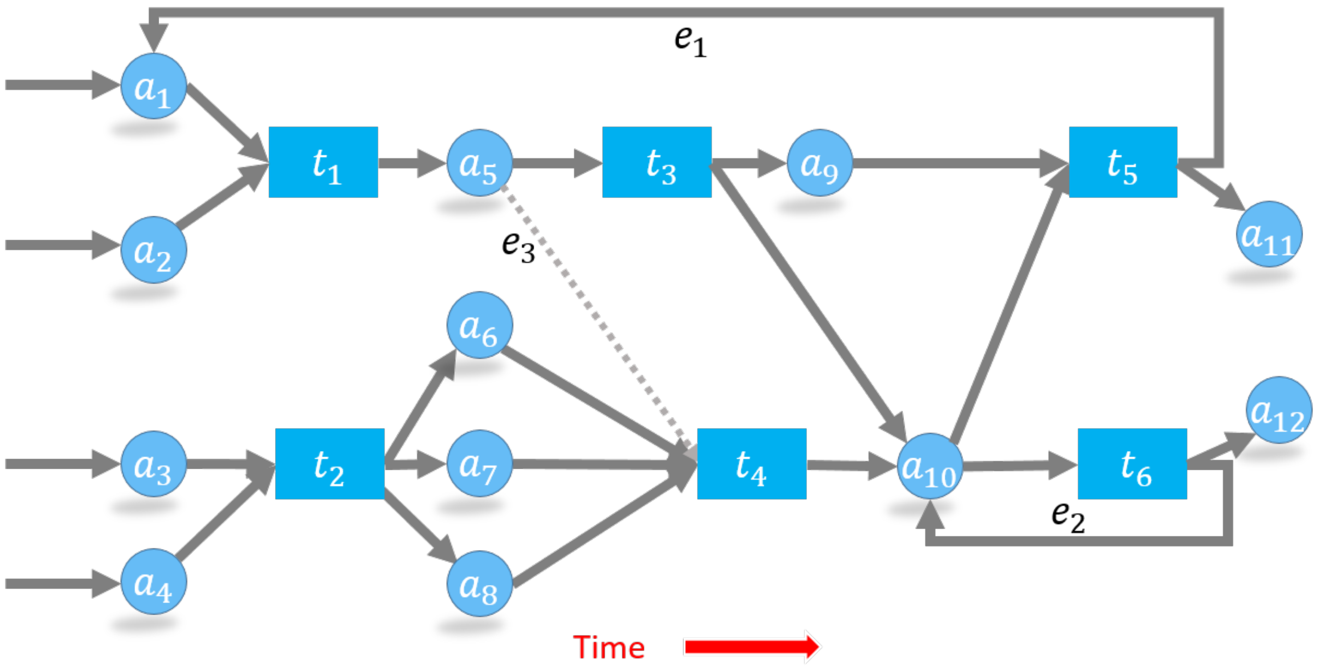
- By omitting addresses, we lose the information that t_5 and t_1 are connected by a_1. The address reuse of a_{10} is hidden in the transaction graph as well.

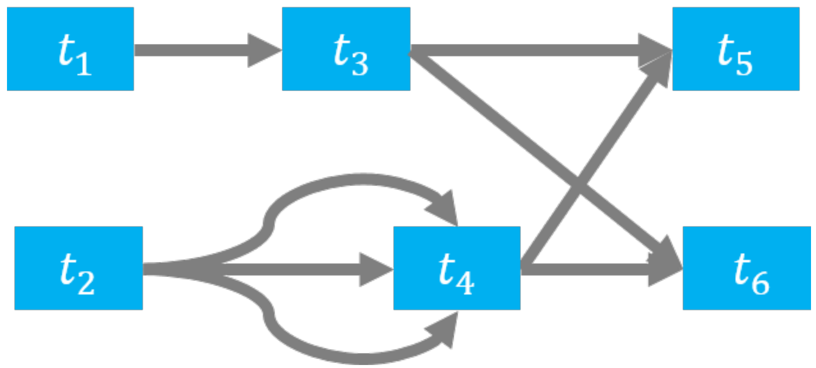




Disadvantages

- Unspent transaction outputs are not visible; we cannot know how many outputs are there in t_5 and t_6. Similarly, if t_3 had an unspent output, we would not learn this information from the graph. In Bitcoin, many outputs stay unspent for years; the transaction graph will ignore all of them.





Advantages

- First, we may be more interested in analyzing transactions than addresses. Many chain analysis companies focus their efforts on identifying transactions that are used in e-crime.

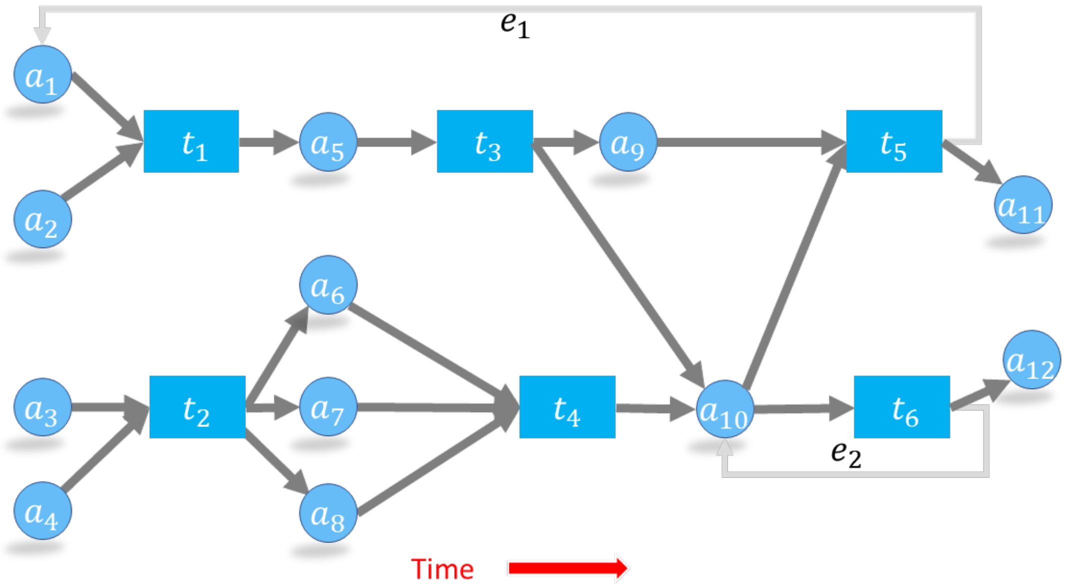
- Second, the graph order (node count) and size (edge count) are reduced from the blockchain network, which is useful for large scale network analysis.

- In UTXO networks, transaction nodes are typically less than half the number of address nodes. For example, Bitcoin contains 400K-800K unique daily addresses but 200K-400K transactions only. However, the real advantage of the transaction graph is its reduced size.

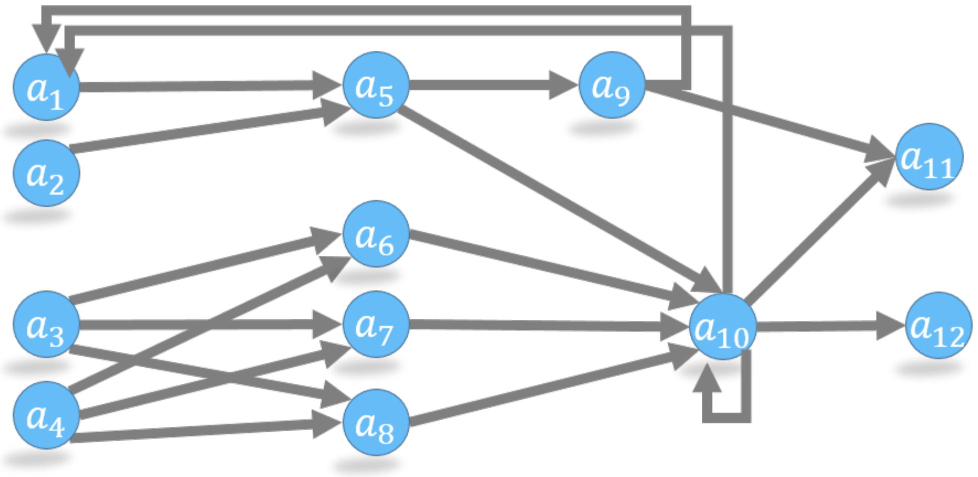
As we will explain in the next section, the address graph contains many more edges than the transaction graph.

UTXO Address Graph

- The address graph omits transactions and creates edges between addresses only.
- Address nodes may appear multiple times, which implies that addresses may create new transactions or receive coins from new transactions in the future.



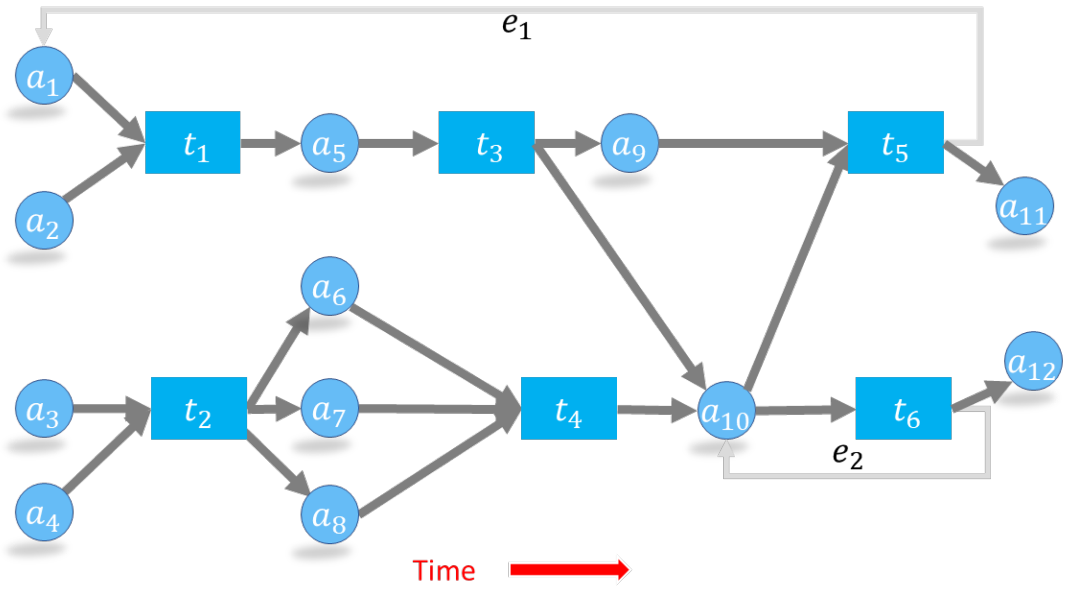
Heterogeneous graph

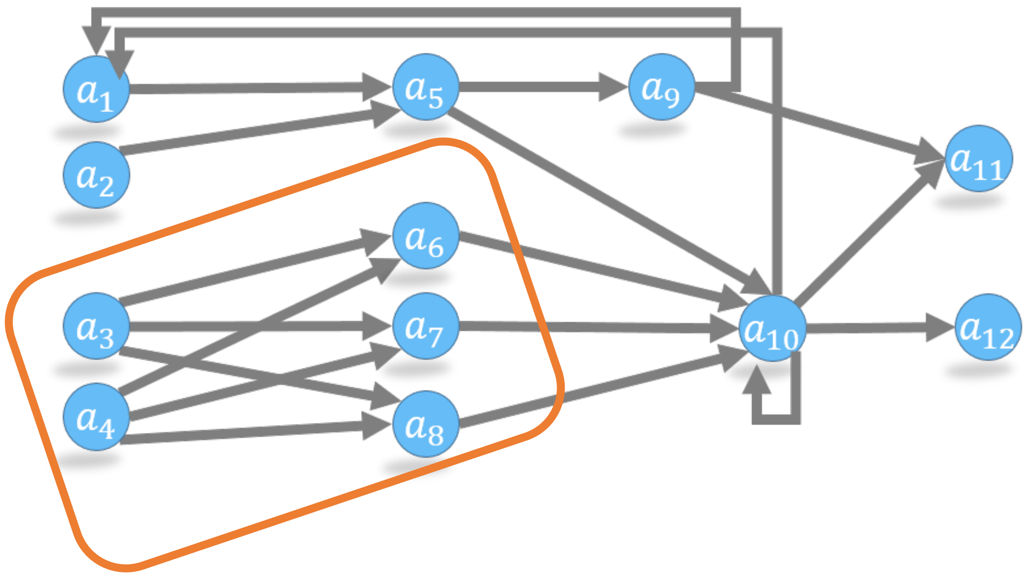


Address graph

UTXO Address Graph

- Address graphs are larger than transaction graphs in node and edge counts.
- As per the mapping rule, we cannot know how to connect input-output address pairs. As a result, we must create an edge between every pair.

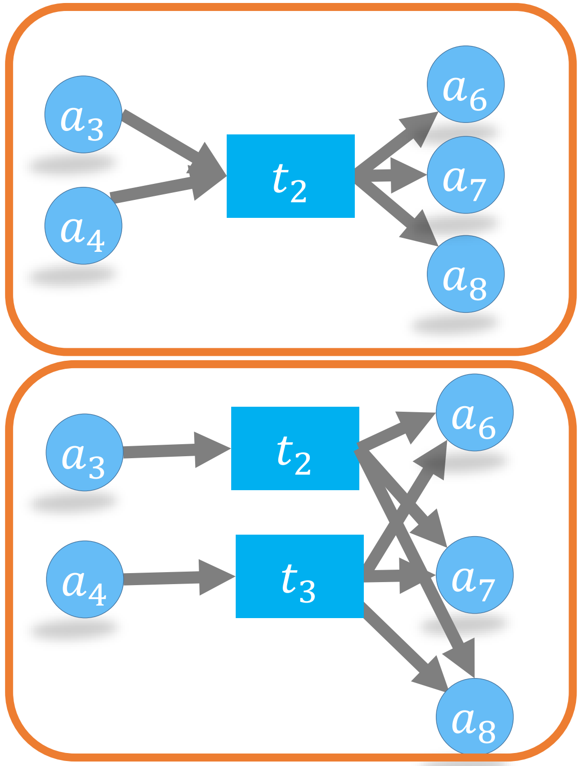




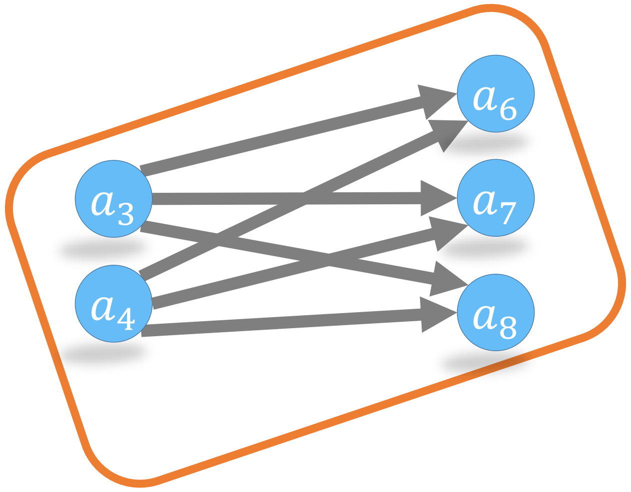
Creating an edge between all address pairs?

UTXO Address Graph

- Graph size is not the only problem. The address graph loses the association of input or output addresses.
- For example, the address graph loses the information that edges a_3 and a_4 were used in a single transaction; address graph edges would be identical if the addresses had used two separate transactions to transfer coins to a_6, a_7 and a_8.



Both create the same address graph



Disadvantages

Address graph: is it worth the trouble searching for graph motifs?

- No: Addresses are not supposed to re-appear in future.
- No: Closed triangles are very rare
- No: Output/input address sets do not have edges to each other – our tools do not consider this, and search for edges in vain (linked transactions within a block are possible but rare)

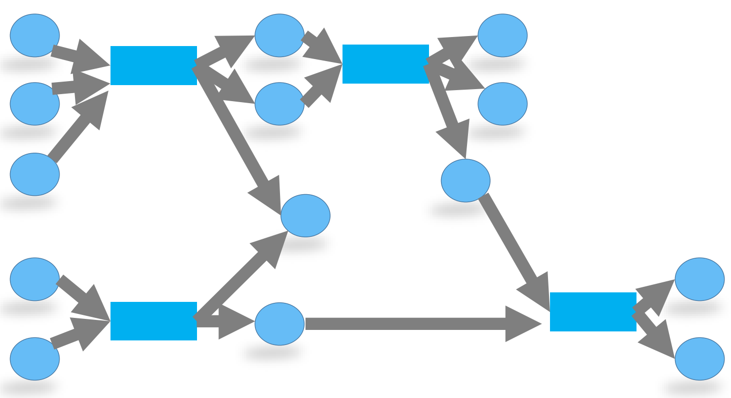
Graph Analysis with single node type:
Not always useful for the forever forward branching tree of Bitcoin

The Chainlet Methodology

Akcora, Cuneyt G., et al. "**Forecasting bitcoin price with graph chainlets**." *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, Cham, 2018.

- Rather than individual edges or nodes, we can use a subgraph as the building block in our Bitcoin analysis.

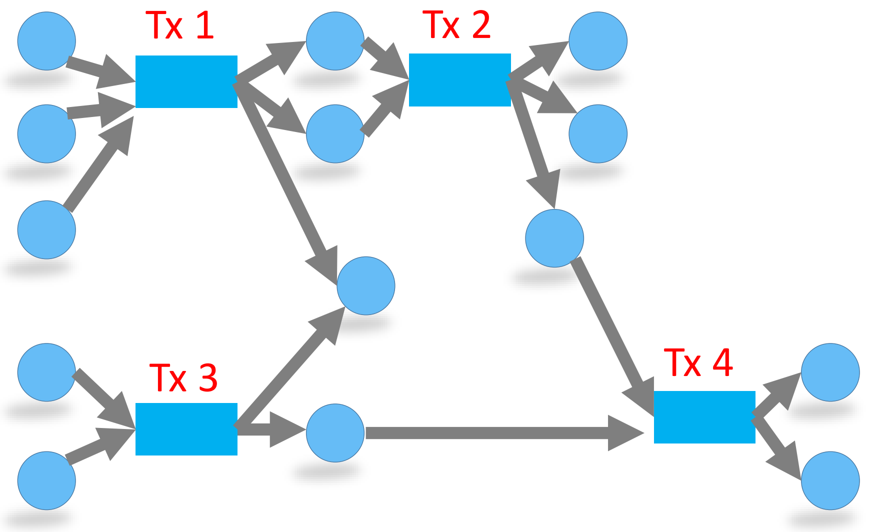
- We use the term **chainlet** to refer to such subgraphs.



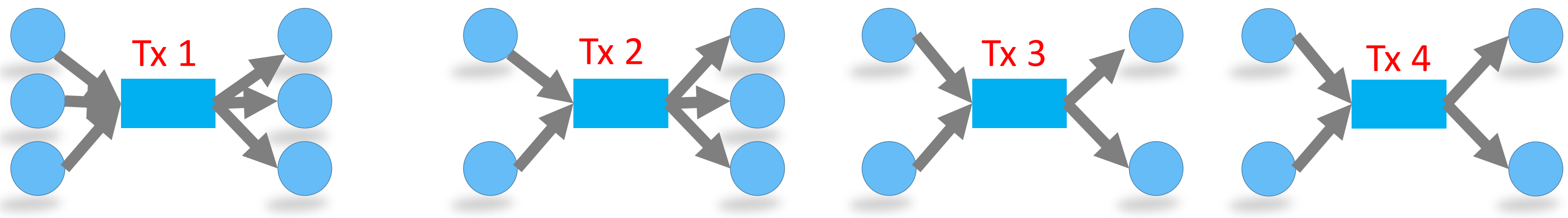
Definition [**K-Chainlets**]:
Let **k-chainlet**$G_k = (V_k, E_k, B)$ be a subgraph of G with **k** nodes of type {**Transaction**}. If there exists an isomorphism between G_k and G', G' ∈G, we say that there exists an occurrence, or embedding of G_k in G.

If a G_k occurs more/less frequently than expected by chance, it is called a Blockchain k-chainlet. A k-chainlet signature $f_G(G_k)$ is the number of occurrences of G_k in G.

Blockchain Chainlets



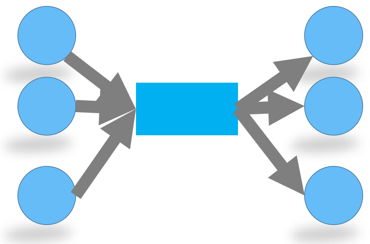
- Chainlets have distinct shapes that reflect their role in the network.
- We aggregate these roles to analyze network dynamics.



Three distinct types of 1-chainlets!

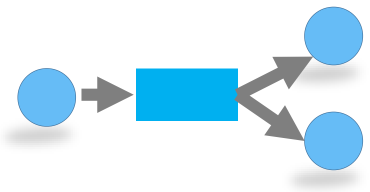
Aggregate Chainlets

$C_{x \to y}$: chainlet with x inputs and y outputs.



Transition. Ex: Chainlet $C_{3 \to 3}$

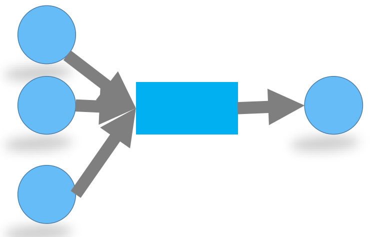
- **Transition Chainlets** imply coins changing address: x = y.



Split. Ex: Chainlet $C_{1 \to 2}$

- **Split Chainlets** may imply spending behavior: y > x.

  But the community practice against address reuse can also create split chainlets.



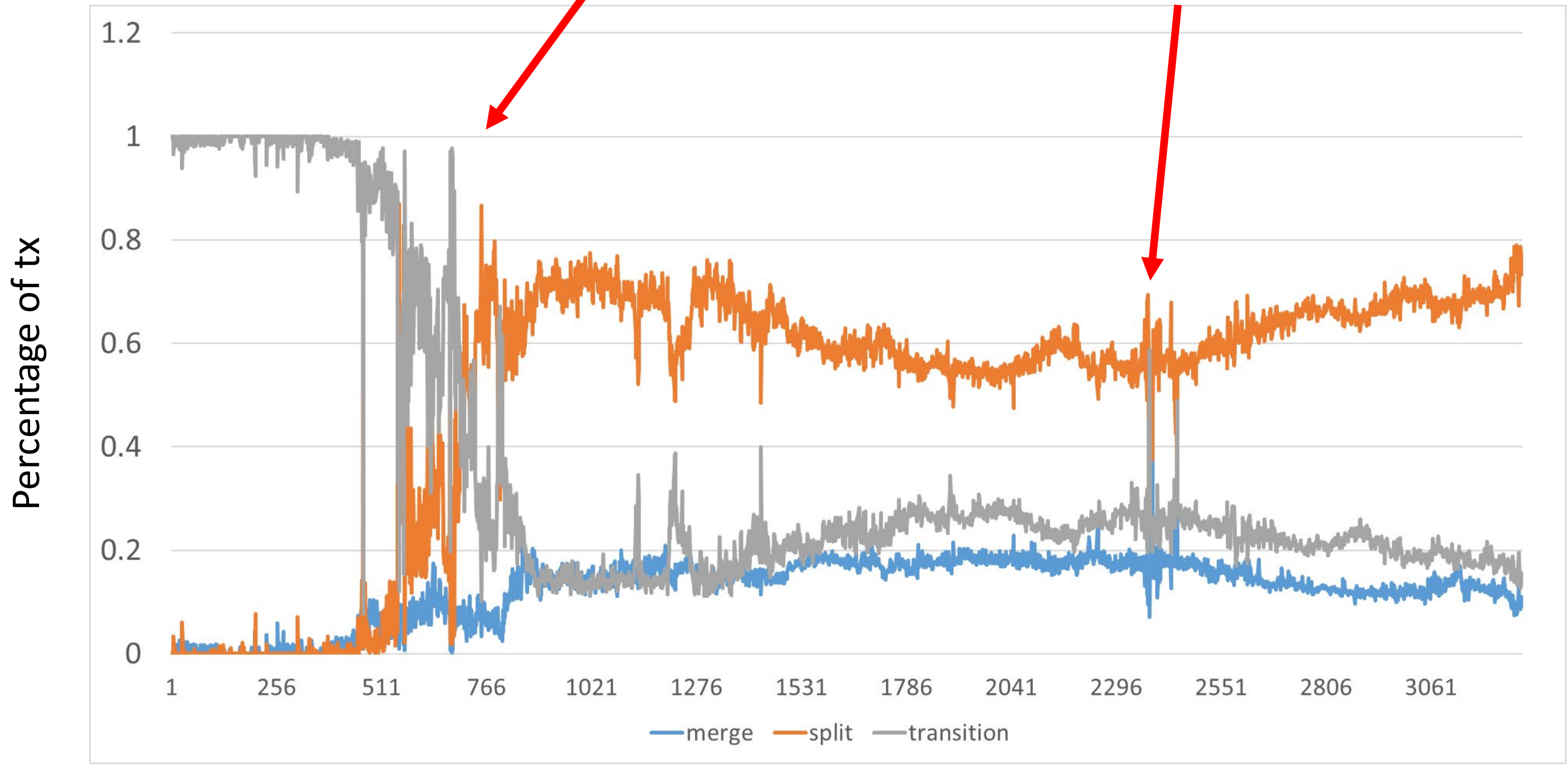
Merge. Ex: Chainlet $C_{3 \to 1}$

- **Merge Chainlets** imply gathering of funds: x > y.

Aggregate Chainlets

Around here 2 pizzas were bought for 10 thousand bitcoins.

Spam attacks to increase block size.

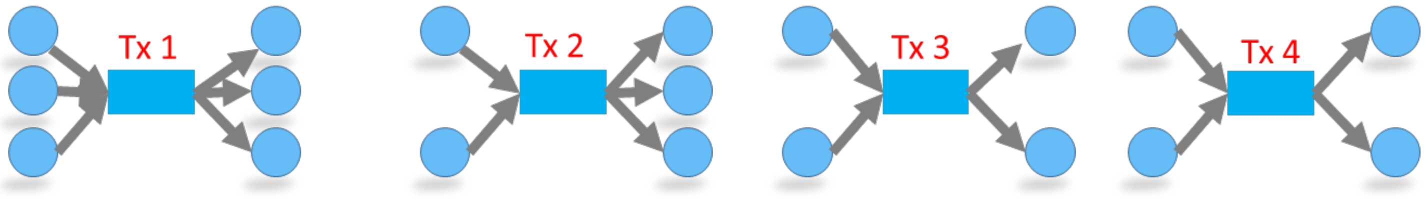


Day since 2009

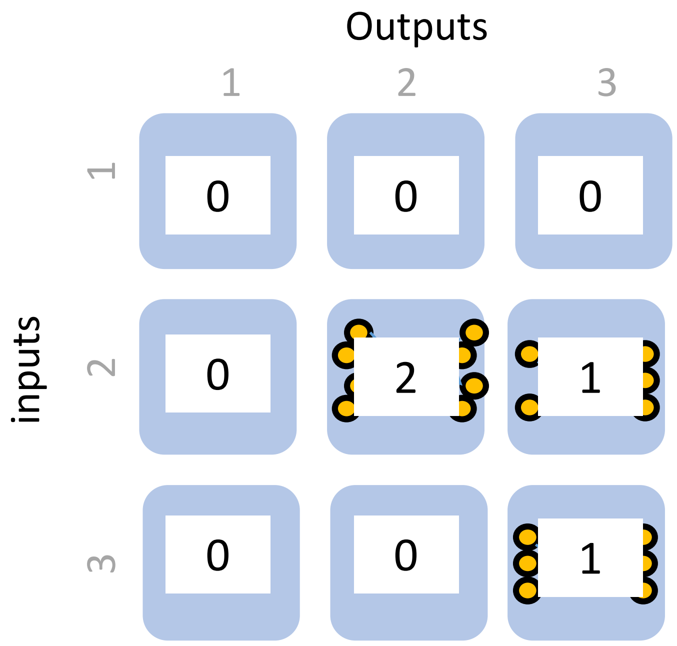
Percentage of aggregate chainlets in the Bitcoin Graph (daily snapshots).

Representing the Network in Time

- For a given time granularity, such as one day, we take snapshots of the Bitcoin graph.
- Chainlet counts obtained from the graph are stored in an N×N matrix.

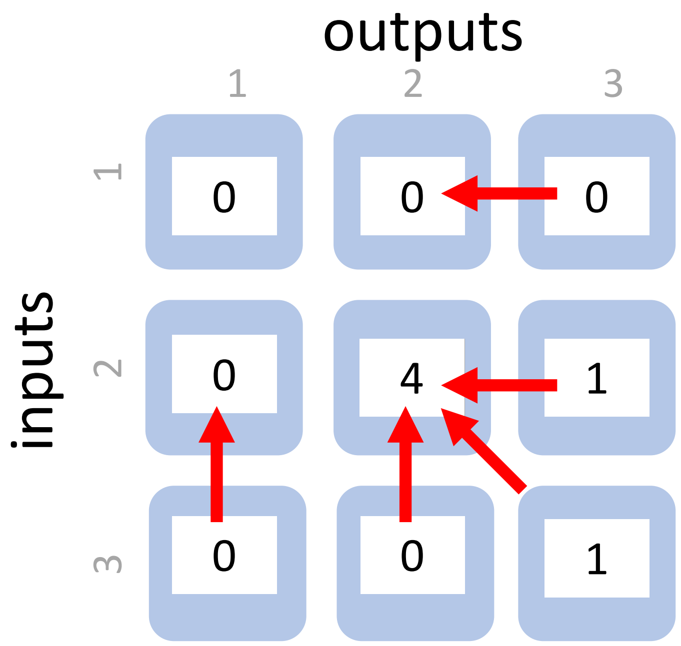


Three distinct types of 1-chainlets!



N: How big should the matrix be?

Extreme Chainlets



- N can reach thousands; the matrix can be 1000×1000.
- On Bitcoin, % 90.50 of the chainlets have N of 5 ($x < 5$ and $y < 5$), and % 97.57for N of 20.

Occurrence matrix

$$O[i,j] = \begin{cases} \#C_{i \to j} & \text{if } i < N \text{ and } j < N \\ \sum_{z=N}^{\infty} \#C_{i \to z} & \text{if } i < N \text{ and } j = N \\ \sum_{y=N}^{\infty} \#C_{y \to j} & \text{if } i = N \text{ and } j < N \\ \sum_{y=N}^{\infty} \sum_{z=N}^{\infty} \#C_{y \to z} & \text{if } i = N \text{ and } j = N \end{cases}$$

Extreme chainlets are the last column/row of the chainlet matrix.

They imply big coin movements in the graph!

Chainlet Behavior

Percentages of all bitcoin chainlets.

Most transactions involve few addresses:
57.04% of transactions have one input and two outputs.

Output address count

Input address count

8.45	57.04	1.56	0.5	0.26	1.05
1.31	12.83	1.79	0.13	0.09	0.25
0.76	3.81	0.38	0.05	0.04	0.11
0.3	2.2	0.1	0.05	0.02	0.07
0.22	1.06	0.05	0.02	0.02	0.05
0.72	2.36	0.11	0.05	0.04	0.45

Account Graphs: Ethereum

Graphs Constructed

- **Survey:** A. Khan, "**Graph analysis of the Ethereum blockchain data: a survey of datasets, techniques, and future direction**", IEEE International Conference on Blockchain 2022

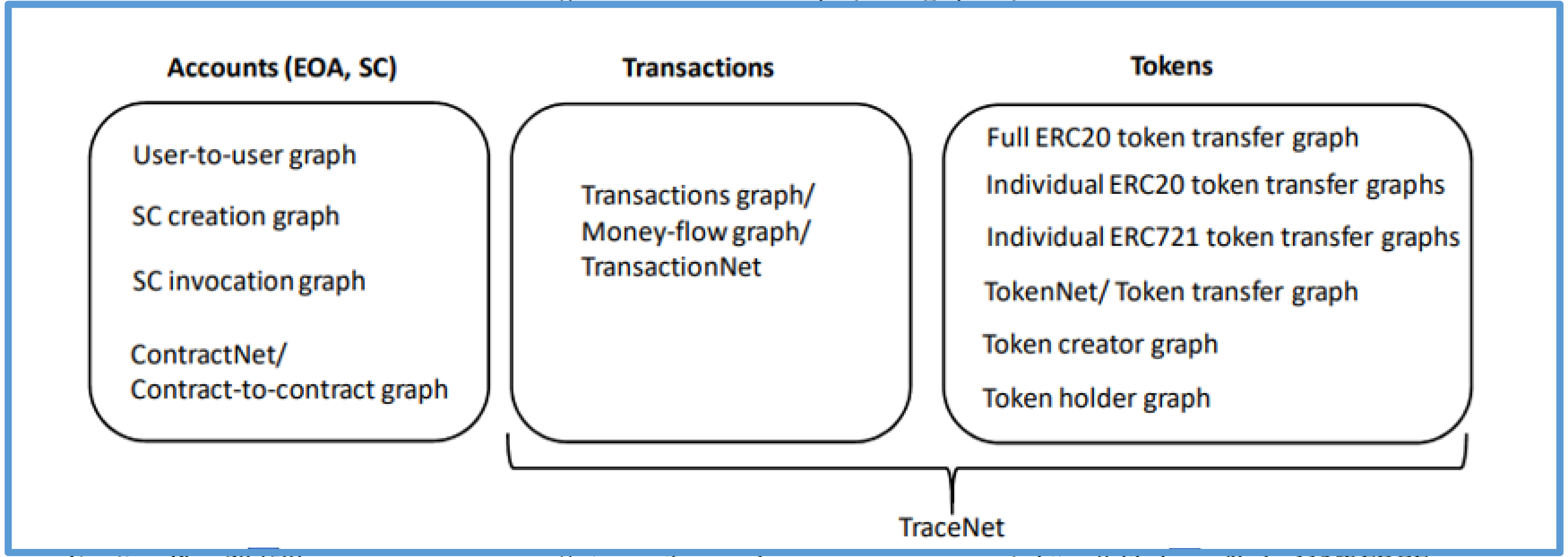
paper	constructed graphs	links to data and/or code
INFOCOM18 [36]	money flow graph, contract creation graph, contract invocation graph	https://github.com/brokendragon/Ethereum_Graph_Analysis
PLOS ONE18 [37]	transaction graph	https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/XIXSPR
Complex Sys18 [38]	(full) ERC20 tokens transfer graph	not given
NTMS18 [39]	user-to-user, user-to-smart contract, and smart contract deployment graphs	not given
FC19 [40]	(individual) ERC20 token transfer graphs	not given
ICDMW19 [41]	Storj token transfer graph	not given
Appl. Netw. Sci.19 [42]	transaction graph	not given
Inf. Sci.19 [43]	transaction graph	not given
WWW20a [44]	trace graph, contract graph, transaction graph, token graph	https://github.com/sgsourav/blockchain-network-analysis
SDM20 [45]	(individual) ERC20 token transfer graphs	https://github.com/yitao416/EthereumCurve
WWW20b [23]	ERC20 token creator, holder, and transfer graphs	http://xblock.pro/#/
Sci Rep20 [46]	(individual) ERC20 token transfer graphs	not given
ACM Meas. Anal. Comput. Syst.20 [47]	ERC20 token creator, holder, and transfer graphs for counterfeit tokens	not given
Concurr. Comput. Pract. Exp.20 [48]	transaction graph	not given
IEEE Trans. Circuits Syst.20 [49]	transaction graph	https://github.com/lindan113/T-EDGE
Frontiers Phys.20 [50]	transaction graph	https://github.com/lindan113/T-EDGE
J. Complex Networks20 [51]	transaction graph	not given
Networking20 [9]	user-to-user, contract-to-contract, and user-contract graphs	not given
SBP-BRiMS20 [52]	(full) ERC20 tokens transfer graph	not given
WWW21 [8]	trace graph, contract graph, transaction graph, token graph	https://github.com/LinZhao89/Ethereum-analysis
ECML PKDD21 [10]	(individual) token transfer graphs, stacked as a multi-layer network	https://github.com/tdagraphs
PAKDD21 [53]	transaction graph	https://github.com/fpour/SigTran
ACM Trans. Internet Techn.21 [55]	transaction graph	http://xblock.pro/#/
Blockchain21 [56]	(individual) ERC721 token transfer graphs	https://github.com/epfl-scistimm/2021-IEEE-Blockchain
IEEE Trans. Syst. Man Cybern. Syst.22 [54]	transaction graph	http://xblock.pro/#/

Graphs Constructed

- **Survey:** A. Khan, " **Graph analysis of the Ethereum blockchain data: a survey of datasets, techniques, and future direction** ", IEEE International Conference on Blockchain 2022

- Static graphs
- Dynamic graphs
- Temporal snapshot graphs
- Directed graphs
- Weighted graphs (?weight)
- Simple and multi-graphs
- Attributed graphs
- Multi-layer networks

paper	constructed graphs	links to data and/or code
INFOCOM18 [36]	money flow graph, contract creation graph, contract invocation graph	https://github.com/brokendragon /Ethereum_Graph_Analysis
PLOS ONE18 [37]	transaction graph	https://dataverse.harvard.edu/dataset.xhtml? persistentId=doi:10.7910/DVN/XIXSPR
Complex Sys18 [38]	(full) ERC20 tokens transfer graph	not given
NTMS18 [39]	user-to-user, user-to-smart contract, and smart contract deployment graphs	not given
Frontiers Phys.20 [50]	transaction graph	https://github.com/lindan113/T-EDGE
J. Complex Networks20 [51]	transaction graph	not given
Networking20 [9]	user-to-user, contract-to-contract, and user-contract graphs	not given
SBP-BRiMS20 [52]	(full) ERC20 tokens transfer graph	not given
WWW21 [8]	trace graph, contract graph, transaction graph, token graph	https://github.com/LinZhao89 /Ethereum-analysis
ECML PKDD21 [10]	(individual) token transfer graphs, stacked as a multi-layer network	https://github.com/tdagraphs
PAKDD21 [53]	transaction graph	https://github.com/fpour/SigTran
ACM Trans. Internet Techn.21 [55]	transaction graph	http://xblock.pro/#/
Blockchain21 [56]	(individual) ERC721 token transfer graphs	https://github.com/epfl-scistimm /2021-IEEE-Blockchain
IEEE Trans. Syst. Man Cybern. Syst.22 [54]	transaction graph	http://xblock.pro/#/

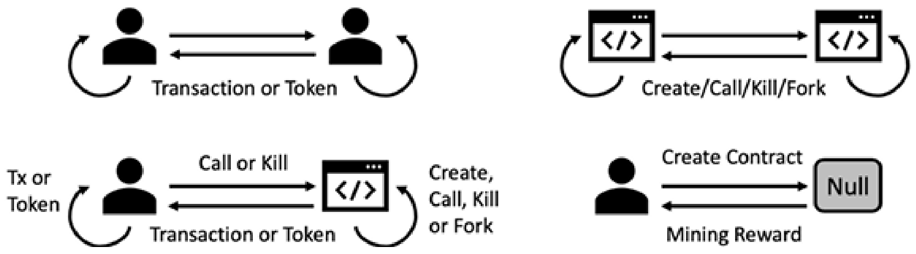


Graphs between Accounts:

○ Ethereum has two types of accounts:

➢ **Externally owned accounts (EOAs)** are accounts controlled by private keys. If a participant own the private key of an EOA, the participant has the ability to send ether and messages from it.

➢ **Smart contract code controlled accounts** have their own code, and are controlled by the code.



- **User-to-User Graph**
- **Smart Contract Creation Graph**
- **Smart Contract Invocation Graph**
- **ContractNet/ Contract-to-Contract Graph**

○ T. Chen, Y. Zhu, Z. Li, J. Chen, X. Li, X. Luo, X. Lin, and X. Zhang, "**Understanding Ethereum via graph analysis**," in INFOCOM, 2018.

○ A. Anoaica and H. Levard, "**Quantitative description of internal activity on the Ethereum public blockchain**," in NTMS, 2018.

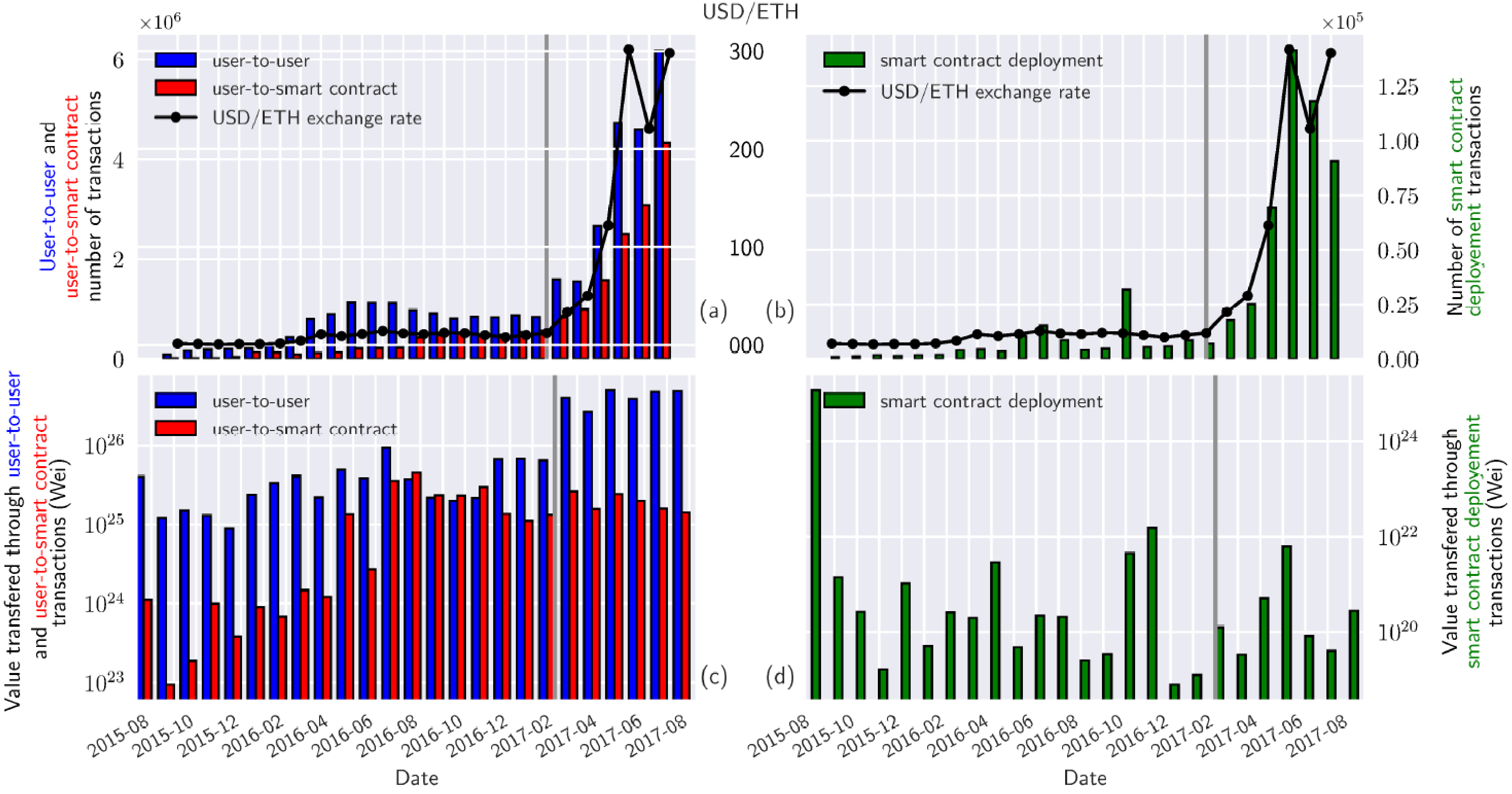
○ Q. Bai, C. Zhang, Y. Xu, X. Chen, and X. Wang, "**Evolution of Ethereum: a temporal graph perspective**," in IFIP Net. Conf., 2020.

○ X. T. Lee, A. Khan, S. S. Gupta, Y. H. Ong, and X. Liu, "**Measurements, analyses, and insights on the entire Ethereum blockchain network**," in WWW, 2020.

○ L. Zhao, S. S. Gupta, A. Khan, and R. Luo, "**Temporal analysis of the entire Ethereum blockchain network**," in WWW, 2021.

Graphs between Accounts:

- A. Anoaica and H. Levard, "**Quantitative description of internal activity on the Ethereum public blockchain**," in NTMS, 2018.



Graphs Based on Transaction of Ether:

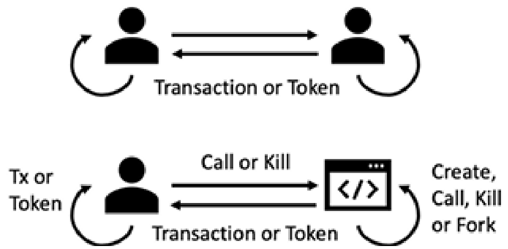
○ **Regular**, or **external transaction** denotes a transaction with the sender address being an EOA.

○ **Internal transaction** refers to a transfer that occurs when the sender address is a smart contract, e.g., a smart contract calling another smart contract or an EOA.

○ **Token transfer** is an event log for transfer of tokens only.

➢Token transfers can be considered as internal transactions. Internal transactions are not broadcast to the network in the form of regular transactions.

○ **Transaction Graph/ Money Flow Graph/ TransactionNet**



○ T. Chen, Y. Zhu, Z. Li, J. Chen, X. Li, X. Luo, X. Lin, and X. Zhang, "**Understanding Ethereum via graph analysis**," in INFOCOM, 2018.

○ J. Liang, L. Li, and D. Zeng, "**Evolutionary dynamics of cryptocurrency transaction networks: an empirical study**," PLoS ONE, vol. 13, no. 8, p. e0202202, 2018.

○ D. Guo, J. Dong, and K. Wang, "**Graph structure and statistical properties of Ethereum transaction relationships**," Inf. Sci., vol. 492, pp. 58–71, 2019.

○ S. Ferretti and G. D'Angelo, "**On the Ethereum blockchain structure: a complex networks theory perspective**," Concurr. Comput. Pract. Exp., vol. 32, no. 12, 2020.

○ D. Lin, J. Wu, Q. Yuan, and Z. Zheng, "**Modeling and understanding Ethereum transaction records via a complex network approach**," IEEE Trans. Circuits Syst., vol. 67-II, no. 11, pp. 2737–2741, 2020.

○ X. T. Lee, A. Khan, S. S. Gupta, Y. H. Ong, and X. Liu, "**Measurements, analyses, and insights on the entire Ethereum blockchain network**," in WWW, 2020.

○ L. Zhao, S. S. Gupta, A. Khan, and R. Luo, "**Temporal analysis of the entire Ethereum blockchain network**," in WWW, 2021.

Graphs Based on Transfer of Tokens:

- **Full ERC20 token transfer graph**
- **Individual ERC20 token transfer graphs**
- **Individual ERC721 token transfer graphs**
- **TokenNet/ Token transfer graph**
- **Token creator graph**
- **Token holder graph**

- S. Somin, G. Gordon, and Y. Altshuler, "**Network analysis of ERC20 tokens trading on Ethereum blockchain**," in Complex Systems, 2018.
- F. Victor and B. K. L¨uders, "**Measuring ethereum-based ERC20 token networks**," in Financial Cryptography and Data Security, 2019.
- Y. Chen and H. K. T. Ng, "**Deep learning Ethereum token price prediction with network motif analysis**," in ICDM Workshops, 2019.
- W. Chen, T. Zhang, Z. Chen, Z. Zheng, and Y. Lu, "**Traveling the token world: A graph analysis of Ethereum ERC20 token ecosystem**," in WWW, 2020
- Y. Li, U. Islambekov, C. G. Akcora, E. Smirnova, Y. R. Gel, and M. Kantarcioglu, "**Dissecting Ethereum blockchain analytics: what we learn from topology and geometry of the Ethereum graph**?" in SDM, 2020.
- B. Gao, H. Wang, P. Xia, S. Wu, Y. Zhou, X. Luo, and G. Tyson, "**Tracking counterfeit cryptocurrency end-to-end**," Proc. ACM Meas. Anal. Comput. Syst., vol. 4, no. 3, pp. 50:1–50:28, 2020.
- X. T. Lee, A. Khan, S. S. Gupta, Y. H. Ong, and X. Liu, "**Measurements, analyses, and insights on the entire Ethereum blockchain network**," in WWW, 2020.
- L. Zhao, S. S. Gupta, A. Khan, and R. Luo, "**Temporal analysis of the entire Ethereum blockchain network**," in WWW, 2021.
- D. Ofori-Boateng, I. Segovia-Dominguez, C. G. Akcora, M. Kantarcioglu, and Y. R. Gel, "**Topological anomaly detection in dynamic multilayer blockchain networks**," in ECML PKDD, 2021.
- S. Casale-Brunet, P. Ribeca, P. Doyle, and M. Mattavelli, "**Networks of Ethereum non-fungible tokens: a graph-based analysis of the ERC-721 ecosystem**," in Blockchain, 2021.

Graph Analysis on Blockchain Graphs

○ X. T. Lee, A. Khan, S. S. Gupta, Y. H. Ong, and X. Liu, "**Measurements, analyses, and insights on the entire Ethereum blockchain network**," in WWW, 2020.

○ L. Zhao, S. S. Gupta, A. Khan, and R. Luo, "**Temporal analysis of the entire Ethereum blockchain network**," in WWW, 2021.

Ethereum Network Properties

- Basic Network Properties
- Local Network Properties
- Global Network Properties
- Temporal Network Properties

Motivation

- Blockchain is a fascinating ecosystem of humans and autonomous agents.
- Not like conventional social networks, where the players are human users.
- Not like cryptocurrencies, where all interactions are transfer of value/asset.

Blockchain network is closer to the Internet or Web, where users interact with one another, as well as with programs.

We study a public permissionless blockchain network as a **complex system**, and we choose **Ethereum**, the most prominent blockchain network, for this purpose.

Ethereum

- Introduced an automation layer on top of a blockchain through contracts.
- Facilitates a decentralized computing environment across the blockchain.

Transaction-based state machine. Global state made up of accounts. Transfer of value/information between accounts cause transitions in the state. Recorded in the blockchain.

We target the **network of interactions** between the User and Contract accounts that make up the global state of Ethereum, and study them as **complex systems**.

Networks

1. **TraceNet**
   v : user and smart contract addresses
   a : all successful traces/transactions

2. **ContractNet**
   v : only smart contract addresses
   a : all successful traces/messages

3. **TransactionNet**
   v : user and smart contract addresses
   a : all successful transactions by users

4. **TokenNet**
   v : user and smart contract addresses
   a : all successful transaction of tokens

While **TraceNet** presents a global view of interactions, **ContractNet** focusses on the multi-agent network of contracts. While **TransactionNet** depicts all of basic ether transactions, **TokenNet** focusses on the rich and diverse token ecosystem.

Network Data

Source : Google Cloud Platform BigQuery
bigquery-public-data.Ethereum_blockchain.

Data extracted/mined : Block #0 till #7185508
Blocks recorded upto 2019-02-07 00:00:27 UTC
Seven different tables in the Ethereum dataset.

	Size of Dataset	Row Count
blocks	8 GB	7 185 509
contracts	15.7 GB	12 950 995
transactions	190 GB	388 018 489
traces	500 GB	974 766 498
logs	160 GB	289 552 838
tokens	11.4 MB	126 181
token transfers	58 GB	173 421 940

Data cleaning : Removing failed traces and handling Null addresses appropriately.

Basic Network Properties

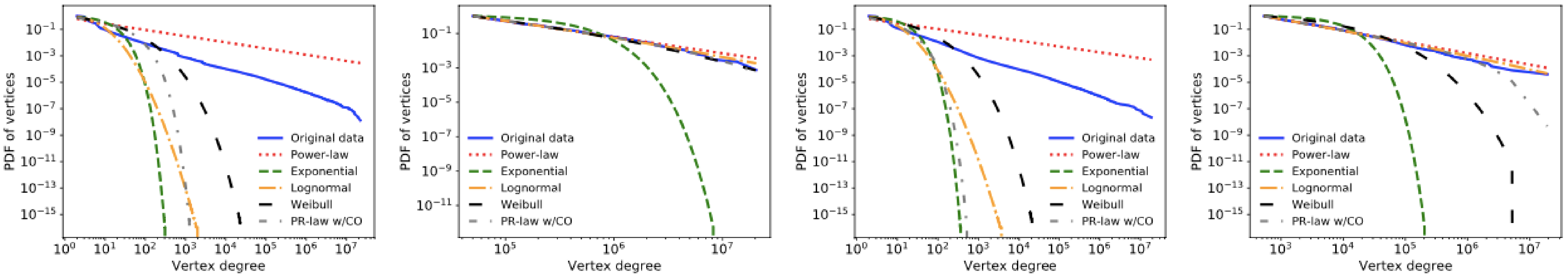
Vertices and Arcs, Self-Loops and Density

	# Vertices	MultiDigraph			Simple, undirected graph		
		# Arcs	# Self-loops (% of Arcs)	Density	# Arcs	# Self-loops (% of Arcs)	Density
TraceNet	75 807 179	768 813 599	3 036 915 (0.40%)	1.34×10^{-7}	191 901 321	178 241 (0.09%)	0.67×10^{-7}
ContractNet	11 332 750	317 967 546	**2 521 670 (0.79%)**	**24.8×10^{-7}**	19 608 452	**63 234** (0.32%)	3.05×10^{-7}
TransactionNet	45 527 529	388 018 489	515 245 (0.13%)	1.87×10^{-7}	128 368 878	115 007 (0.09%)	1.24×10^{-7}
TokenNet	30 429 099	173 421 940	326 557 (0.19%)	**1.87×10^{-7}**	93 844 445	36 950 (0.04%)	**2.03×10^{-7}**

We observe that self-loop percentage in ContractNet MultiDiGraph is significantly higher than that in the three other networks. Moreover, the number of self-loops in its MultiDiGraph is **almost 40 times** than that in its own simple, undirected graph, indicating that a lot of **smart contracts make multiple calls to itself**.

Local Network Properties

Vertex Degree Distribution



(a) TraceNet (b) ContractNet (c) TransactionNet (d) TokenNet

We compare power-law distribution model against (i) exponential, (ii) log-normal, (iii) power-law with exponential cutoff, and (iv) stretched exponential or Weibull.

We see that for our larger networks, TraceNet and TransactionNet, three of the four alternative heavy-tailed distributions are better fit than the power-law.

Local Network Properties

Indegree and Outdegree Correlation

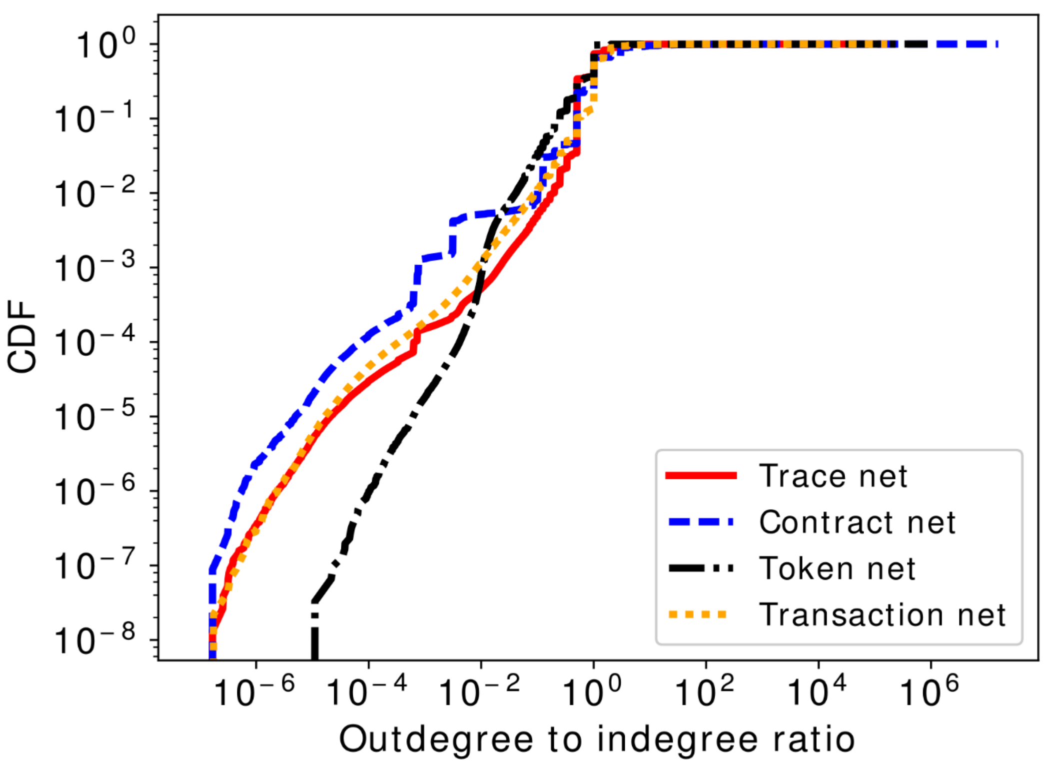
Indegree and outdegree of vertices in the four network MultiDiGraphs.

≈ 50% have similar in and out.

≈ 30% have significantly higher in (ICO smart contracts appear a lot in the to_address).

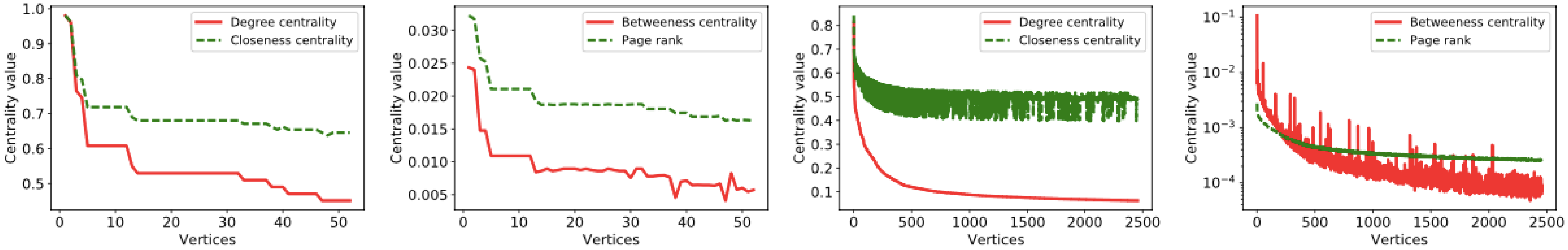
≈ 20% have significantly higher out (mining pools and mixers generally appear a lot in the from_address).

This is similar to the Web, involving hubs and authorities, and it is unlike the case of standard social networks.



Local Network Properties

Centrality Measures



(a) ContractNet (b) ContractNet (c) TokenNet (d) TokenNet

Vertex centrality aims at scoring, ranking, and identification of important vertices.

We identify the most central vertices from the innermost core of the largest strongly connected component and find that high-degree vertices in blockchain networks are also most central based on betweenness, closeness, and PageRank.

Global Network Properties

Reciprocity and Assortativity

Reciprocity: Measure of vertices being mutually linked in network.

Assortativity: Measure of vertices being linked to similar-degree ones.

Network (#vertices, #arcs)	Reciprocity	Assortativity
TraceNet (76M, 198M)	0.06	-0.13
ContractNet (11M, 22M)	**0.21**	**-0.64**
TransactionNet (46M, 130M)	0.03	-0.12
TokenNet (30M, 95M)	0.03	-0.13

Unlike social networks, all four of our blockchain networks are Disassortative.
Negative assortativity implies relatively more scenarios of addresses (vertices) with different degrees transacting with each other in the blockchain networks.

Global Network Properties

Strong and Weakly Connected Components

Simple, directed networks (#vertices, #arcs)	# Strongly connected components	Largest strongly connected component (#vertices, #arcs)	# Weakly connected components	Largest weakly connected component (#vertices, #arcs)
TraceNet (**76M, 198M**)	35 215 962	40M, 116M	**7 324**	**76M, 192M**
ContractNet (**11M, 22M**)	**9 013 144**	2M, 4M	**12 555**	**11M, 20M**
TransactionNet (**46M, 130M**)	15 560 831	30M, 76M	**8 181**	**46M, 128M**
TokenNet (**30M, 95M**)	16 980 001	13M, 56M	**54 271**	**30M, 94M**

Number of WCC is significantly lesser than the number of SCC in their respective networks, due to lesser bidirectional edges between majority pairs of vertices.

ContractNet has the least # of SCC in the networks, indicating relatively stronger connectivity within smart contracts. Similar to the Web, the blockchain networks have a single, large SCC, with about 98% of the remaining vertices within reach.

Global Network Properties

Core Decomposition

k-core is the maximal subgraph, where each vertex is connected to at least **k** other vertices within the subgraph.

Largest Weakly Connected Component (#vertices, #arcs)	# Cores	Innermost core (#vertices, #arcs)
TraceNet (76M, 192M)	98	(221, 12 058)
ContractNet (11M, 20M)	**264**	**(1071, 143 352)**
TransactionNet (46M, 128M)	105	(682, 55 926)
TokenNet (30M, 94M)	**218**	(475, 57 124)

ContractNet and TokenNet have larger core indices for vertices in the innermost cores, indicating higher density of their innermost cores. ContractNet's innermost core is the largest, implying more vertices participating in denser substructures.

Global Network Properties

Triangles, Transitivity, Clustering Coefficients

Transitivity is quite low. This suggests that in the blockchain networks, we do not have a conducive environment for creation of triangles. **Indeed, non-social networks have lower transitivity coefficients.**

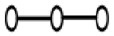
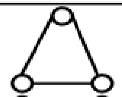
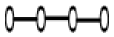
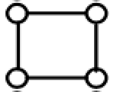
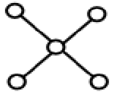
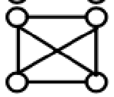
	Largest strongly connected comp. (Simple, undirected)			Largest weakly connected comp. (Simple, undirected)		
	# Triangles	T	C	# Triangles	T	C
TraceNet	4 008 794	10.0×10^{-7}	0.099	5 813 165	1.2×10^{-7}	0.077
ContractNet	405 265	$\mathbf{38.0\times10^{-7}}$	**0.212**	871 359	6.7×10^{-7}	0.078
TransactionNet	1 908 138	8.3×10^{-7}	0.064	4 550 517	$\mathbf{12.4\times10^{-7}}$	0.100
TokenNet	2 803 894	8.6×10^{-7}	0.209	5 296 640	5.5×10^{-7}	**0.175**

High-degree vertices are often “loner-star”, that is, connected to mostly low-degree vertices, resulting in lack of community structure in blockchain graphs.

Global Network Properties

Higher-Order Motifs Counting

The most frequent motifs in the blockchain graphs are primarily chain and star-shaped. Counts for more complex patterns, e.g., cliques and cycles, are less.

	#	Motif density		#	Motif density
	13 669	1×10^{-1}		2 214	2×10^{-2}
	17 081	3×10^{-3}		60 297	9×10^{-3}
	387 816	12×10^{-3}		2 578	4×10^{-4}

We check the density of a motif, the ratio of its count to its count in a complete graph having same number of vertices as the innermost core. The densities for more complex patterns are quite less, indicating lack of community structure.

Global Network Properties

Articulation points, Adhesion, Cohesion, Average path lengths, Radius, Diameter

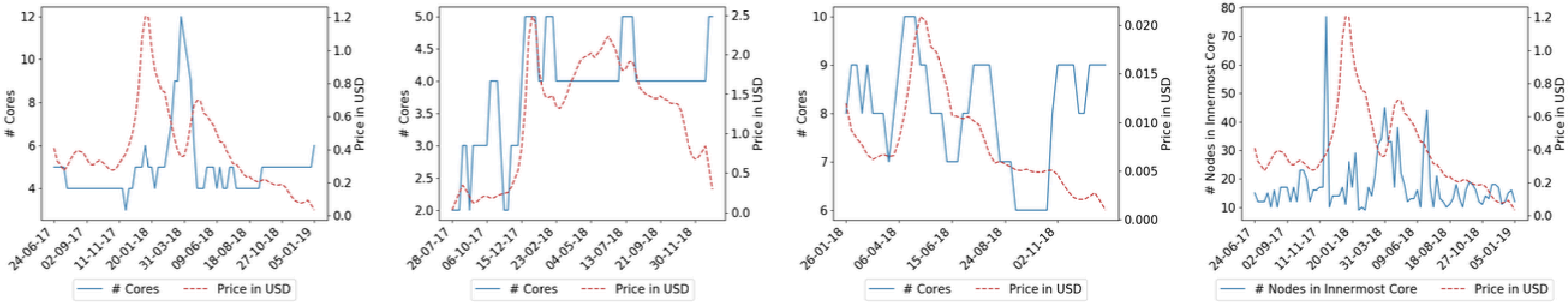
	# Articulation points (% of all vertices)	Largest strongly conn. comp.		Largest weakly conn. comp.		Largest weakly connected component		
		Adhesion	Cohesion	Adhesion	Cohesion	Avg. path length	Radius	Diameter
TraceNet	1 214 137 (1.6%)	1	1	1	1	5.25	5 002	8 267
ContractNet	28 309 (0.2%)	1	1	1	1	5.94	14	27
TransactionNet	1 337 527 (2.9%)	1	1	1	1	5.33	5 002	8 267
TokenNet	75 513 (2.5%)	1	1	1	1	3.87	82	164

Adhesion and Cohesion for all blockchain networks are 1, indicating that removal of the only one vertex or only one arc disconnects the respective SCCs and WCCs.

Interestingly, similar to social networks, blockchain graphs are also small-world. However, in both our larger networks, TraceNet and TransactionNet, there are vertices which are far apart, making the radius and the diameter quite large.

Temporal Network Properties

Progress of Core Decomposition in Token Networks



(a) Bancor : Number of Cores vs. Price (b) Binance Coin : Number of Cores vs. Price (c) Zilliqa : Number of Cores vs. Price (d) Bancor : Vertices in Inner Core vs. Price

We study temporal evolution of the number of cores in token subgraphs against the corresponding evolution of price of the token in the cryptocurrency market. Observations clearly show a significant relationship between activity and price.

Summary of Observations

the Web

- In/Out-degree characteristics are very similar to the Web (hub/authority).
- The blockchain networks are disassortative, having very low transitivity.
- Complex motifs occur quite less, indicating lack of community structure.
- Removal of one vertex or arc can disconnect the entire largest SCC/WCC.

social network

- Blockchain networks are surprisingly small-world and well-connected.

both networks

- Networks contain a single, large SCC, with 98% of the vertices reachable.
- ContractNet and TokenNet yield larger core indices for vertices in the innermost cores, indicating higher density of their innermost cores.

financial

- Significant relationship between temporal relationship of inner cores of prominent token networks and the price of the tokens in the market.

https://github.com/sgsourav/blockchain-network-analysis

Future work may include **analysis of prominent token networks** in terms of activity signatures to forecast trading behavior and token prices. Identifying **influential vertices and complex motifs** may also detect fraudulent activities.

Motivation and Research Questions

- Investigate the **evolutionary nature of Ethereum interaction networks** from a temporal graph perspective

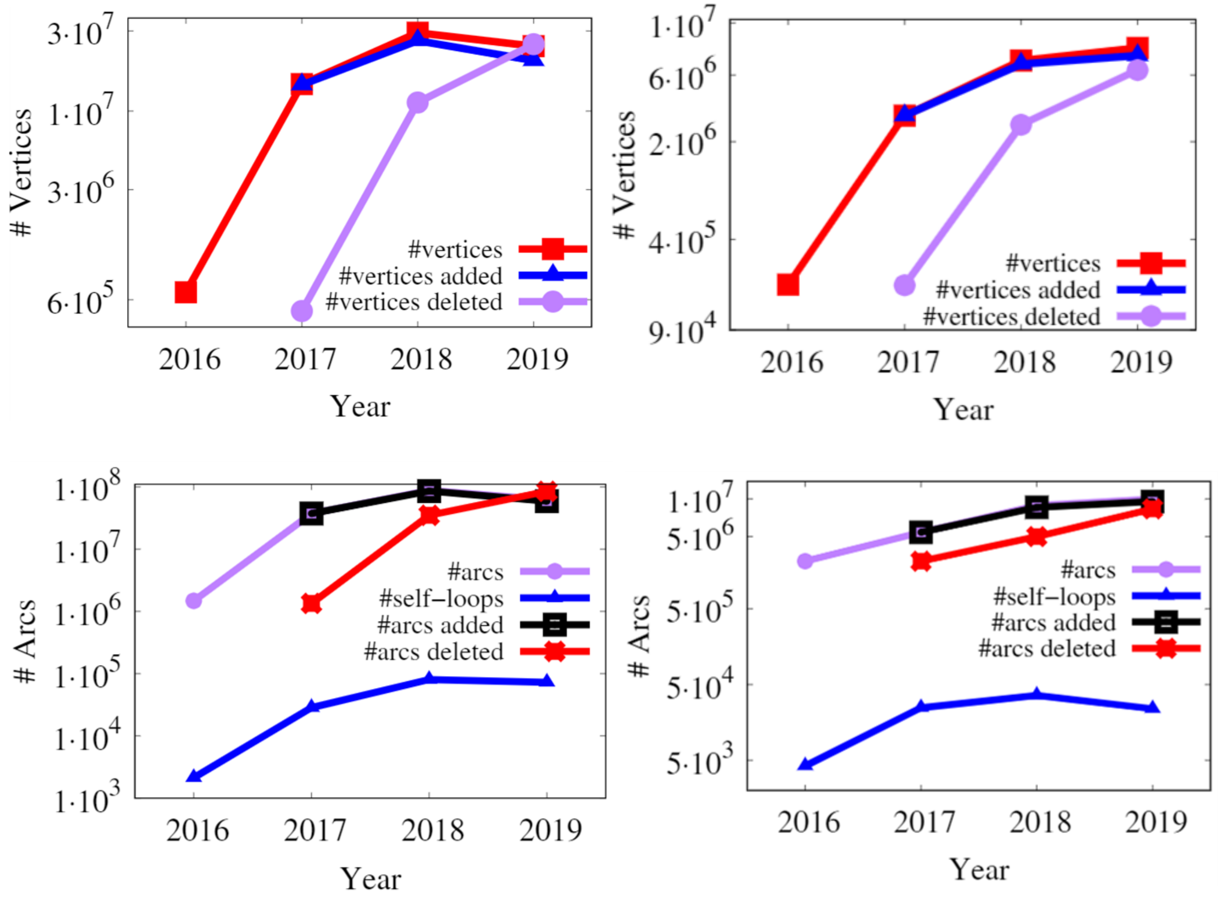
L. Zhao, S. S. Gupta, A. Khan, and R. Luo, "**Temporal analysis of the entire Ethereum blockchain network**," in WWW, 2021.

- Address 3 main questions:
  - ➢ How do Ethereum network evolve over time?
  - ➢ How network properties changes over time, what is the right "time granularity" for such temporal analysis?
  - ➢ Detect meaningful communities and forecast the survival of communities in succeeding months.

Evolution of Ethereum Network (Vertex)

- The number of new vertices and arcs added is almost of the same order of total number of vertices and arcs at that time => Ethereum interaction networks growing at a fast speed. (highly active network).

- Vertices which are disappeared keep increasing.



(a) TransactionNet (b) ContractNet

Network Growth Model

The increasing percentage (3rd column) indicates:

- As the Ethereum network matures, more accounts remain active.
- And more than half of new vertices participate in interaction with old vertices.

Table 3: TransactionNet: New vertices connecting with old vertices

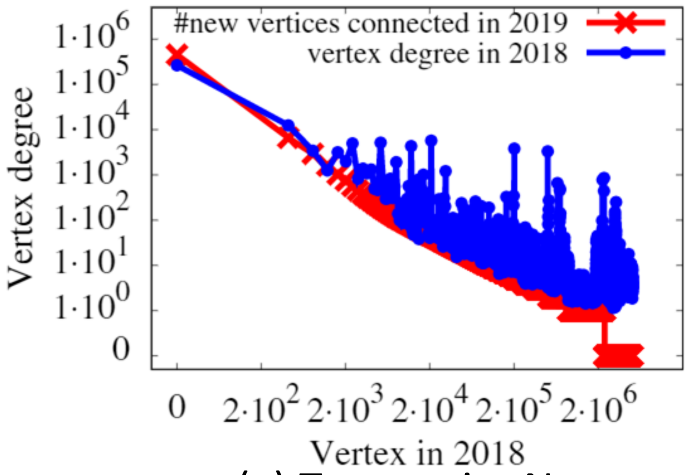
year	# old vertices	# new vertices	# new vertices with arc to old vertices (% of new vertices)	# new vertices without arc to old vertices (% of new vertices)
2017	163982	14789934	5646964 (38.18%)	9142970 (61.82%)
2018	3599770	28583252	14279239 (49.96%)	14304013 (50.04%)
2019	5060613	21240780	14807280 (69.71%)	6433500 (30.29%)

Table 4: ContractNet: New vertices connecting with old vertices

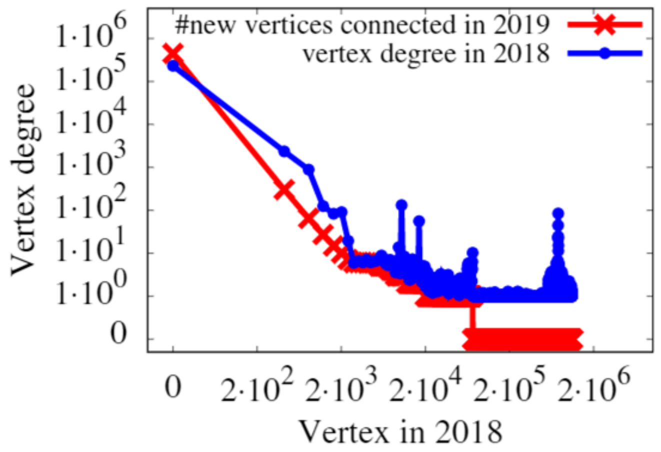
year	# old vertices	# new vertices	# new vertices with arc to old vertices (% of new vertices)	# new vertices without arc to old vertices (% of new vertices)
2017	1859	3070553	182920 (5.96%)	2887633 (94.04%)
2018	426000	7196954	2927928(40.68%)	4269026 (59.32%)
2019	1108567	8266061	6086678(73.63%)	2179383 (26.37%)

Network Growth Model

- Correlation between old vertex degree in previous year (2018) to its number of new connections in the current year (2019).
- High degree vertices are highly likely to have more new vertex connections in next year.
- The observation indicates that the Ethereum graphs follow the **preferential attachment** growth model.



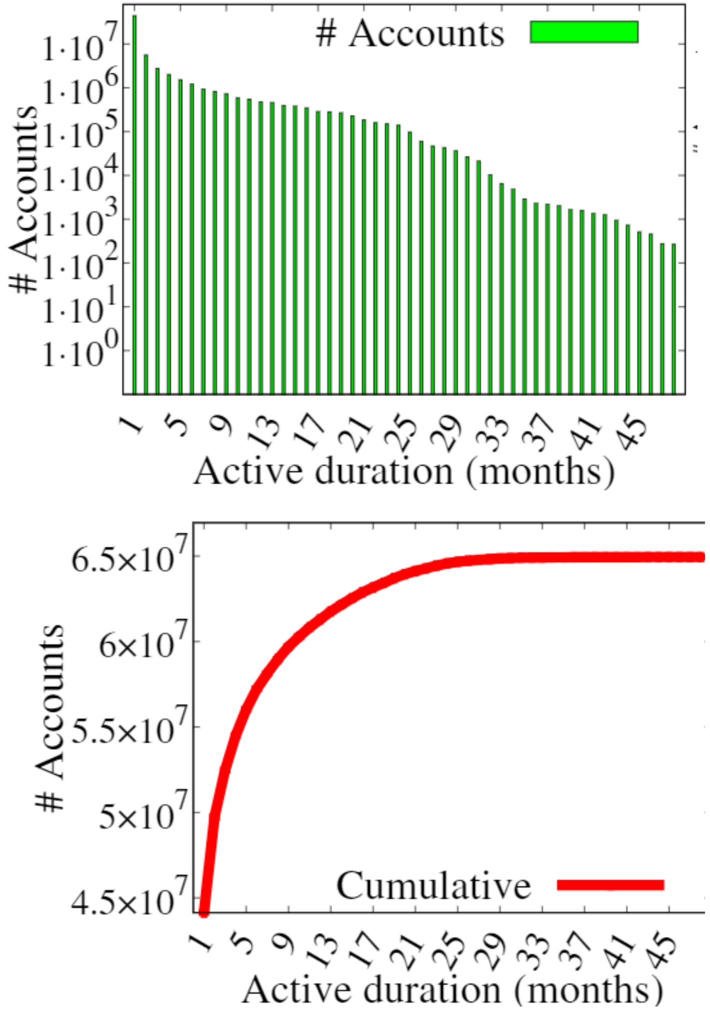
(a) TransactionNet



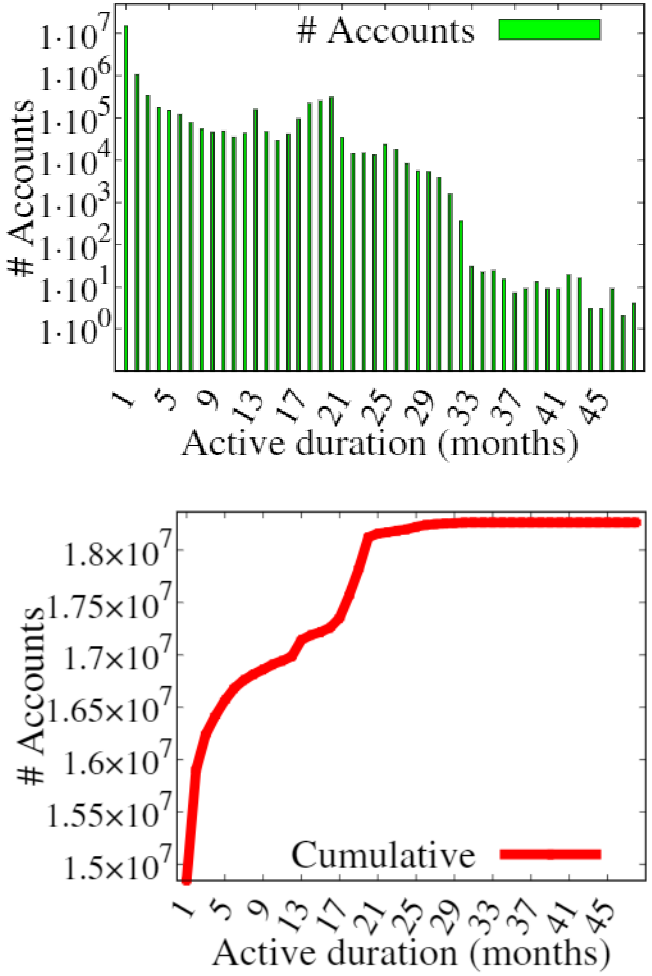
(b) ContractNet

Average Activity Period of Vertices

- **Active period** = duration (month) from its first transaction to the last transaction between Jan 2016 and Dec 2019.
- **ContractNet:** 91% has no more than 6 month active period.
- **TransactionNet:** Longer active period.
- In general, 88% of accounts have an active period of no more than 6 months, and up to 68% of accounts are only active within a month.



TransactionNet



ContractNet

Temporal Evolution of Network Properties

- Investigate network properties changes over time to understand how the network is connected and changed over time.
- Reveal any anomaly (beyond average) occurred in a specific time duration.
- A good time granularity as the shortest time duration by which we can detect an anomaly.

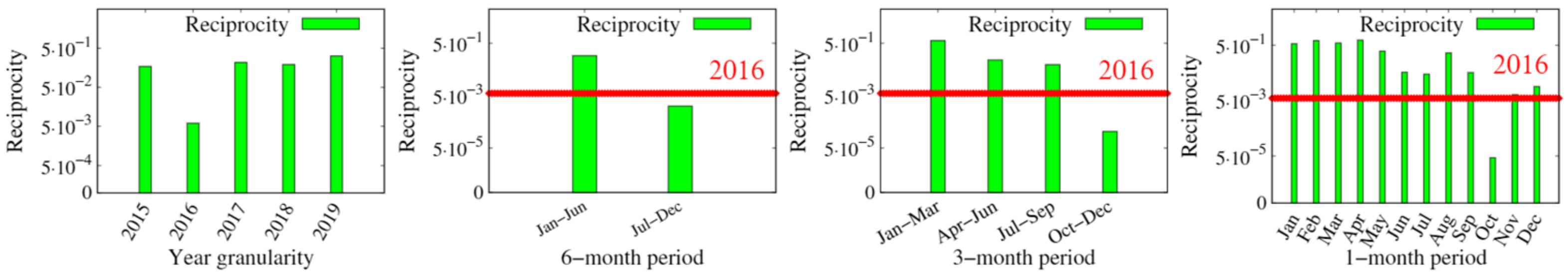


Figure 8: Time granularity analysis for reciprocity; ContractNet 2016

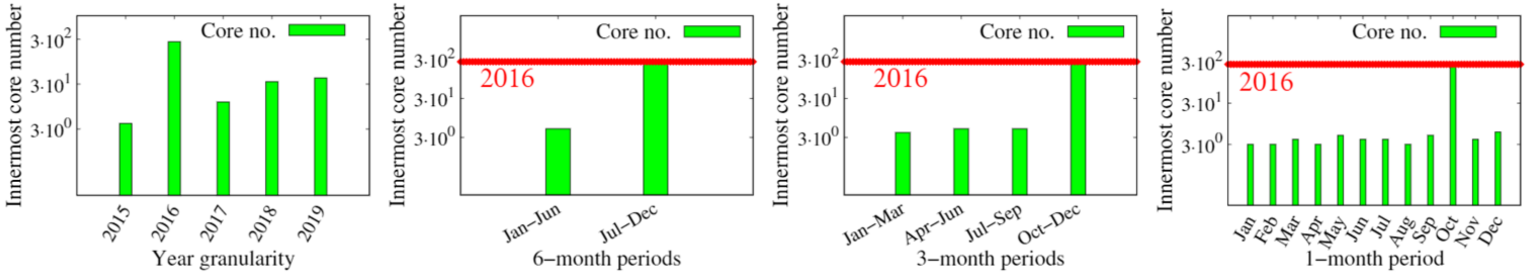


Figure 10: Time granularity analysis for core number in the innermost core; ContractNet 2016

Temporal Evolution of Network Properties

- **Oct 2016:** Plenty of positive news on Ethereum in the media → a lot of tokens were deployed on the network, which increased the number of one-directional arcs to the token contracts.

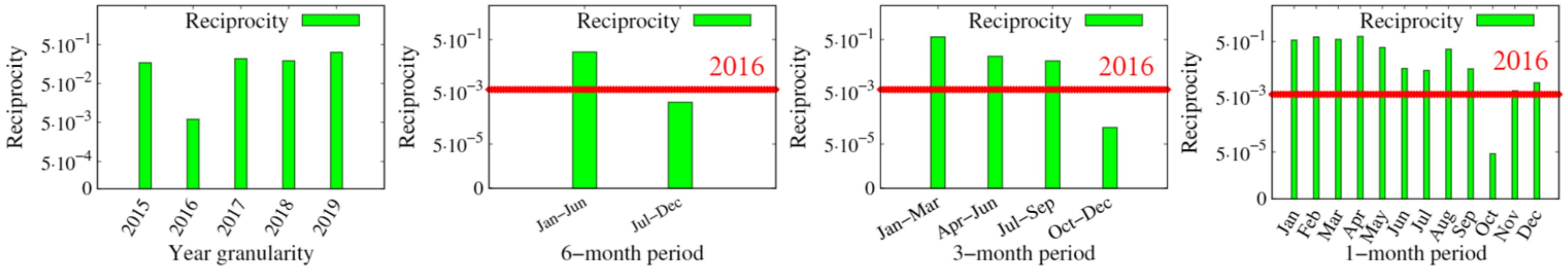


Figure 8: Time granularity analysis for reciprocity; ContractNet 2016

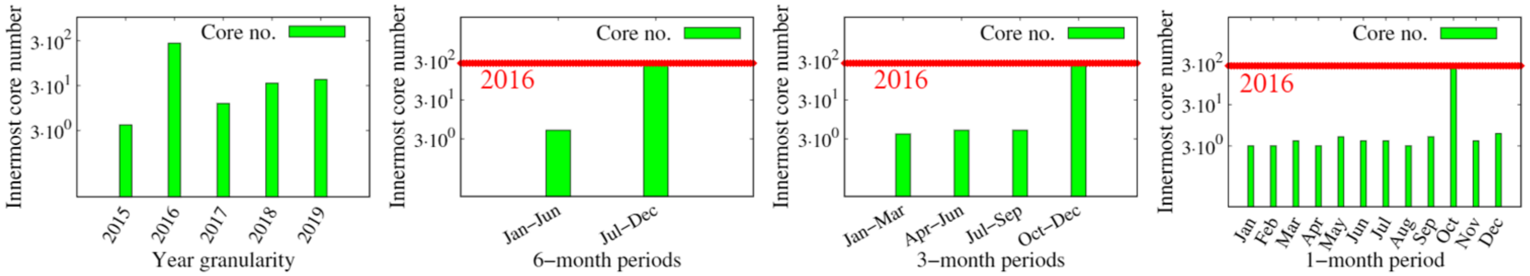
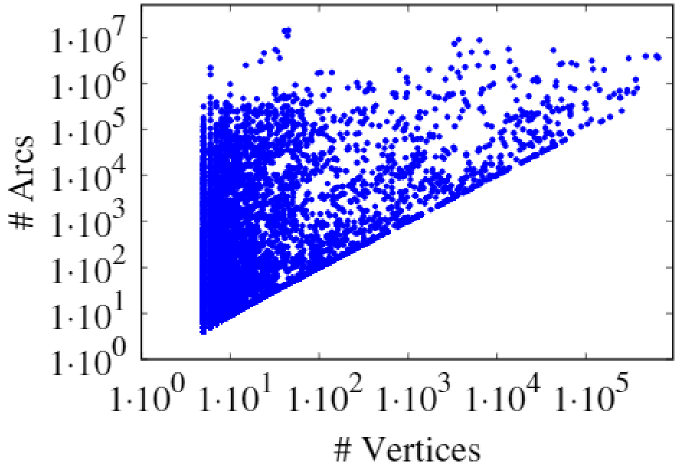


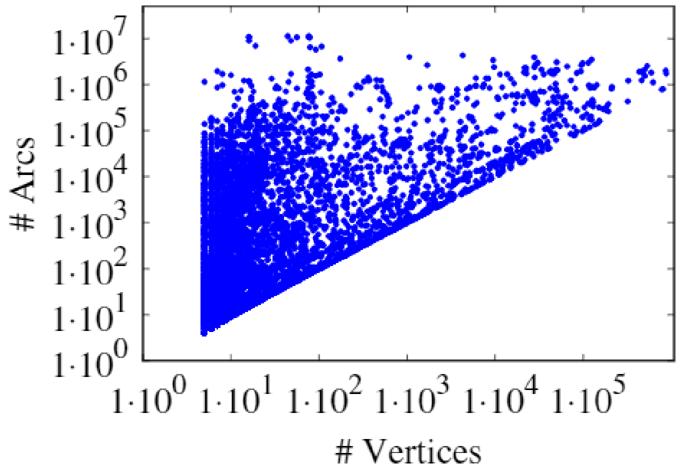
Figure 10: Time granularity analysis for core number in the innermost core; ContractNet 2016

Detection of ContractNet Communities

- Multilevel algorithm scales well over large-scale datasets and produce good-quality communities.
- Consider multi, undirected version of graph .
- # vertices and arcs in each community obtained over ContractNet 2018 and 2019 networks.
- The size of the communities follows power-law: **a few large communities followed by a long-tail of remaining small communities.**



(a) 2018

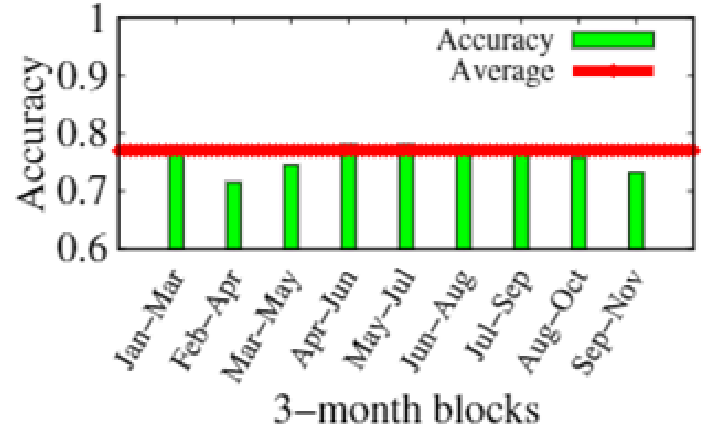


(b) 2019

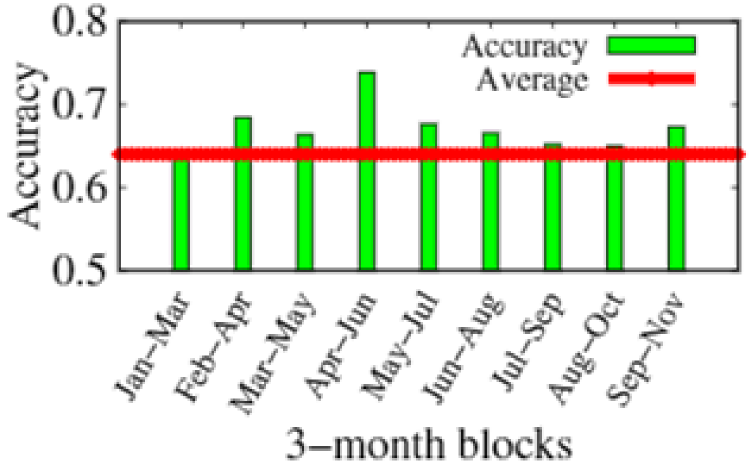
V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre. 2008. **Fast unfolding of communities in large networks.** Journal of Statistical Mechanics: Theory and Experiment 2008, 10 (2008), 10008.

Community Continuation Prediction

- Data preparation: window size of 3 months and slide stride of 1 month.
- Training dataset: the network properties of communities existing in 3-month period dataset.
- Aim: predict whether the communities still exists in next 1 month.
- Model: Logistic Regression & Random Forest.



Logistic Regression prediction accuracy for ContractNet 2019



Random Forest prediction accuracy for ContractNet 2019

Summary of Observation

- Ethereum interaction network grows at a fast speed.
- Networks follow the preferential attachment growth model.
- User accounts remain active much longer than smart contracts.
- Reveal anomalies occurred in a specific time duration and correlate them with external 'real-life' aspects of network.
- Detect meaningful communities in Ethereum network using multilevel algorithm.
- Forecast the continuation of communities in succeeding months leveraging on the relevant graph properties and ML models. Achieving up to 77% correct predictions for continuation.

https://github.com/LinZhao89/Ethereum-analysis

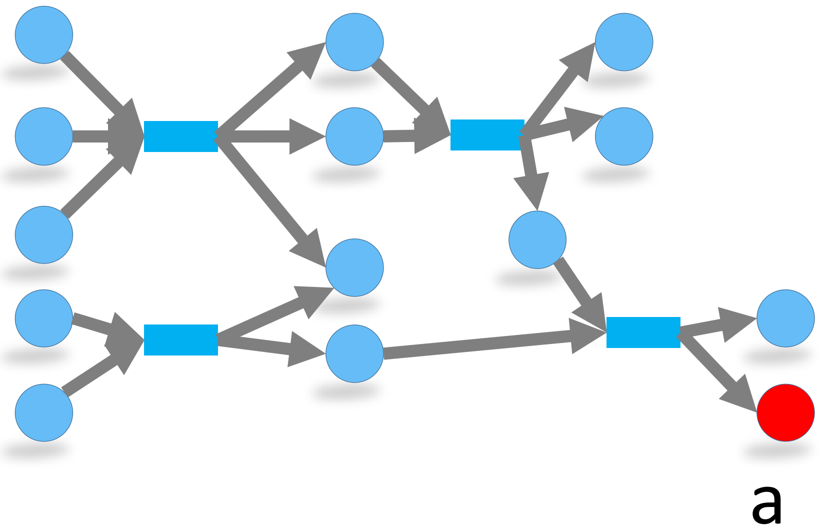
Address Clustering, Coin Mixing, Traceability and Obfuscation

Coin-mixing, Obfuscation, and Money Laundering

- Why? Foremost, ordinary citizens need privacy in cryptocurrency.
- Criminals need to sell their coins for fiat currency – on online exchanges which require customer identification.
- Law enforcement can find the person behind an address by asking for customer information from exchanges.
- Criminals need to launder their coins before they sell them.

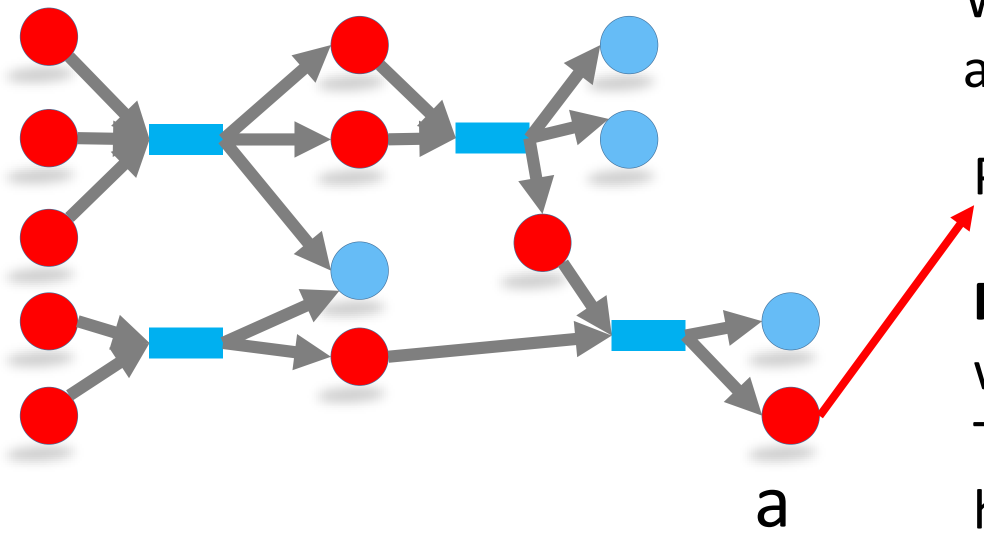
How to not get caught when you launder money on blockchain?
CG Akcora, S Purusotham, YR Gel, M Krawiec-Thayer, M Kantarcioglu
arXiv preprint arXiv:2010.15082

Clustering on UTXO Blockchains



Where do the bitcoins at address a come from?

Clustering on UTXO Blockchains



Where do the bitcoins at address a come from?

Possibly, from nine addresses!

Fungibility: Is a specific bitcoin worth a bitcoin everywhere? Taint analysis studies a bitcoin's history

- Can we tell which addresses are controlled by the same user, entity, organization?
- In order to answer this question, we need to link addresses.

https://twitter.com/cuneytgurcan/status/1361354903885553664

The **Nile Fallacy** is the false belief that Bitcoin is more traceable than fiat money.

Anyone who has parsed, and mined Bitcoin data will tell you that

…finding out the source of a drop of Nile water at Alexandria is not easier than finding the source of a bitcoin.

The drop may have come from 13 countries in the Nile basin.

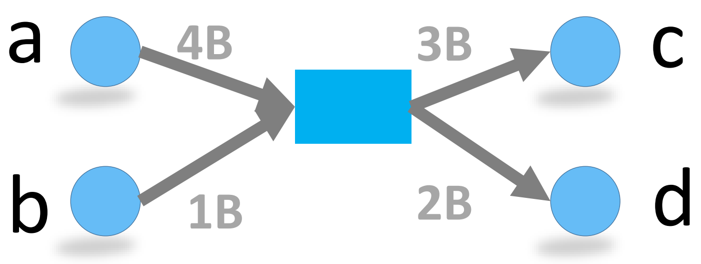
A bitcoin may have come from 70% of all Bitcoin transactions.

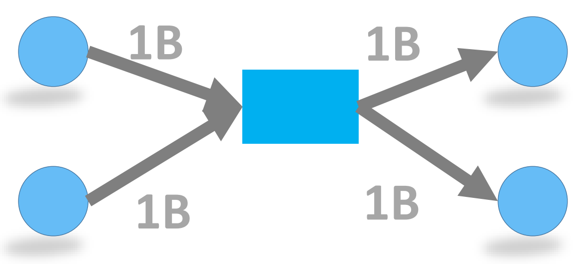
> Research many-to-many Bitcoin transactions to find out why.

The **Nile Fallacy** by Cuneyt Akcora

Heuristics

Heuristics are used to detect which input and output addresses are controlled by the same user.





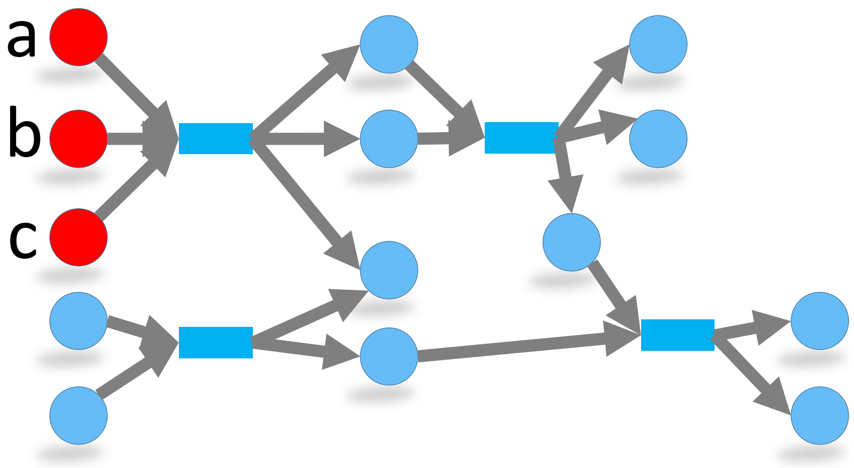
Considering amounts may help in basic cases (at least some coins at c and d came from a).

Schemes exist to use multiple rounds of flows with equal amounts to hide tracks.

Meiklejohn, Sarah, Marjori Pomarole, Grant Jordan, Kirill Levchenko, Damon McCoy, Geoffrey M. Voelker, and Stefan Savage. **A fistful of bitcoins: characterizing payments among men with no names**. In *Proceedings of the 2013 conference on Internet measurement conference*, pp. 127-140. ACM, 2013

Heuristics

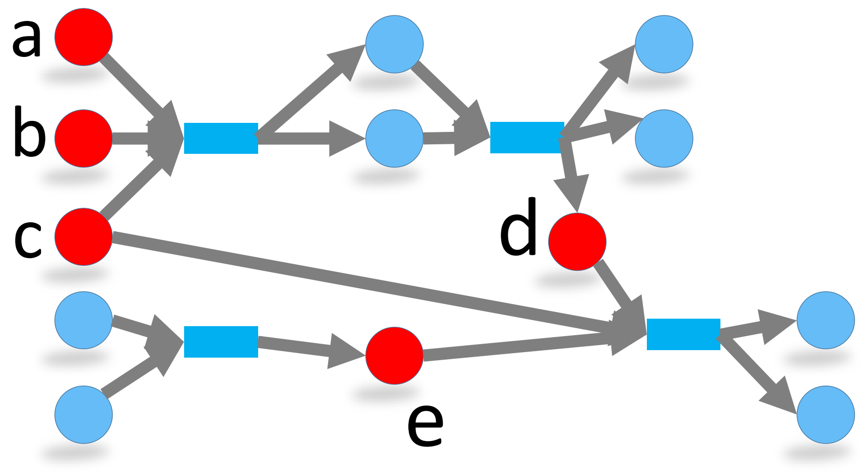
1- **Idioms of Use**: posits that all input addresses in a transaction should belong to the same entity because only the owner could have signed the inputs with the associated private keys.



Addresses a, b, and c belong to the same user.

Heuristics

2- **Transitive Closure**: extends Idioms of Use: if a transaction has inputs from a and b, whereas another transaction has from a and c, b and c belong to the same user.



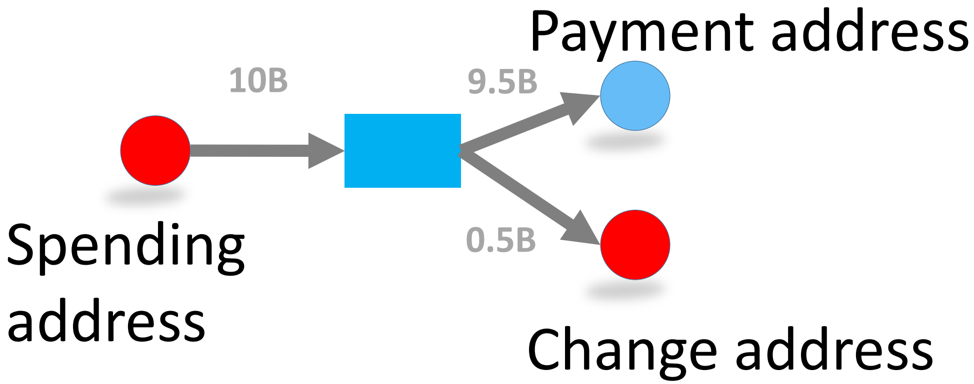
Addresses a, b, c, d, and e belong to the same user.

Heuristics

The heuristic posits that the one-time change (output) address— if one exists— is controlled by the same user as the input addresses.

3- **Change address**: the following four conditions must be met:

(1) the output address has not appeared in any previous transaction;
(2) the transaction is not a coin generation;
(3) there is no self-change address in the outputs;
(4) all the other output addresses in the transaction have appeared in previous transactions.



Traceability Problems and Privacy Coins

- Privacy coins break the mapping between input-output addresses, and even hide the transaction amounts.

Monero

- **Monero** (April 2014) uses ring signatures and allows users to mix other transaction outputs as (fake) inputs, so that the mapping between inputs and outputs are blurred.

- Transaction structure is transaction output based (TXO), amounts could be visible or hidden. Alphabay adopted Monero in 2016.

Confidential Transaction — amounts are not disclosed.

Inputs (3)

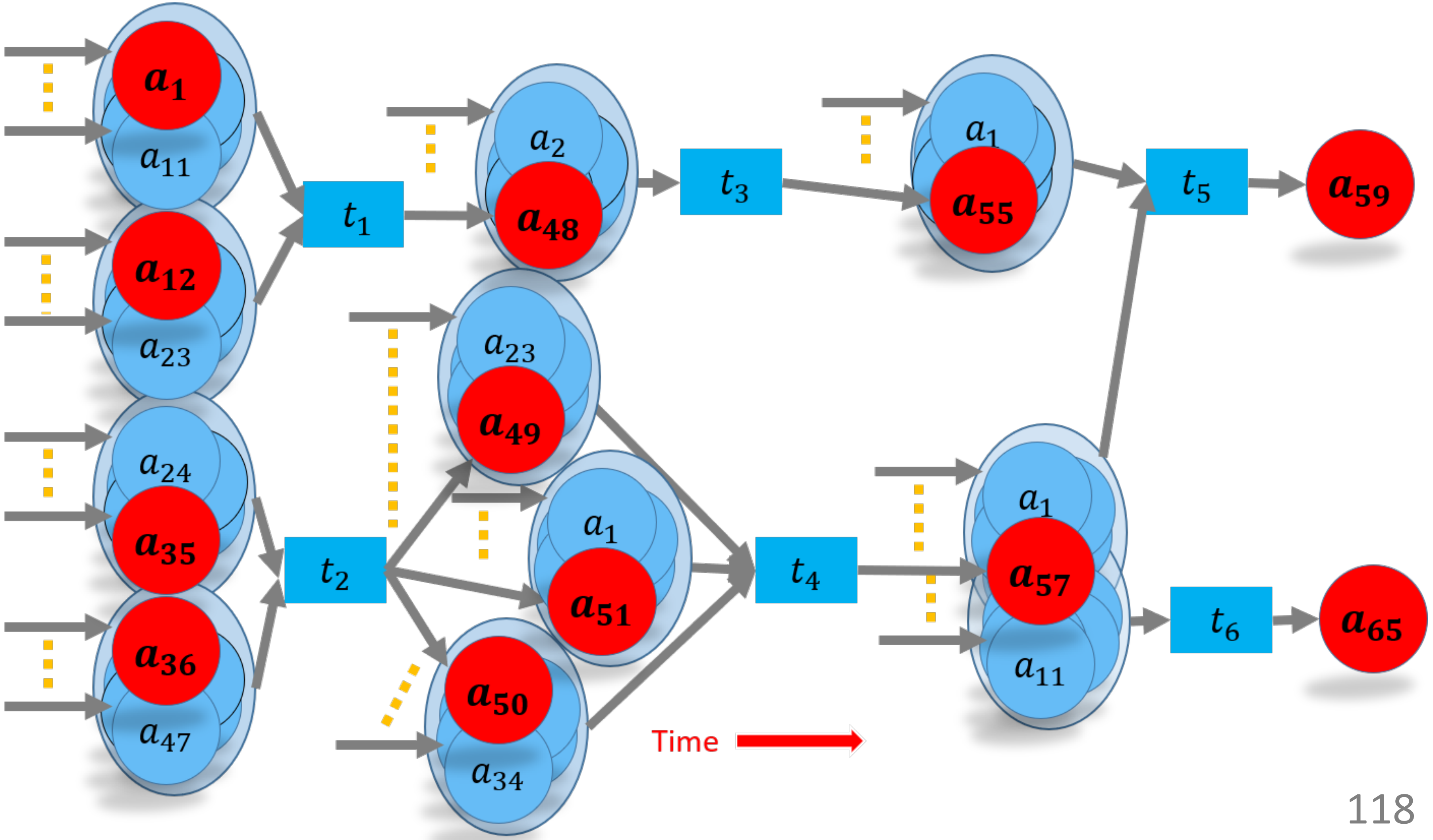
	Amount	Key Image
+	0.008000000000	d582442d895e2bea7a3c605dab0ab2fdc89dc509829087e29ca9cd2fceb5431f
+	0.000000000000	7c2874b22e49428ed77546fb8b9e56aa8624cc201718acc1ca1845466d13bc88
+	0.010000000000	572e2ac6a50c01b51f3eb12a030eb0c556eb1669b0fe73f030ade5d471b0831d

Outputs (2)

Amount	Public Key
0.000000000000	95c16aef66d1eaf1b3db676b9e3f68579b329c39f327be39fc627a2325a6e1bf
0.000000000000	8201c43798760afe6ab42f7b4083bcb1d7f9f50c1b9b2d564fa66875ecd9d185

Monero

Hiding transaction amount, sender and receiver address behind mixins. Reds are actual used addresses, blues are mixins.



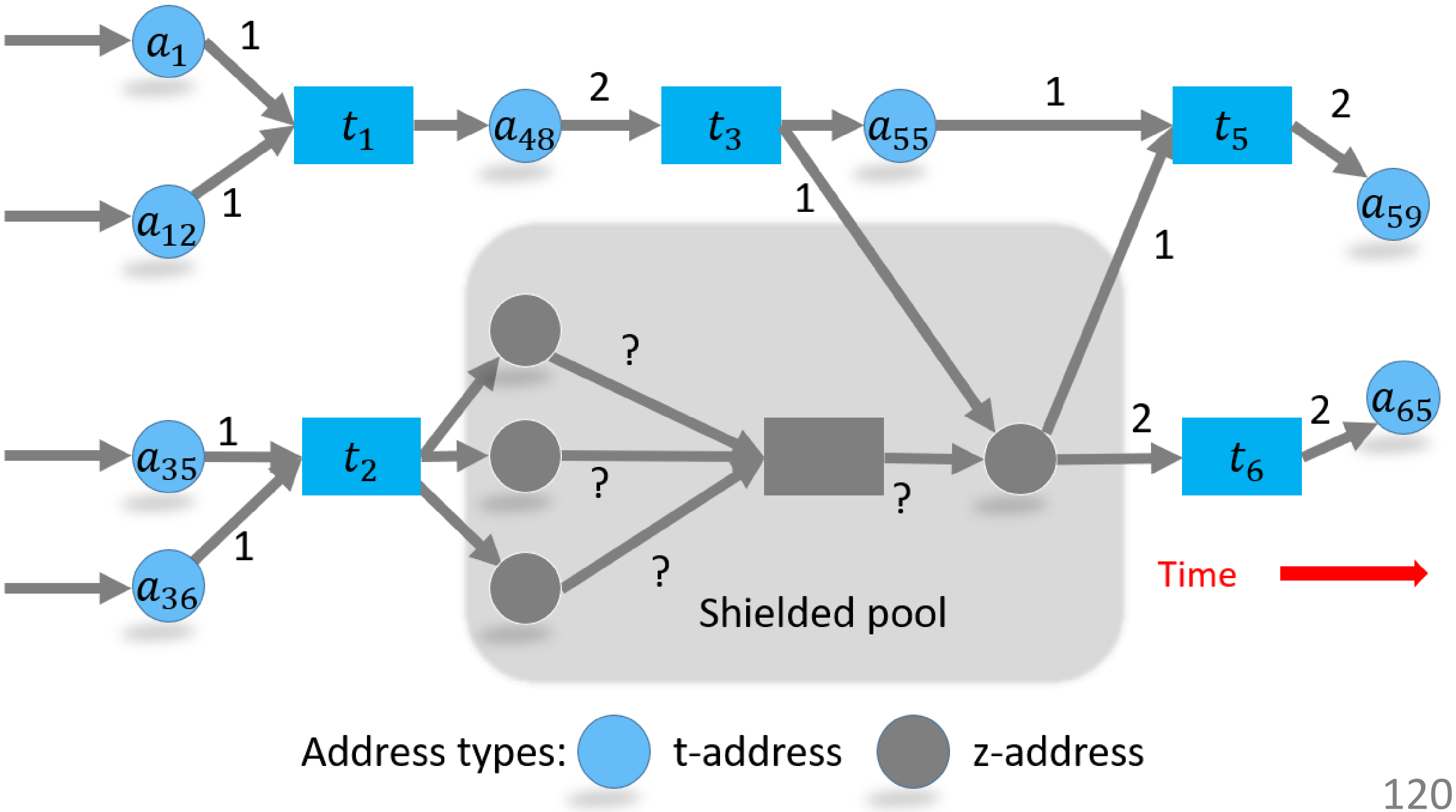
ZCash

Zcash can hide both transaction amounts and user entities, however less than 10% of all transactions were done by using z-addresses.

Kappos, G., Yousaf, H., Maller, M. and Meiklejohn, S., 2018. **An empirical analysis of anonymity in zcash**. In *27th USENIX Security Symposium (USENIX Security 18)* (pp. 463-477).

Zcash

Hiding transaction amount, sender and receiver address behind zero knowledge proofs.



Obfuscation Efforts

- Obfuscation: hiding coin movements in the network to finally cash out of the system by using an online exchange.

- Three regimes with increasing sophistication:
  - 2009-2013: Hiding patterns. Assumes that analyst cannot trace payments in the large network,
  - 2013-now: Coin-mixing,
  - 2018-now: Shapeshifting. Moving coins to privacy coins and bringing them back.

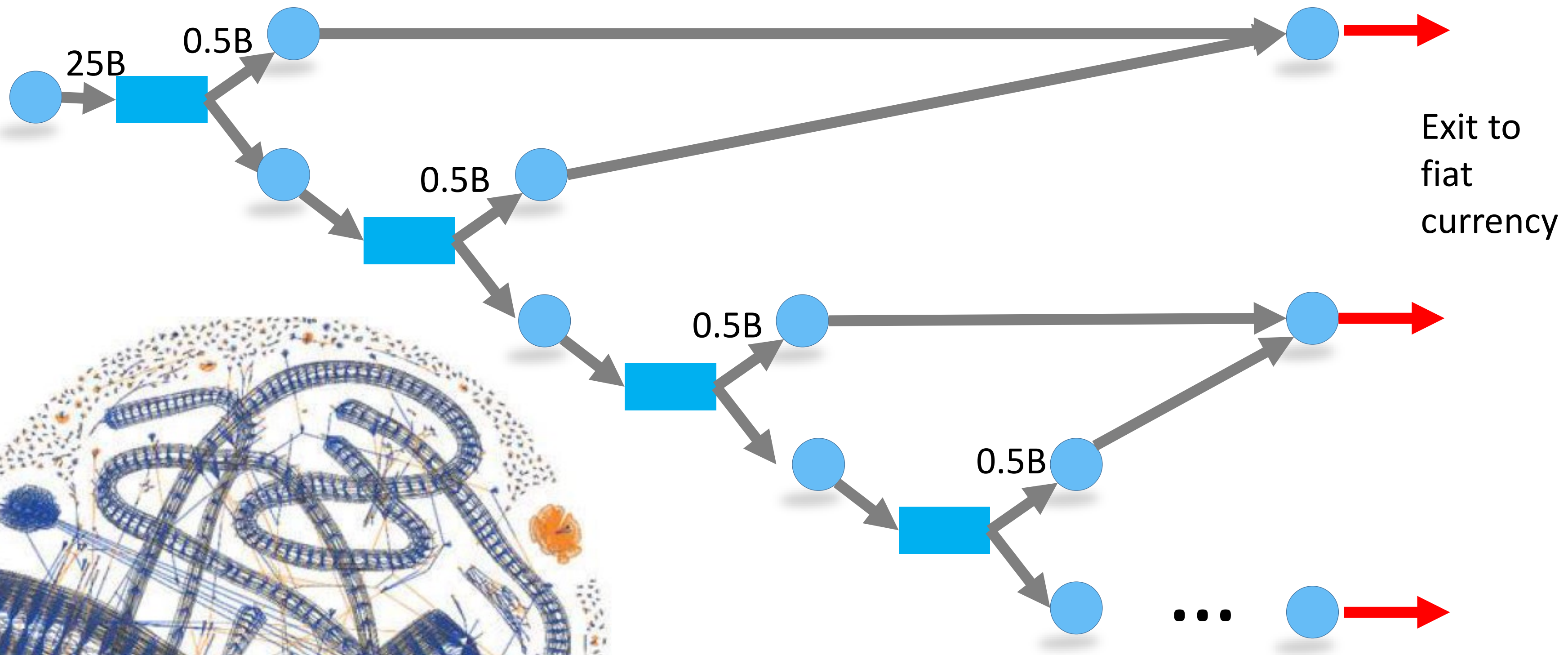
Narayanan, Arvind, and Malte Möser. **Obfuscation in bitcoin: Techniques and politics**. arXiv preprint arXiv:1706.05432 (2017).

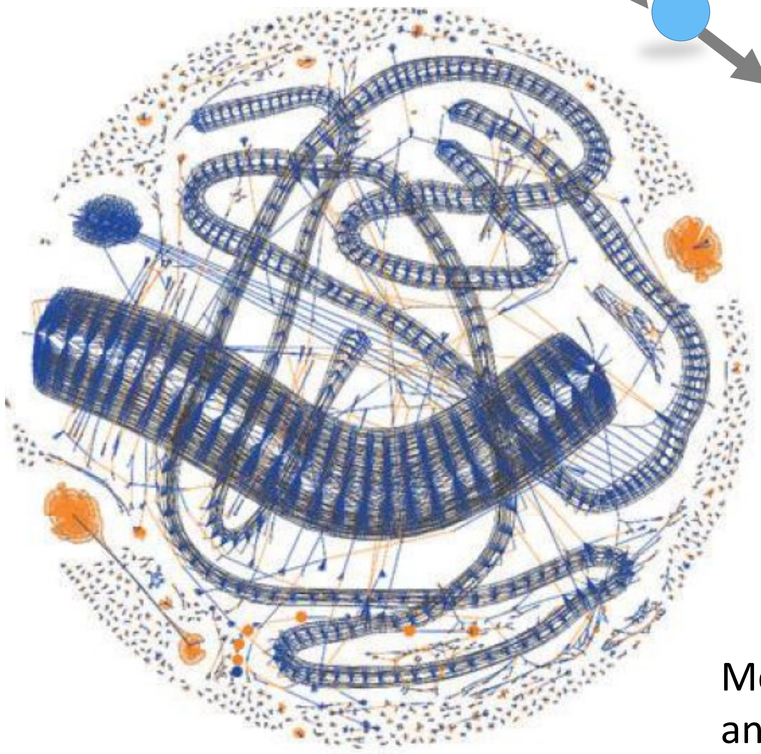
Obfuscation Efforts 1 – Peeling Chains

- In a peeling chain, a single address begins with a relatively large amount of bitcoins.
- A smaller amount is then "peeled" off this larger amount, creating a transaction in which a small amount is sent to one address and the remainder is sent to a one-time change address.
- This process is repeated— potentially for hundreds or thousands of hops— until the larger amount is pared down.

Di Battista, Giuseppe, Valentino Di Donato, Maurizio Patrignani, Maurizio Pizzonia, Vincenzo Roselli, and Roberto Tamassia. **Bitconeview: visualization of flows in the bitcoin transaction graph**. In Visualization for Cyber Security (VizSec), 2015 IEEE Symposium on, pp. 1-8. IEEE, 2015.

Obfuscation Efforts 1 – Peeling Chains





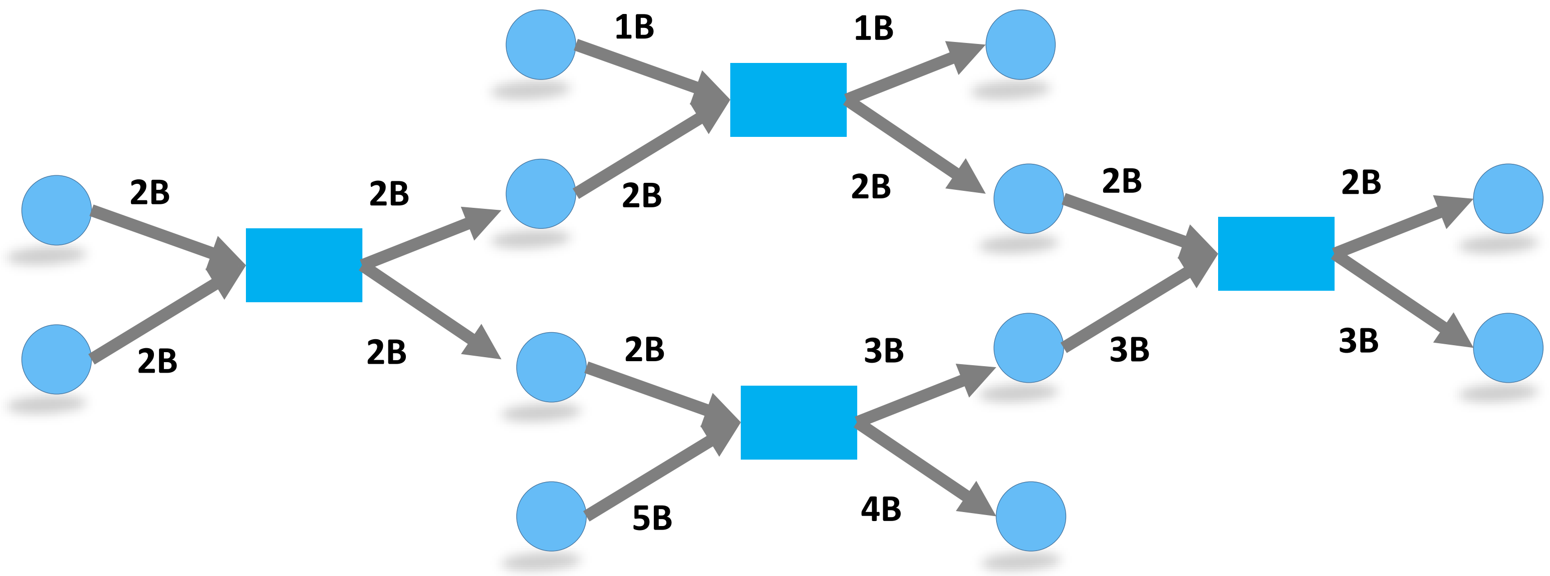
Repeated patterns are frequently found on the Bitcoin blockchain (spam transactions in the figure)

McGinn, Dan, David Birch, David Akroyd, Miguel Molina-Solana, Yike Guo, and William J. Knottenbelt. **Visualizing dynamic bitcoin transaction patterns**. *Big data* 4, no. 2 (2016): 109-119.

Obfuscation Efforts 2- Coin Mixing

- A measure to prevent matching addresses to users is known as Coin Mixing, or its improved version, CoinJoin.

- The initial idea in mixing was to use a central server to mix inputs from multiple users.

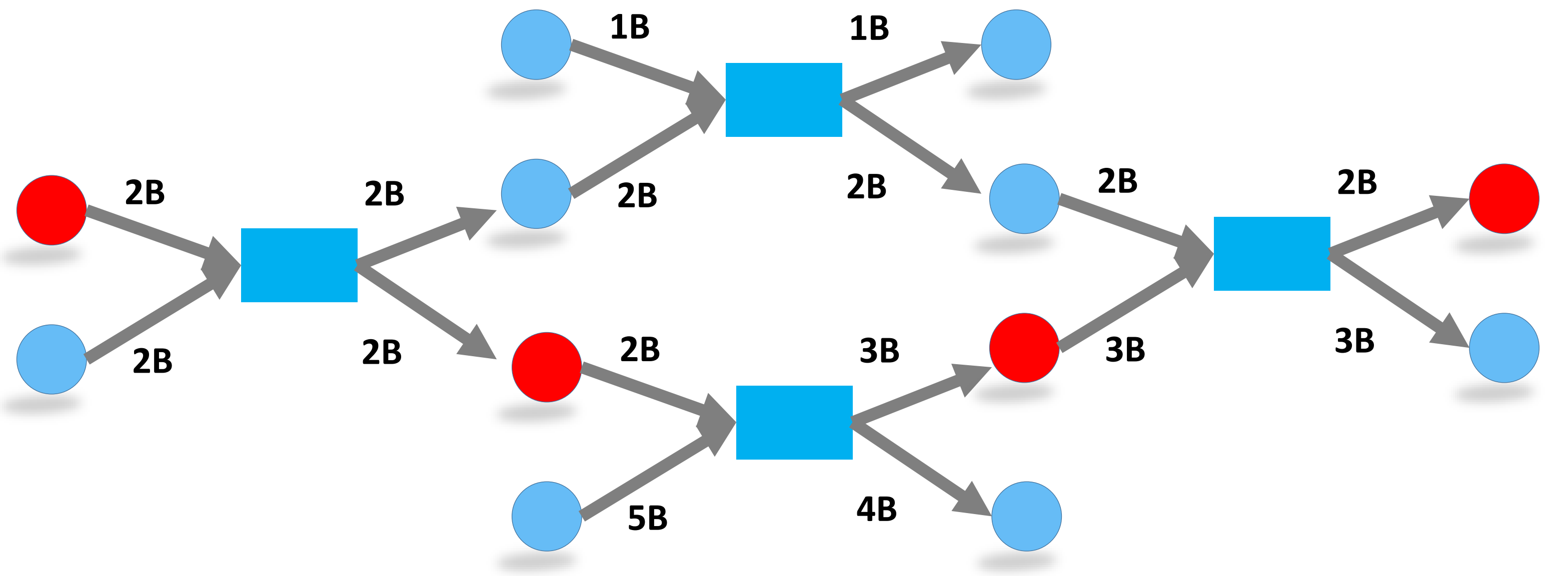


Ruffing, Tim, Pedro Moreno-Sanchez, and Aniket Kate. **CoinShuffle: Practical decentralized coin mixing for Bitcoin**. In *European Symposium on Research in Computer Security*, pp. 345-364. Springer, Cham, 2014.

Obfuscation Efforts 2- Coin Mixing

- A measure to prevent matching addresses to users is known as Coin Mixing, or its improved version, CoinJoin.

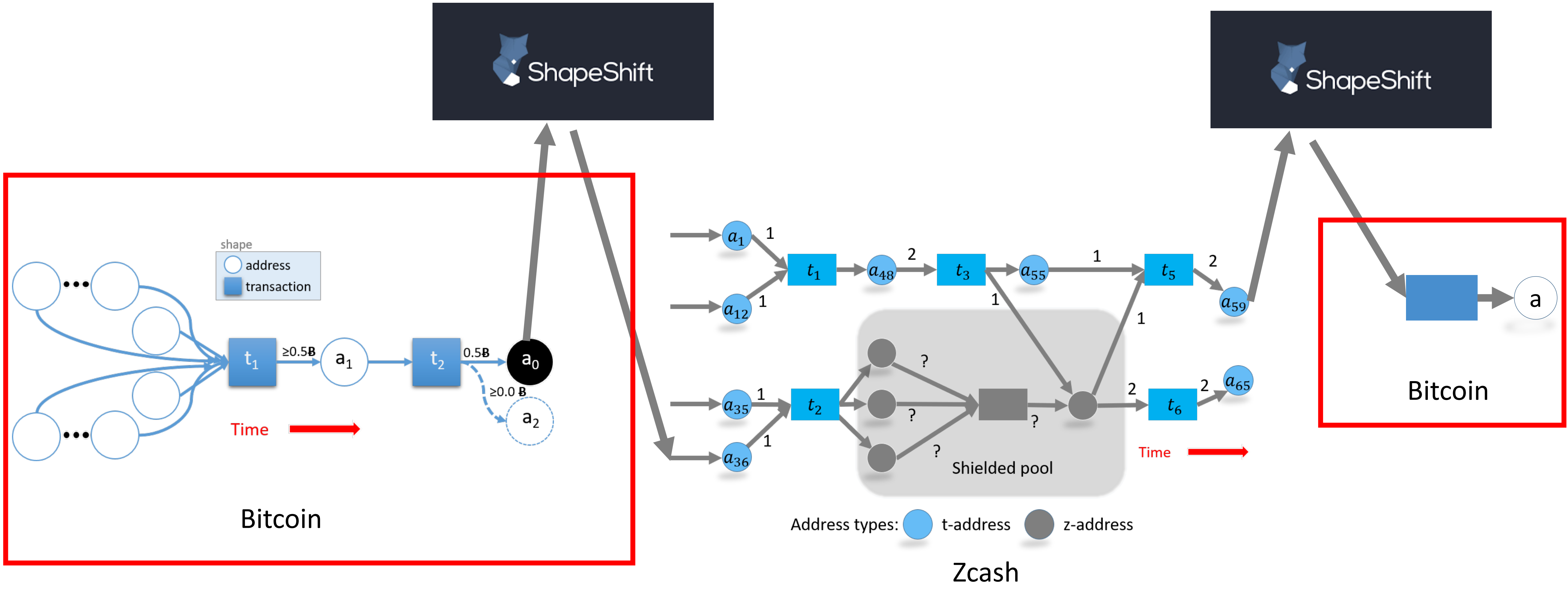
- The initial idea in mixing was to use a central server to mix inputs from multiple users.



Ruffing, Tim, Pedro Moreno-Sanchez, and Aniket Kate. **CoinShuffle: Practical decentralized coin mixing for Bitcoin**. In *European Symposium on Research in Computer Security*, pp. 345-364. Springer, Cham, 2014.

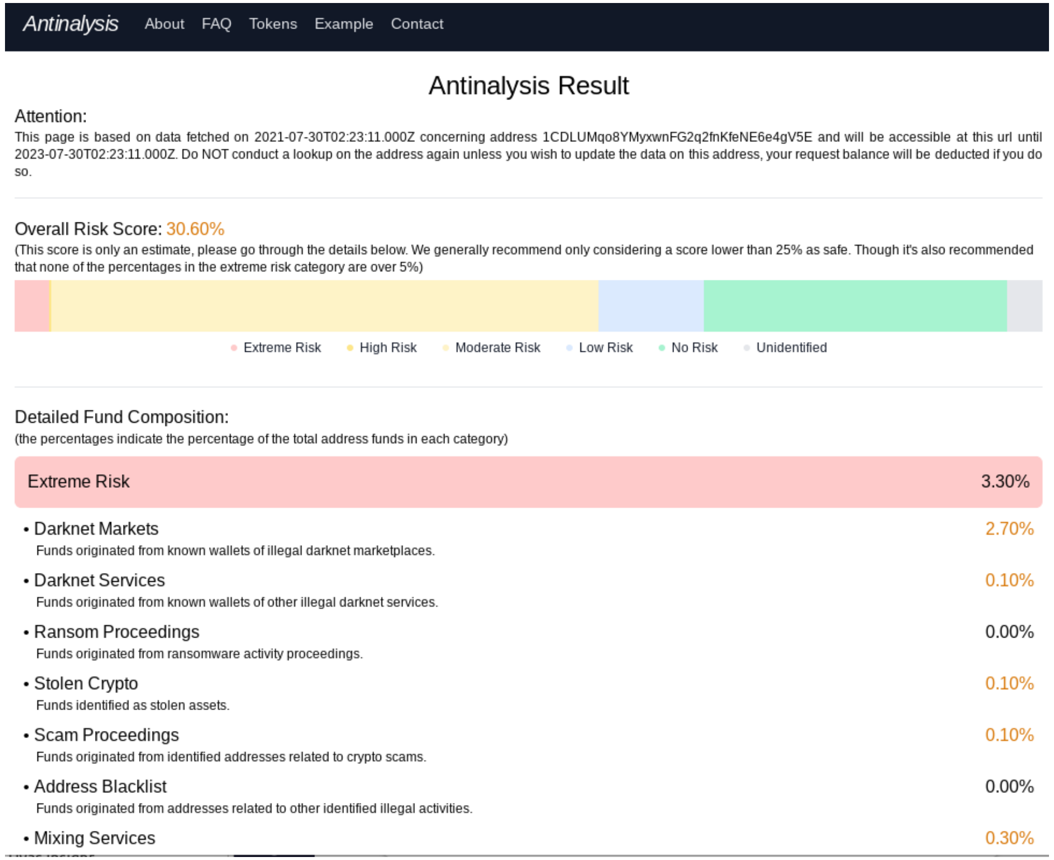
Obfuscation Efforts 3 - Shape Shifting

- Shapeshifting is moving exchanging bitcoins for Zcash/Monero, moving the coins within the privacy coin securely and bringing them back to bitcoin.



Counter-Counter Measures - Antinalysis

"Worried about dirty funds in your BTC address? Come check out Antinalysis, the new address risk analyzer" – a darknet market.



Antinalysis About FAQ Tokens Example Contact

Antinalysis Result

Attention:
This page is based on data fetched on 2021-07-30T02:23:11.000Z concerning address 1CDLUMqo8YMyxwnFG2q2fnKfeNE6e4gV5E and will be accessible at this url until 2023-07-30T02:23:11.000Z. Do NOT conduct a lookup on the address again unless you wish to update the data on this address, your request balance will be deducted if you do so.

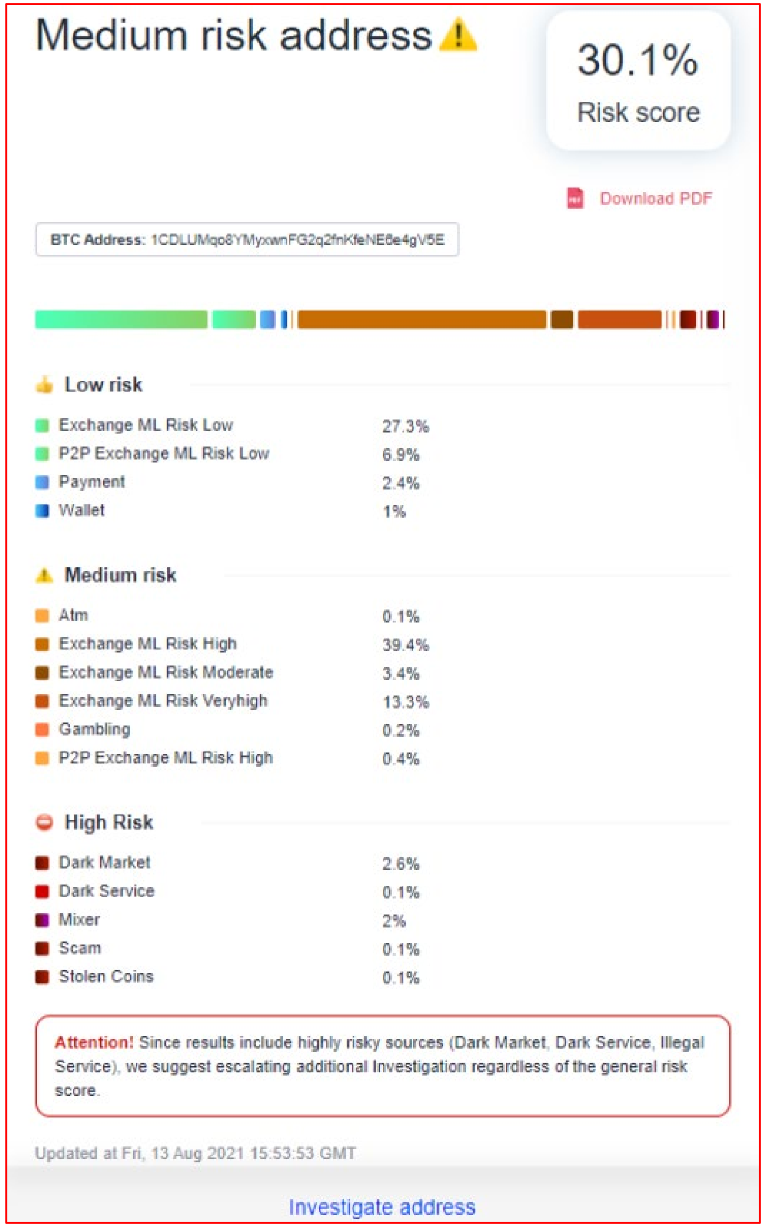
Overall Risk Score: 30.60%
(This score is only an estimate, please go through the details below. We generally recommend only considering a score lower than 25% as safe. Though it's also recommended that none of the percentages in the extreme risk category are over 5%)

Extreme Risk · High Risk · Moderate Risk · Low Risk · No Risk · Unidentified

Detailed Fund Composition:
(the percentages indicate the percentage of the total address funds in each category)

Extreme Risk	3.30%
• Darknet Markets Funds originated from known wallets of illegal darknet marketplaces.	2.70%
• Darknet Services Funds originated from known wallets of other illegal darknet services.	0.10%
• Ransom Proceedings Funds originated from ransomware activity proceedings.	0.00%
• Stolen Crypto Funds identified as stolen assets.	0.10%
• Scam Proceedings Funds originated from identified addresses related to crypto scams.	0.10%
• Address Blacklist Funds originated from addresses related to other identified illegal activities.	0.00%
• Mixing Services	0.30%

AMLBot



Medium risk address ⚠

30.1%
Risk score

Download PDF

BTC Address: 1CDLUMqo8YMyxwnFG2q2fnKfeNE6e4gV5E

Low risk

Exchange ML Risk Low	27.3%
P2P Exchange ML Risk Low	6.9%
Payment	2.4%
Wallet	1%

Medium risk

Atm	0.1%
Exchange ML Risk High	39.4%
Exchange ML Risk Moderate	3.4%
Exchange ML Risk Veryhigh	13.3%
Gambling	0.2%
P2P Exchange ML Risk High	0.4%

High Risk

Dark Market	2.6%
Dark Service	0.1%
Mixer	2%
Scam	0.1%
Stolen Coins	0.1%

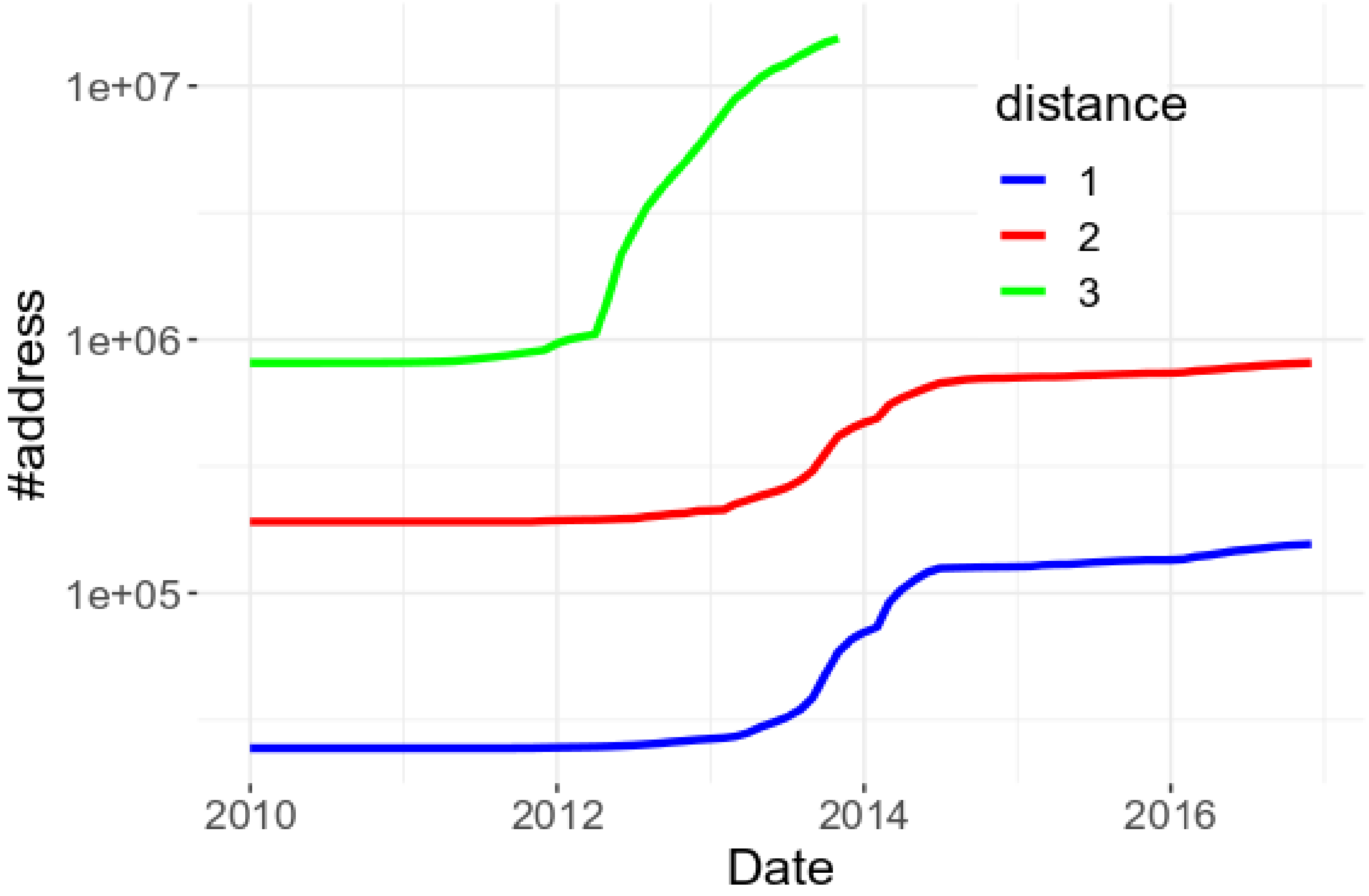
Attention! Since results include highly risky sources (Dark Market, Dark Service, Illegal Service), we suggest escalating additional Investigation regardless of the general risk score.

Updated at Fri, 13 Aug 2021 15:53:53 GMT

Investigate address

Locating Payments – Tx Fingerprinting

- What is difficult about transaction fingerprinting (matching sale amounts to transaction amounts)?

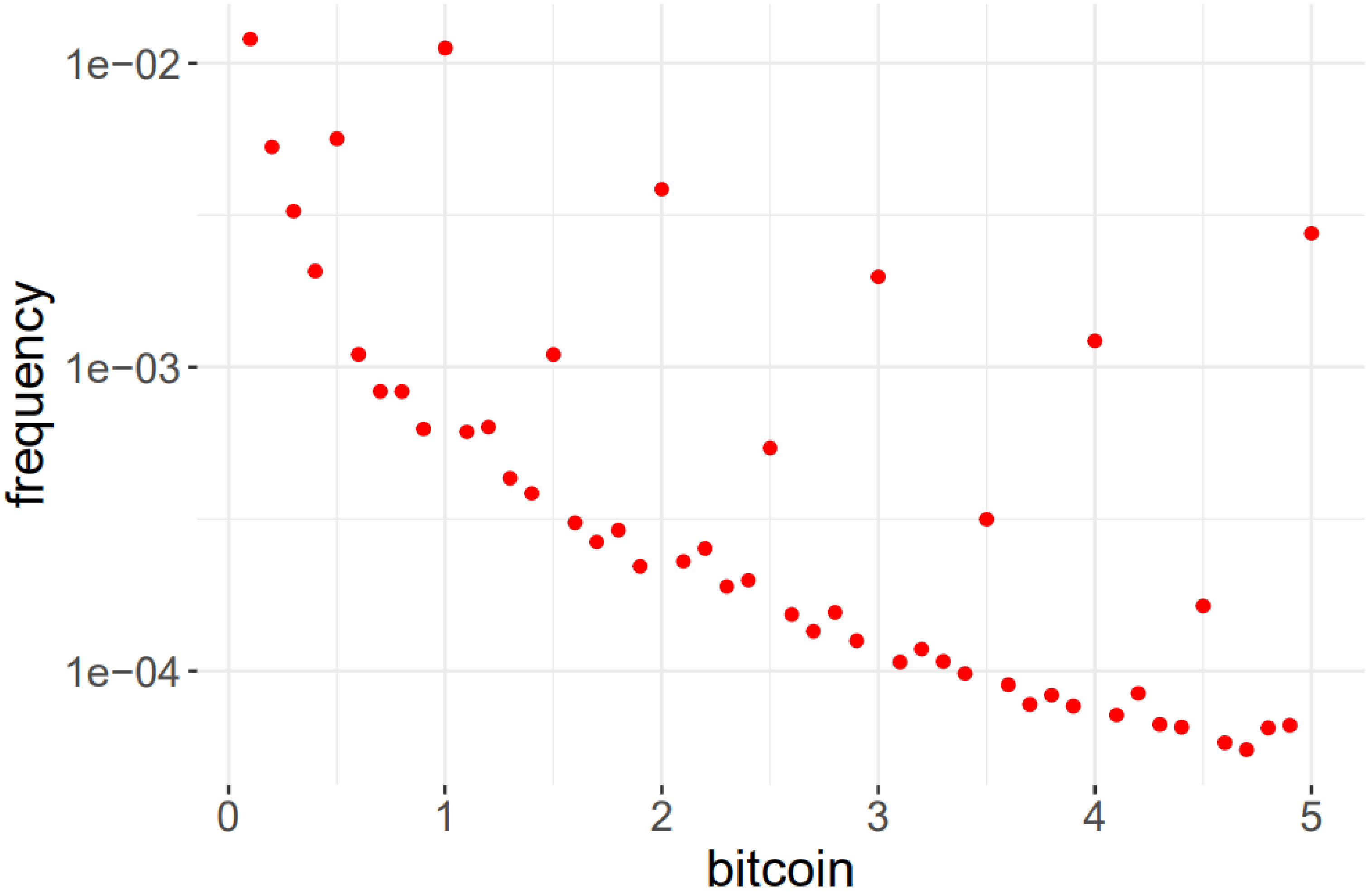


We used the Wannacry ransomware addresses in this analysis.

In 2-3 hops from certain addresses (e.g., ransomware addresses) of interest, too many bitcoin addresses are caught in the search net.

Amount Matching (Fingerprinting)

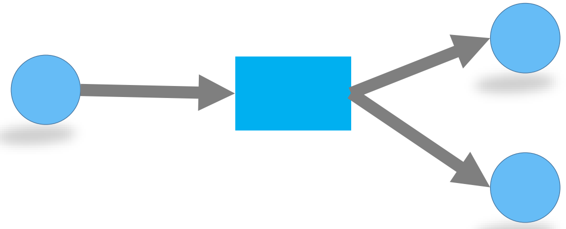
- What is difficult about transaction fingerprinting?



- Amounts can be chosen carefully to complicate transaction detection.
- Do not use too specific amounts like 0.1457 btc.

Figure: Amounts in all Bitcoin transactions.

Amount Matching (Fingerprinting)



- What is difficult about transaction fingerprinting?

57.04% of all transactions are one input, two output chainlets

Output address count

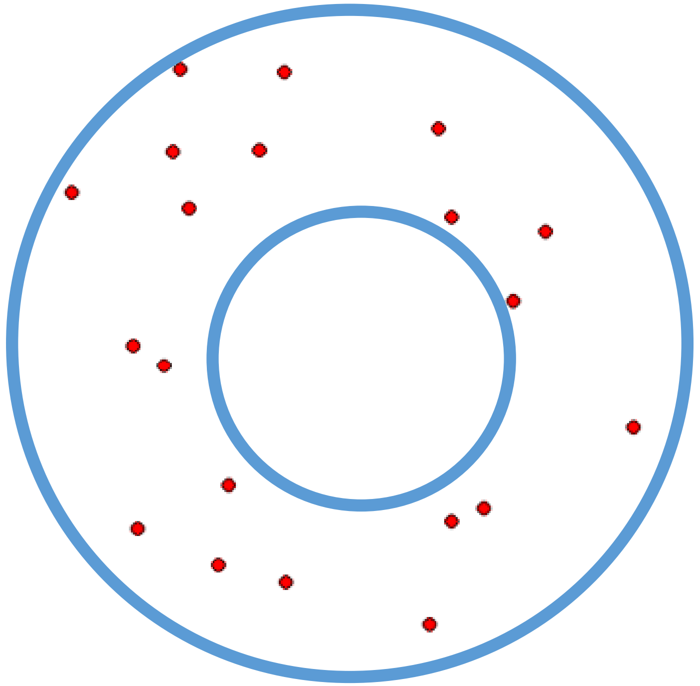
Input address count

8.45	57.04	1.56	0.5	0.26	1.05
1.31	12.83	1.79	0.13	0.09	0.25
0.76	3.81	0.38	0.05	0.04	0.11
0.3	2.2	0.1	0.05	0.02	0.07
0.22	1.06	0.05	0.02	0.02	0.05
0.72	2.36	0.11	0.05	0.04	0.45

Patterns can be chosen carefully – using transactions with one input and two outputs in every payment puts you in a large privacy pool.

Topological Data Analysis on Blockchain Graphs

Why TDA?



What is the true shape of this data?

Why TDA?

- Is there a set of tools which detects the shape of the object underlying a dataset?

- Persistent Homology of TDA is a way to watch how the homology of a filtration (sequence) of topological spaces changes so that we can understand something about the space.

TDA on Point Clouds

Let X be a discrete set in some metric space.

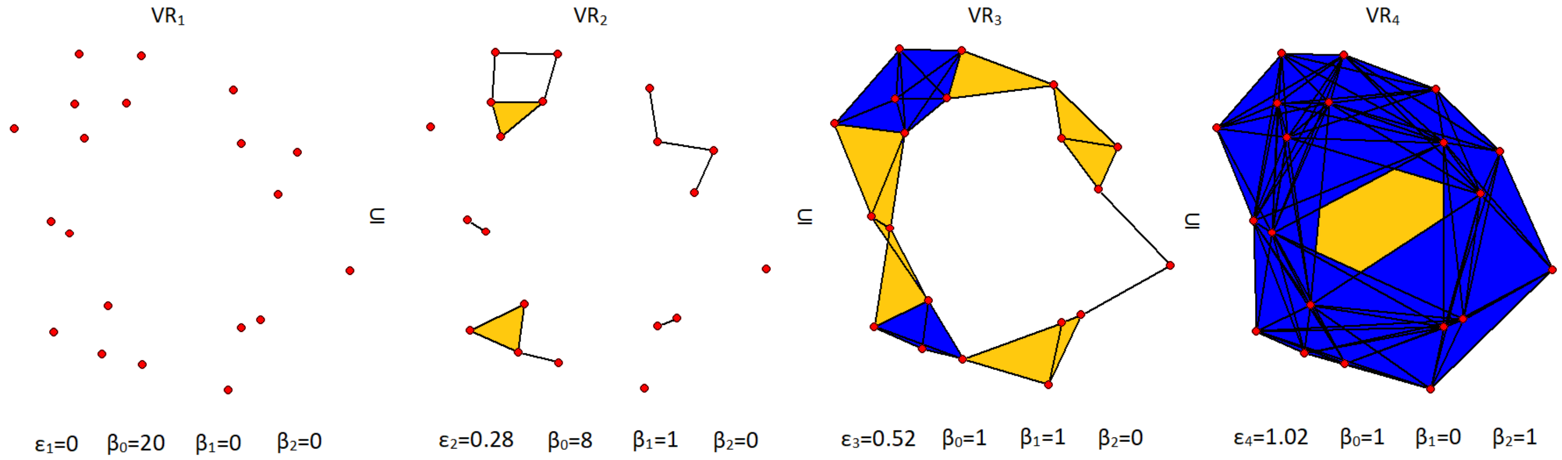
- Now, we fix an increasing sequence of scales $\epsilon_1 < \epsilon_2 < \cdots < \epsilon_n$ and construct a chain of nested Vietoris-Rips complexes called a finite VR filtration $VR_{\epsilon_1} \subseteq VR_{\epsilon_2} \subseteq \cdots \subseteq VR_{\epsilon_n}$, where VR_{ϵ_k}, $k = 1, \ldots, n$.

We expect that features with a longer lifespan, i.e. persistent features, have a higher role in explaining structure and functionality of the data than features with a shorter lifespan.

Topological Data Analysis – Persistent Homology

- To extract summaries of such topological features at a mesoscopic level, we use **Betti numbers**.
- Betti-p number of a simplicial complex $\mathcal{C}$ of dimension d, denoted by $\beta_p(\mathcal{C})$, is defined as

- $$\beta_p(\mathcal{C}) = \begin{cases} \text{\# of connected components of } \mathcal{C} & p = 0 \\ \text{\# of 1–D holes or tunnels of } \mathcal{C} & p = 1 \\ \text{\# of 2–D holes or cavities of } \mathcal{C} & p = 2 \\ \quad\cdots\cdots & \end{cases}$$



Betti numbers at increasing dissimilarity scales.

Topological Data Analysis of Blockchain – Ethereum Case

- Let $G = (V, E, \omega)$ be a weighted graph, with the node set V and edge set E and $\omega: E \to R^+$ is a function encoding dissimilarity between two nodes connected by an edge.

- To account for dissimilarity between two disconnected nodes, we introduce the weight $\widetilde{\omega}: V \times V \to R^+$

$$\widetilde{\omega}_{uv} = \begin{cases} \omega_{uv} & (u, v) \in E \\ \infty & (u, v) \notin E. \end{cases}$$

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

Topological Data Analysis of Blockchain – Ethereum Case

- In the context of a weighted network, we define ω_{uv} as

$$\omega_{uv} = [1 + \frac{(A_{uv} - A_{min}) \cdot (a - b)}{(A_{max} - A_{min})}]^{-1},$$

where A_{uv} is the weight of the edge (total amount of tokens traded) between nodes u and v. Values of a and b create a scale.

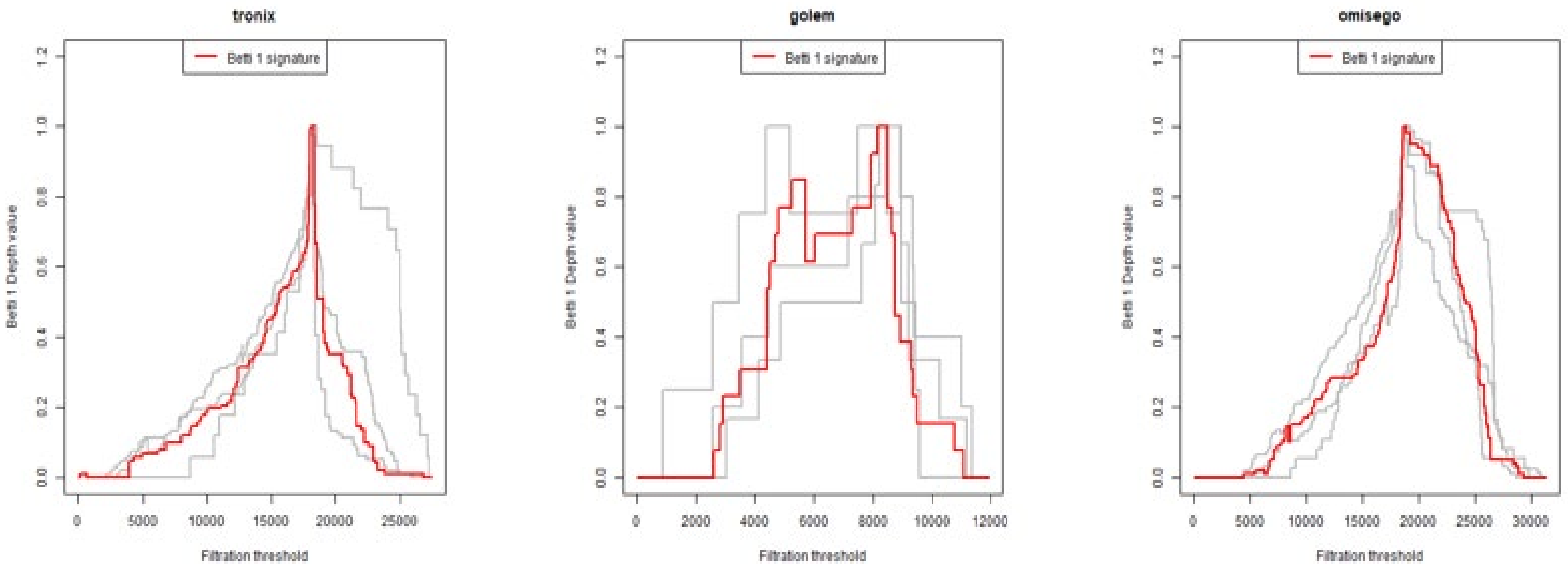
- A_{min} and A_{max} are the smallest and the largest edge weights, respectively.

Topological Data Analysis of Ethereum Networks

- In this context, we introduce a novel notion of Betti functions which relate these counts to the scale parameter viewed as continuum.
- The **Betti-$\boldsymbol{p}$**function $\mathcal{B}_p: R^+ \to \{0,1,2,3,\dots\}$, $p = 0, \dots, d$, associated with $\{\mathcal{C}_\epsilon\}_{\epsilon \in R^+}$ is defined as

$$\mathcal{B}_p: \epsilon \mapsto \beta_p(\mathcal{C}_\epsilon).$$

- Sequence of Betti numbers are finite dimensional realizations of Betti functions.

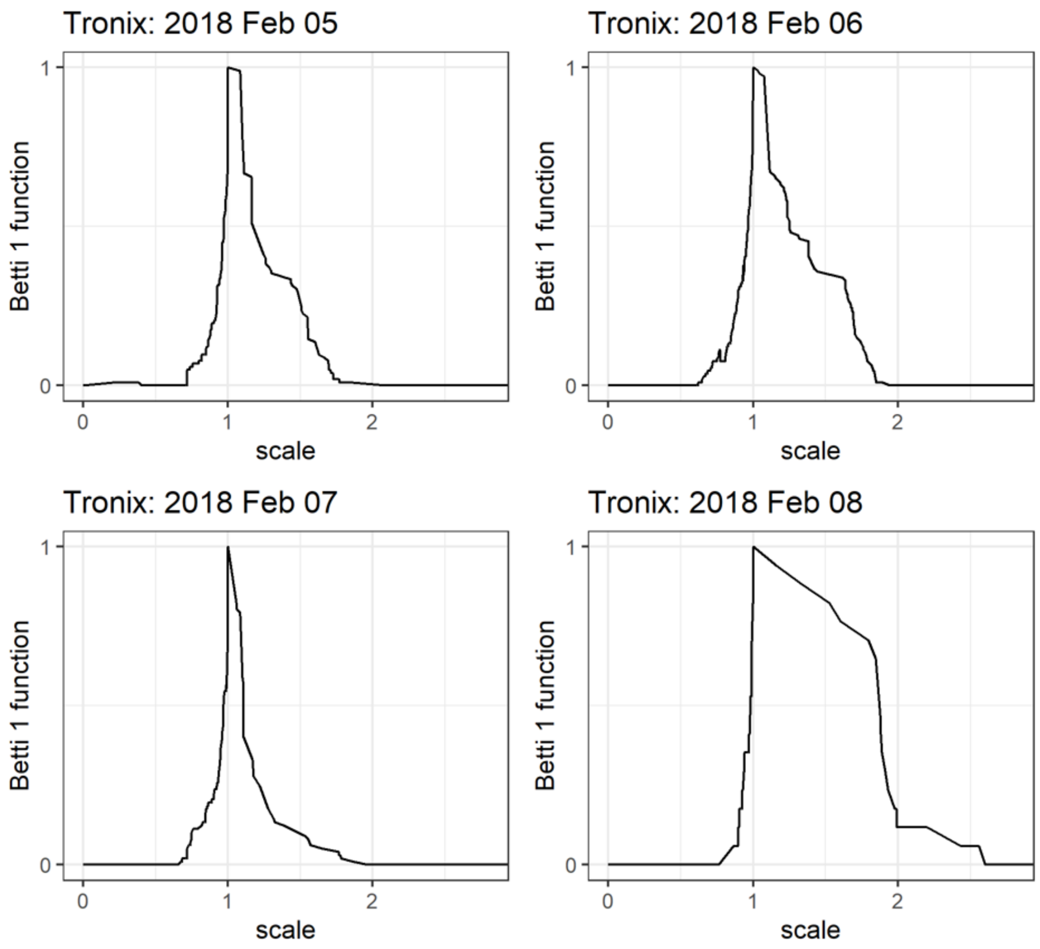


Topological Data Analysis of Ethereum Networks

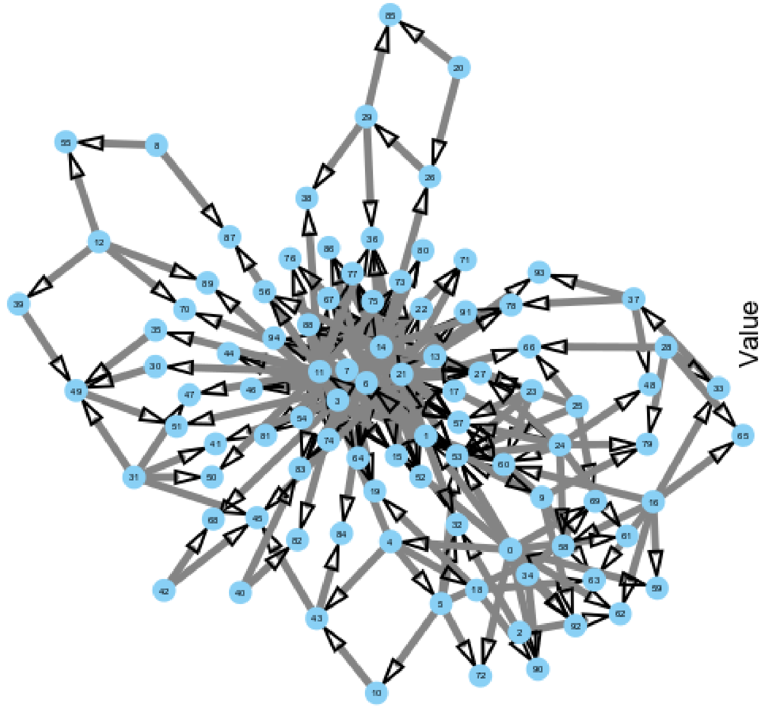
- The *Betti functions* can be regarded as a functional summary statistic of the network's topological structure.

- Due to the functional dependency among Betti numbers at different scales, it is important to view $\{\mathcal{B}_p(\epsilon_k)\}_{k=1}^{n}$ as a function as opposed to a vector in R^n.

- This point of view allows us to utilize methods from functional data analysis such as a concept of functional data depth.



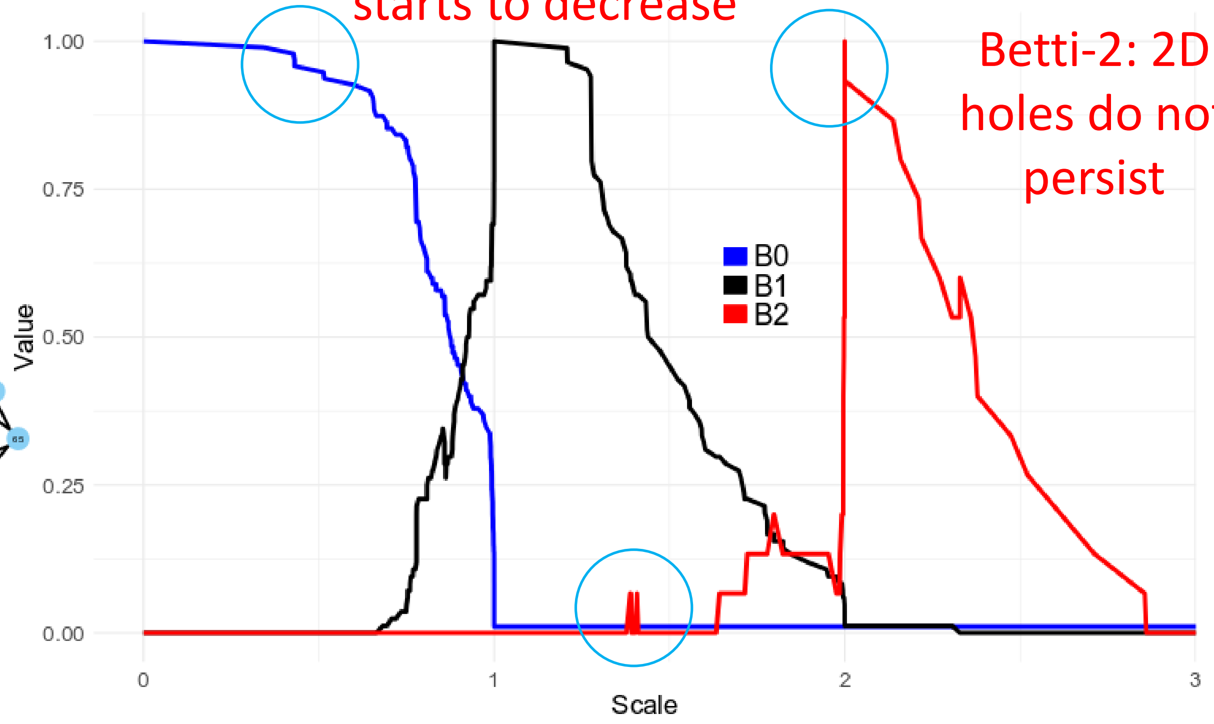
Topological Data Analysis of Ethereum Networks



The biggest connected component of the Storj network (for a single day).

Scale at which Betti-0 starts to decrease

Betti-2: 2D holes do not persist



Betti-2: early 2D holes disappear

Corresponding Betti functions.

Topological Data Analysis of Ethereum Networks

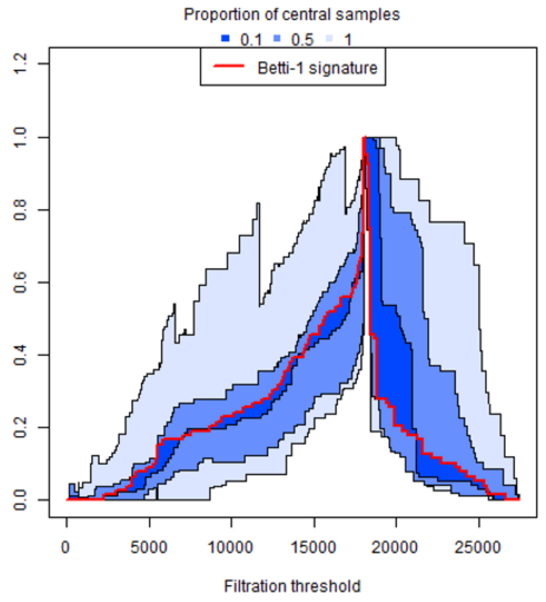
- Consider Betti functions $\{\mathcal{B}_{p,t}\}_{t=1}^{T}$ associated with an evolving token transaction network over days $t = 1, 2, \ldots, T$.

- Although each day visually looks different, some days present a clear anomaly in terms of their shape.

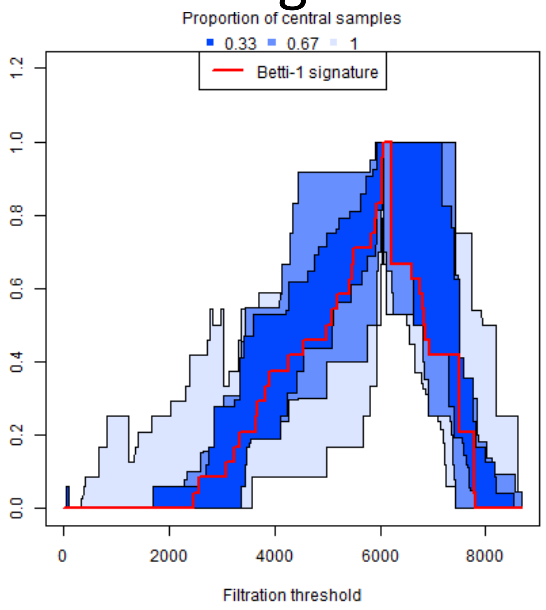
- We use a notion of rolling band depth: $RD_w(\mathcal{B}_{p,t})$:
$= MBD(\mathcal{B}_{p,t} | \mathcal{B}_{p,t}, \mathcal{B}_{p,t-1}, \ldots \mathcal{B}_{p,t-w+1})$.

- We introduce a concept of *Betti signature* which is defined as the deepest or most central Betti function.

Tronix token

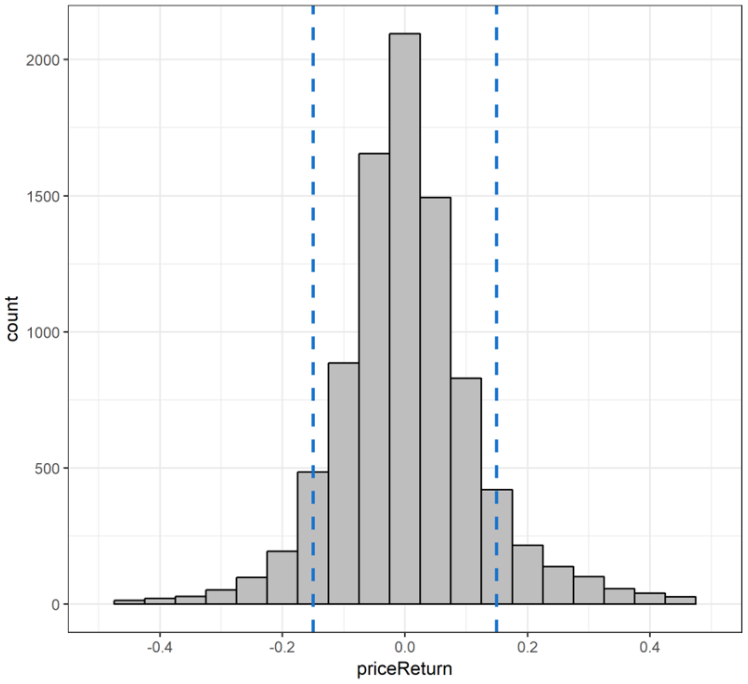


PowerLedger token

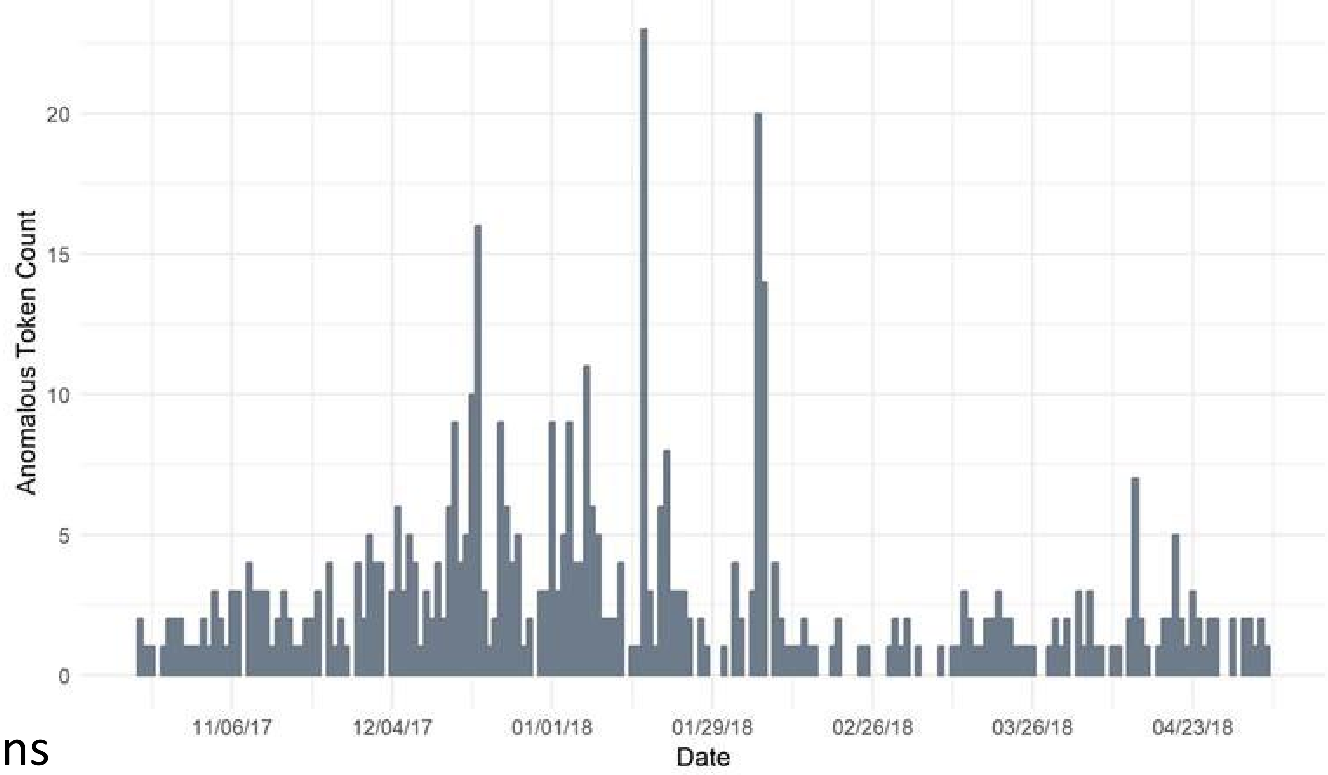


Next: Predictive Models

Problem Definition: Given the transaction network of an Ethereum token and time series of the token price in fiat currency, predict whether the token price will change more than δ in the next h days. Identify the maximum horizon value h such that the prediction accuracy is at least ρ.



A histogram of absolute price returns of 31 tokens $R_t = (Price_t - Price_{t-1})/(Price_{t-1})$.



Number of (price) anomalous tokens in time.

TDA in UTXO Networks

A brief background on a use case:

Ransomware is a type of malware that infects a victim's data and resources and demands ransom to release them.

Images: gdatasoftware.com, healthcareitnews.com

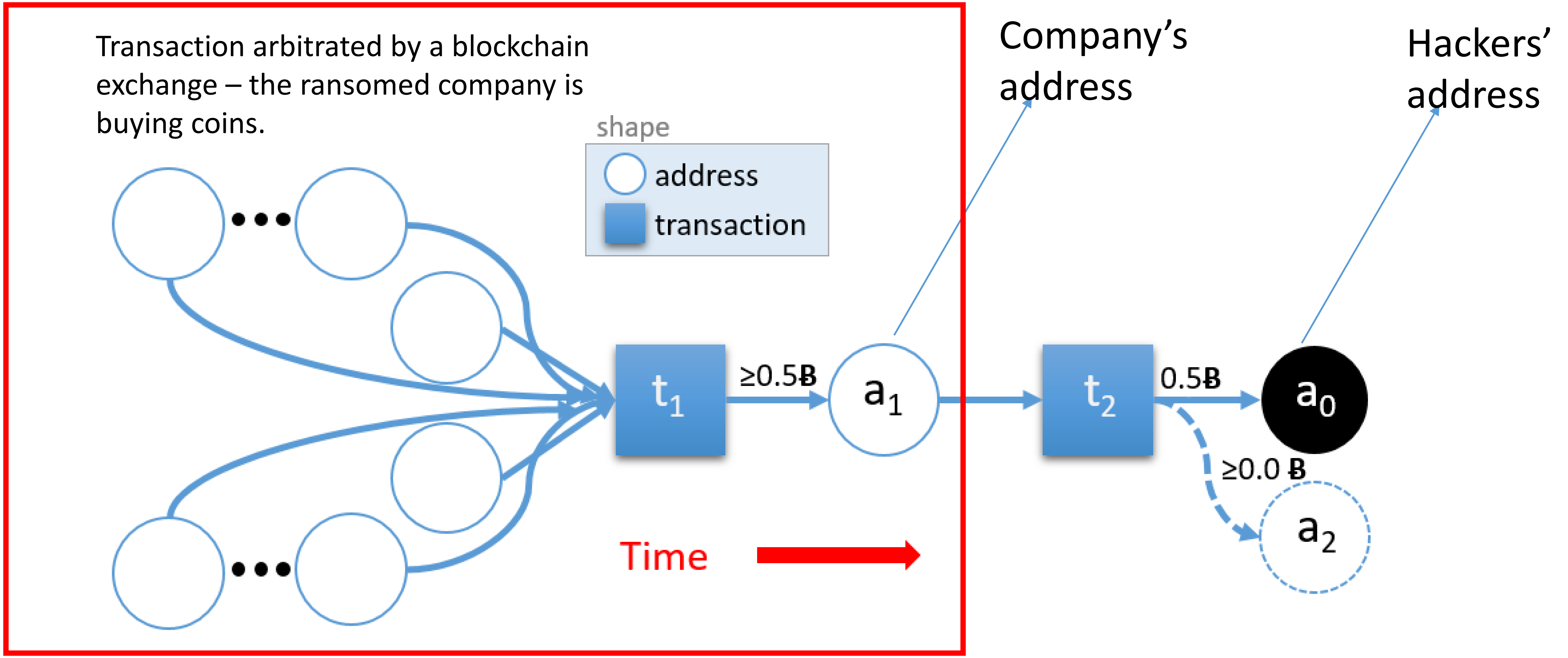
Why Now?

The combination of strong and well-implemented cryptographic techniques to take files hostage, the Tor protocol to communicate anonymously, and the use of a cryptocurrency to receive unmediated payments provide altogether a high level of impunity for ransomware attackers.

Paquet-Clouston, "Ransomware payments in the Bitcoin ecosystem (2019)"

https://arxiv.org/abs/1804.04080

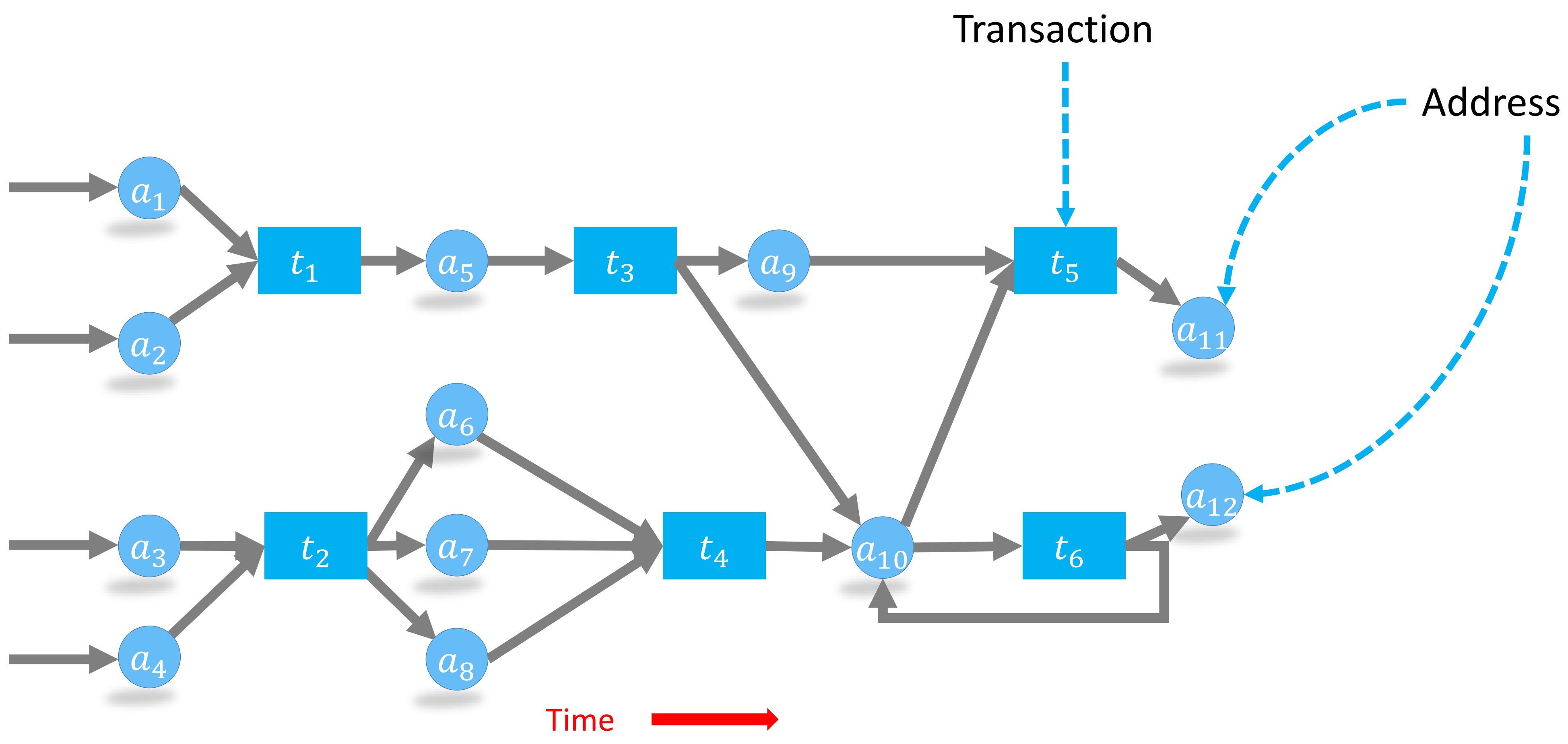
The Anatomy of a Ransom Payment



- There is a considerable time (e.g., 20 hours) gap between t_1 and t_2.
- Searching this exact pattern catches many true positives.

Public Network

Bitcoin transaction network is public – we can see all coin transfers.



Our Tasks

Can we identify ransomware victims automatically?

Our two tasks!

Can we discover new ransomware families?

On Bitcoin

Our Data

Our ransomware dataset is a union of datasets from three widely adopted studies:

Montreal, Princeton and Padua.

The combined dataset contains 24,486 addresses from 27 ransomware families.

Huang, D.Y., Aliapoulios, M.M., Li, V.G., Invernizzi, L., Bursztein, E., McRoberts, K., Levin, J., Levchenko, K., Snoeren, A.C. and McCoy, D., 2018, May. **Tracking ransomware end-to-end.** In *2018 IEEE Symposium on Security and Privacy (SP)* (pp. 618-631). IEEE.

Paquet-Clouston, M., Haslhofer, B. and Dupont, B., 2019. **Ransomware payments in the bitcoin ecosystem.** *Journal of Cybersecurity*, *5*(1).

Conti, M., Gangwal, A. and Ruj, S., 2018. **On the economic significance of ransomware campaigns: A Bitcoin transactions perspective.** *Computers & Security*, *79*, pp.162-189.

Network Snapshots

We divide the Bitcoin network into 24-hour long windows by using the UTC-6 timezone as reference.

On the Bitcoin network, an address may appear multiple times.

An address u that appears in a transaction at time t can be denoted as a_u^t.

Notation

Let $\{a_u\}_{u \in Z^+}$ be a set of addresses and let each address a_u be associated with a pair $(\vec{x}_u, y_u)$, where $\vec{x}_u \in \mathcal{R}^D$ is a vector of its features and y_u is its label.

The label y_u can designate a white (i.e., non-ransomware) address or a ransomware address.

White vs. Dark Addresses

Let $f_1, \ldots, f_n$ be labels of known ransomware families which have been observed until time point t.

We set f_0 to be the label of addresses which are not known to belong to any ransomware family, and we assume them to be white addresses.

Assumption: those addresses that we do not know as ransomware are white (non-ransom) addresses.

Why the Window?

The window approach serves two purposes:

- The induced 24-hour network allows us to capture how fast a coin moves in the network.
- Temporal information of transactions, such as the local time, has been found useful to cluster criminal transactions.

Features

On the heterogeneous Bitcoin network, in each snapshot we extract the following six features for an address:

Income of an address u is the total amount of coins output to u: $I_u = \sum_{t_n \in \Gamma_u^o} A_u^o(n)$.

Neighbors of an address u is the number of transactions which have u as one of its output addresses: $\left|\Gamma_u^i\right|$.

Income and neighbors do not consider position of the address in the network!

Features

We designed graph features to quantify specific obfuscation patterns used by ransomware operators:

- Loop counts how many transactions i) split their coins; ii) move these coins in the network by using different paths and finally, and iii) merge them in a single address.

- Weight quantifies the merge behavior, where coins in multiple addresses are each passed through a succession of merging transactions and accumulated in a final address.

Features

Count represents information on the number of transactions, whereas the weight feature represents information on the amount (what percent of starter transactions’ output?).

Length quantifies mixing rounds on Bitcoin, where transactions receive and distribute similar amounts of coins in multiple rounds with newly created addresses to hide the coin origin.

Table 1: Most frequent feature values in ransomware addresses.

Len	Wei	Nei	Cou	Loo	Inc	# addresses	OverallRank
0	0.5	2	1	0	1	327	1
0	0.5	2	1	0	1.2	250	113
0	1	2	1	0	1	189	4
0	1	1	1	0	0.5	178	9
0	0.5	2	1	0	0.8	160	116
0	1	1	1	0	1	146	3
0	1	2	1	0	1.2	127	121
0	0.5	2	1	0	1.25	119	327
0	0.5	1	1	0	0.5	118	6
0	1	1	1	0	2	117	18

Most Payments are N-1 or N-2!

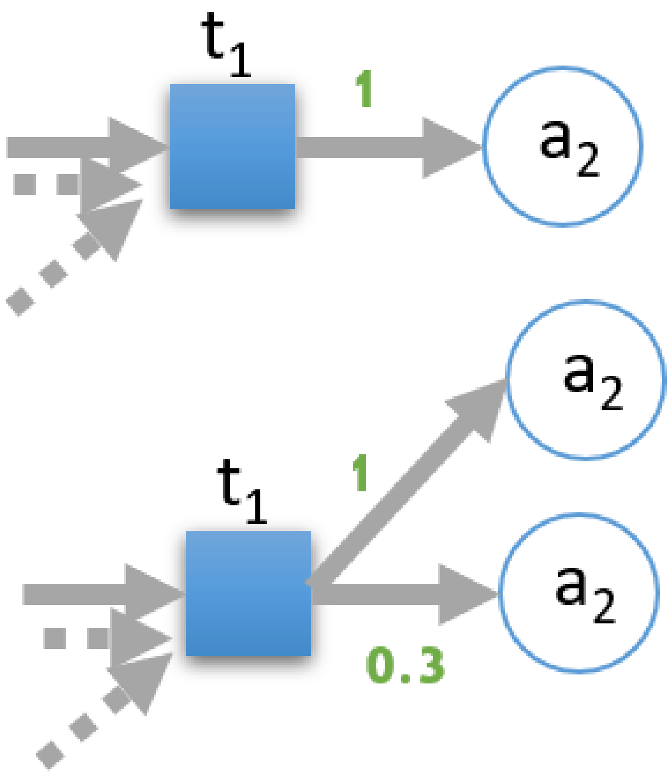
Length 0: The first transaction involving these coins in the day.

Weight 1: All output goes into the address.

Neighbor 1: One transaction makes a payment into the address.

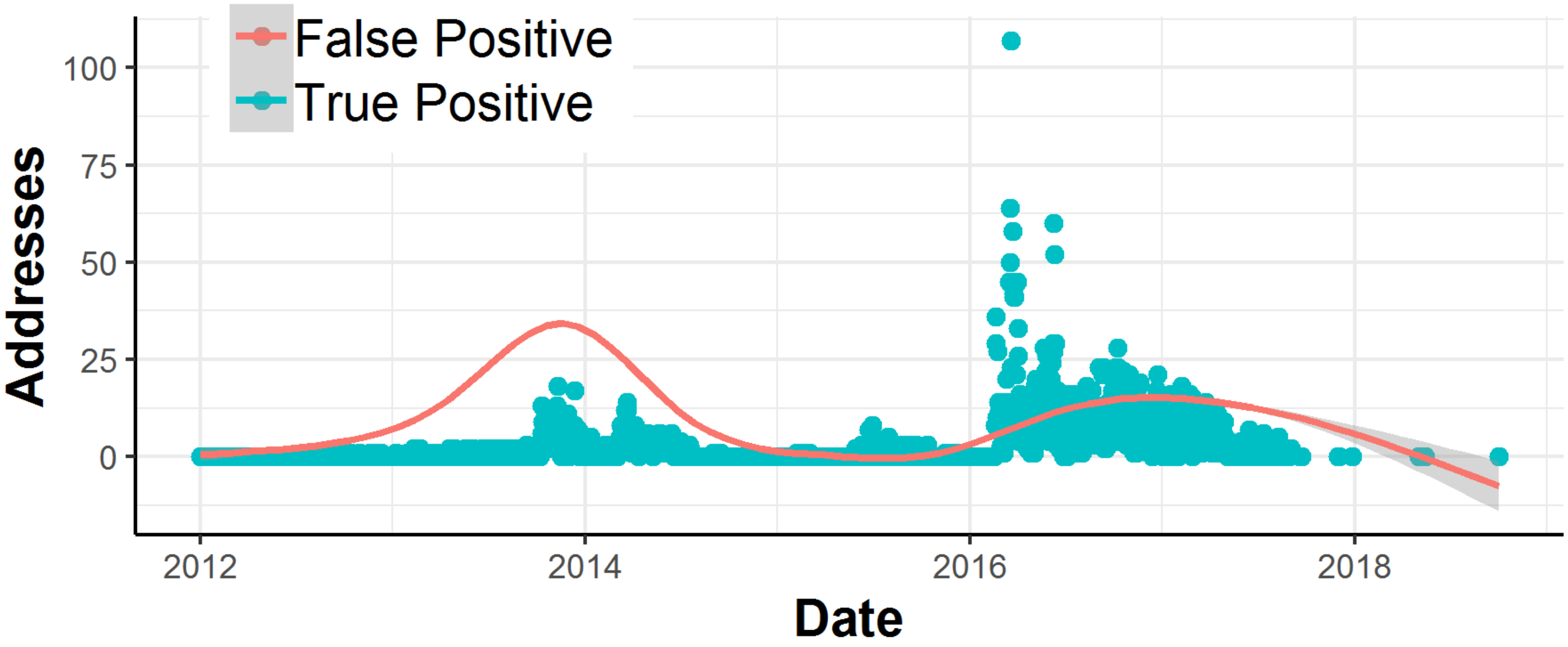
Count 1: One starter transaction reaches the address.

Loop 0: No obfuscation, coins are directly paid.



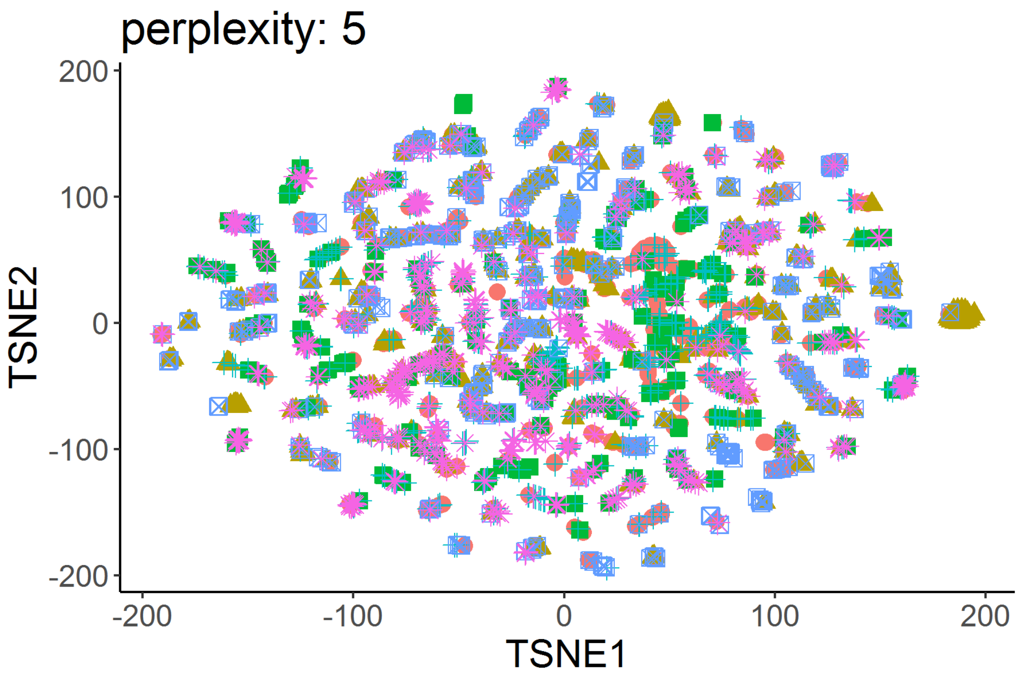
Experiment 1: Detecting Undisclosed Payments

Naïve approach: Similarity search all history. Not so bad!



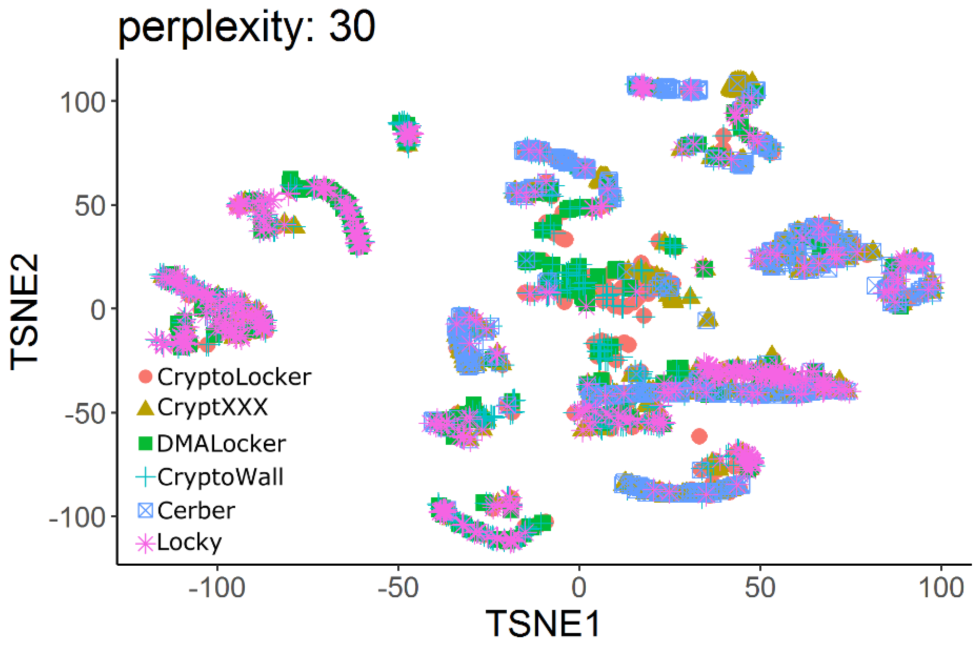
However, this naive approach creates 21,371 FP addresses overall.

Patterns



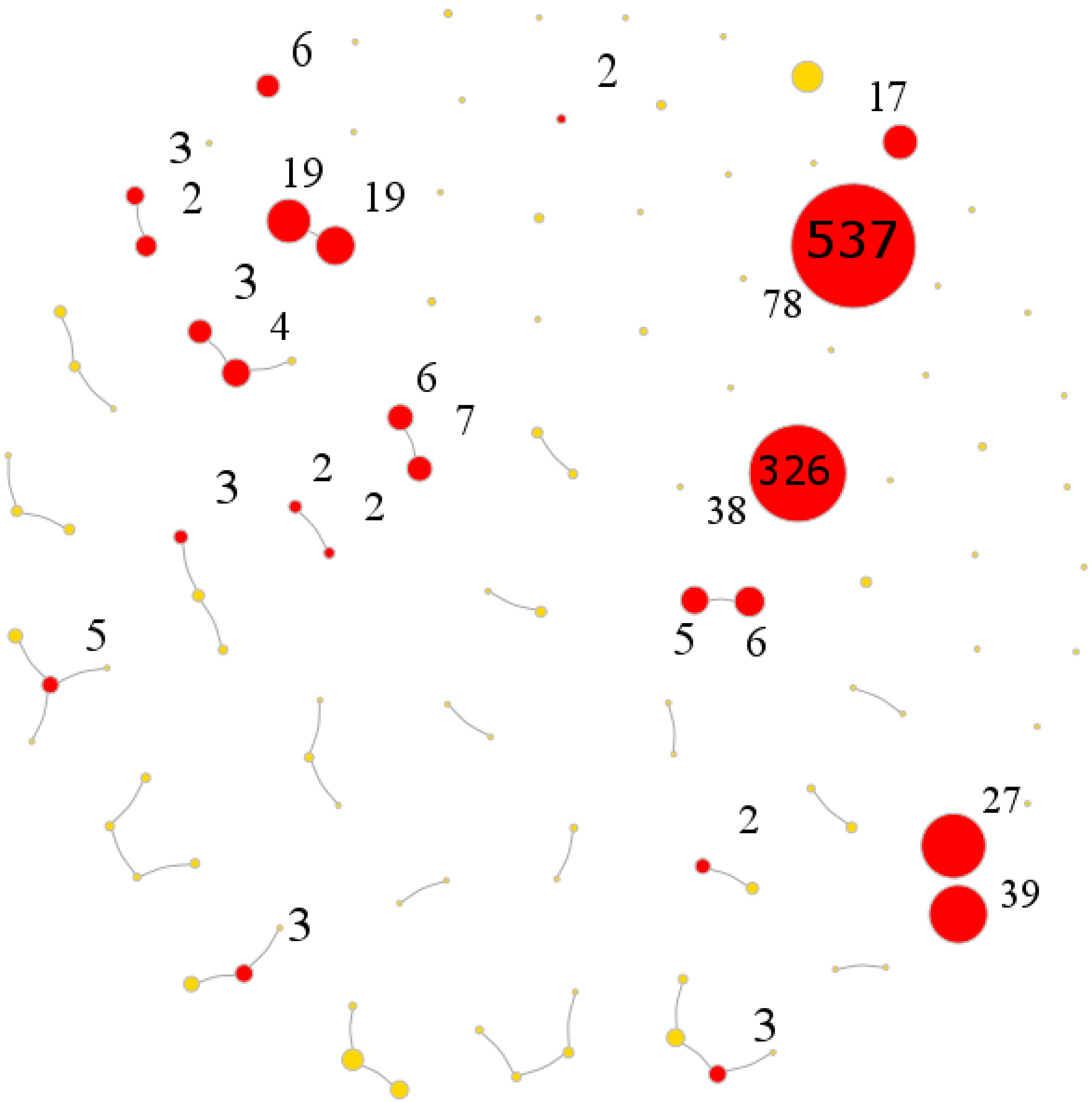
T-Stochastic neighbor embeddings of ransomware addresses

Address patterns are diverse!



Topological Analysis

We apply Topological Data Analysis for ransomware payment detection and compare our node classification results to ML techniques.



Problem

Network node classification with past labeled data.

1. Naïve Cosine similarity search
2. Transition and co-spending heuristics
3. Tree based methods: XGBoost, Random Forest
4. Clustering: DBSCAN, K-means

5. TDA Mapper

TDA Mapper

The key idea behind Mapper is the following:

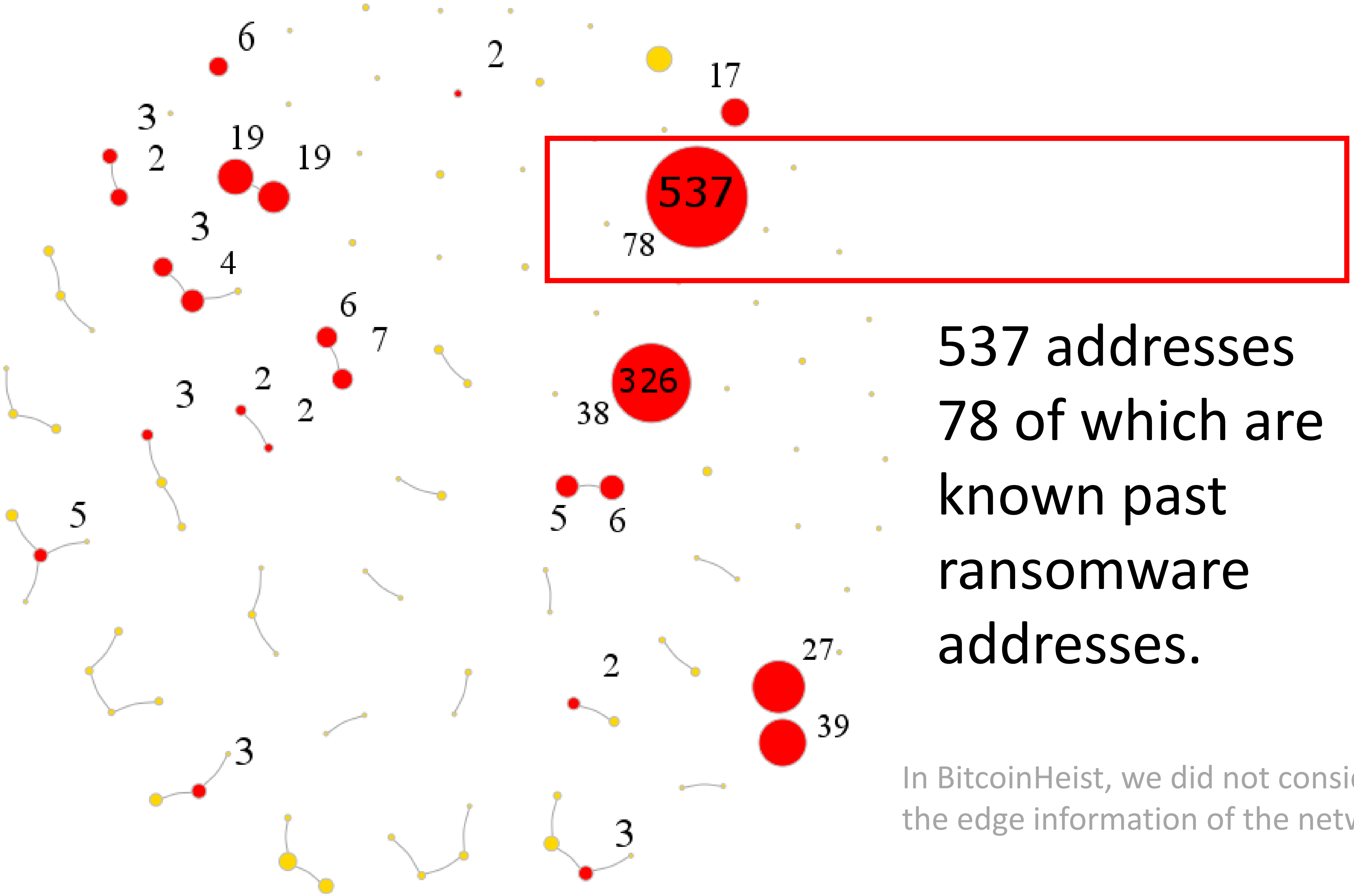
- Let U be a total number of observed addresses and $\{\vec{x}_u\}_{u=1}^{U} \in \mathcal{R}^D$ be a data cloud of address features.
- Select a filter function $\xi: \{\vec{x}_u\}_{u=1}^{U} \to \mathbb{R}$.
- Let I be the range of ξ, that is, $I = [m, M] \in \mathbb{R}$, where $m = \min_u \xi(\vec{x}_u)$ and $M = \max_u \xi(\vec{x}_u)$.

Features

- Now place data into overlapping bins by dividing the range I into a set S of smaller overlapping intervals of uniform length.

- Let $u_j = \{u: \xi(\vec{x}_u) \in I_j\}$ be addresses corresponding to features in the interval $I_j \in S$.

- For each u_j perform a single linkage clustering to form clusters $\{u_{jk}\}$.



537 addresses 78 of which are known past ransomware addresses.

In BitcoinHeist, we did not consider the edge information of the network.

TDA Mapper

- If current addresses are contained in clusters that also contain many past known ransomware addresses, by association, we deem these current addresses potential ransomware addresses.
- We filter the TDA mapper graph by using each of our six graph features. As a result, we get six filtered graphs $\mathcal{CT}_1, \ldots, \mathcal{CT}_6$ for each time window.
- Afterwards, we assign a suspicion, or risk score to an address a_u.

Experiment 1: Detecting Undisclosed Payments

- ML Methods: TDA gives the best F1. For each ransomware family, we predict 16.59 false positives for each true positive.

- In turn, this number is 27.44 for the best non-TDA models.

RS	Method	l	N	TP	FP	FN	TN	#w	Prec	Rec	F1	PLR
Locky	$\mathbb{TDA}^{.8\|.5}_{.9}$	240	300	451	2350	50	8221	11	0.161	0.900	0.273	0.192
	$\mathbb{COSINE}$	90	300	2395	41681	3990	146369	194	0.054	0.375	0.095	0.057
Crypto	$\mathbb{TDA}^{.8\|.65}_{.9}$	240	600	217	3087	155	11200	15	0.066	0.583	0.118	0.070
Wall	$\mathbb{DBSCAN}_{.2}$	240	600	728	18960	794	16913	59	0.037	0.478	0.069	0.038
Crypto	$\mathbb{TDA}^{.65\|.65}_{.9}$	240	300	439	9686	212	22129	34	0.043	0.674	0.081	0.045
Locker	$\mathbb{DBSCAN}_{.15}$	60	300	935	42771	295	11316	67	0.021	0.760	0.042	0.022
Cerber	$\mathbb{TDA}^{.5\|.35}_{.9}$	120	300	187	5174	459	23027	29	0.035	0.289	0.062	0.036
	$\mathbb{XGBOOST}$	240	300	1606	47307	7279	374169	436	0.033	0.181	0.056	0.034
Crypt	$\mathbb{TDA}^{.35\|.35}_{.9}$	90	300	77	2460	271	11057	14	0.030	0.221	0.053	0.031
XXX	$\mathbb{COSINE}$	30	600	589	20872	610	42952	65	0.027	0.491	0.052	0.028

Experiment 2: Predicting a New Family

RS	Method	Prec	Rec	TN	FP	TP	FN	PLR
CryptXXX	$\mathbb{TDA}^{0.2\|0.2}_{0.9}$	0.500	0.026	917	1	1	37	1.0
	$\mathbb{COSINE}$	0.046	0.342	654	264	13	25	0.049
Locky	$\mathbb{COSINE}$	0.098	0.138	795	37	4	25	0.108
	$\mathbb{TDA}^{0.05\|0.95}_{0.9}$	0.047	0.586	489	343	17	12	0.049
CryptoWall	$\mathbb{TDA}^{0.05\|0.95}_{0.9}$	0.0625	0.500	810	165	11	11	0.067
	$\mathbb{TDA}^{0.35\|0.8}_{0.9}$	0.061	0.500	805	170	11	11	0.0647
Cerber	$\mathbb{TDA}^{0.05\|0.95}_{0.9}$	0.029	0.214	849	100	3	11	0.030
	$\mathbb{TDA}^{0.35\|0.8}_{0.9}$	0.023	0.642	570	379	9	5	0.023
DMALocker	$\mathbb{DBSCAN}_{0.2}$	0.019	0.875	120	367	7	1	0.019
	$\mathbb{DBSCAN}_{0.15}$	0.015	0.875	4	459	7	1	0.015

In CryptXX we catch two addresses, one is a TP!

In general, we predict 27.53 false positives for each true positive

Through some ~~black magic~~ Topological Data Analysis methods

In locating ransomware addresses

We predict 16.59 false positive ransom addresses for each true positive.

In identifying new ransomware families.

We predict 27.53 false positive ransom addresses for each true positive.

Among 600K Bitcoin addresses daily!

Data and Article

UCI

Machine Learning Repository

Center for Machine Learning and Intelligent Systems

BitcoinHeistRansomwareAddressDataset

Download: Data Folder, Data Set Description

Abstract: BitcoinHeist datasets contains address features on the heterogeneous Bitcoin network to identify ransomware payments.

Data Set Characteristics:	Multivariate, Time-Series	**Number of Instances:**	2916697	**Area:**	Computer
Attribute Characteristics:	Integer, Real	**Number of Attributes:**	10	**Date Donated**	2020-06-17

BitcoinHeist: Topological data analysis for Ransomware prediction on the bitcoin blockchain
Cuneyt G. Akcora, Yitao Li, Yulia R. Gel, Murat Kantarcioglu.

Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, 2020.
https://www.ijcai.org/proceedings/2020/612

Machine Learning on Blockchain Graphs

Machine Learning on Blockchain Graphs

D. Lin, J. Wu, Q. Yuan, and Z. Zheng. **Modeling and understanding Ethereum transaction records via a complex network approach**. IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II: EXPRESS BRIEFS, VOL. 67, NO. 11, NOVEMBER 2020.

D. Lin, J. Wu, Q. Yuan, and Z. Zheng. **T-EDGE: Temporal WEighted MultiDiGraph Embedding for Ethereum transaction network analysis**. Front. Phys., 2020, Sec. Social Physics.

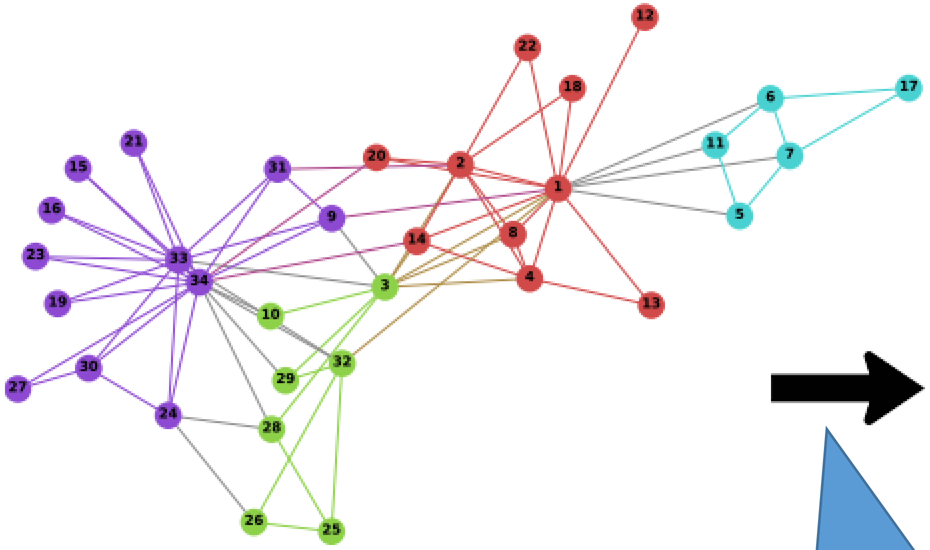
F. Poursafaei, R. Rabbany, and Z. Zilic. **SIGTRAN: Signature vectors for detecting illicit activities in Blockchain transaction networks**. PAKDD 2021.

J. Wu , Q. Yuan, D. Lin , W. You, W. Chen, C. Chen. **Who are the phishers? Phishing scam detection on Ethereum via network embedding**. IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS 2020.

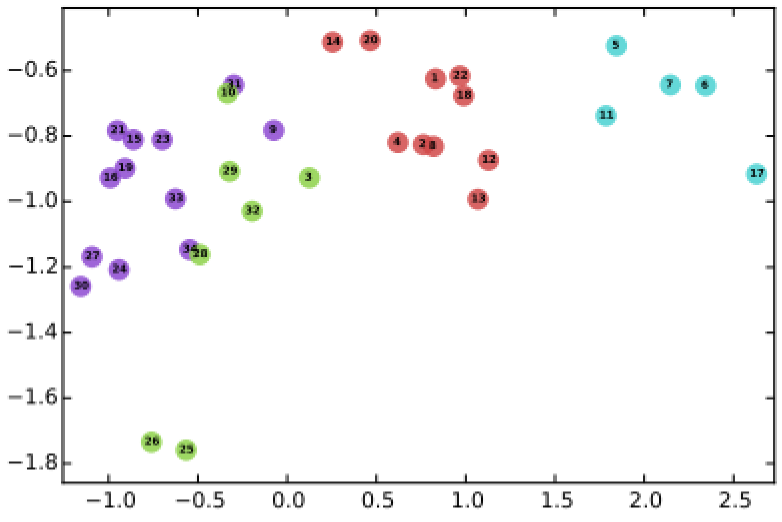
L. CHEN, J. PENG, Y. LIU, J. LI, F. XIE, and Z. ZHENG. **Phishing scams detection in Ethereum transaction network**. ACM Trans. Internet Technol. 2021.

T. Yu , X. Chen, Z. Xu, and J. Xu. **MP-GCN: a phishing nodes detection approach via graph convolution network for Ethereum**. Appl. Sci. 2022.

Graphs Representation Learning



Graph



Node embedding/ vectors

Node classification
Link prediction
Graph classification
Entity resolution
Question Answering
… … …

Downstream tasks

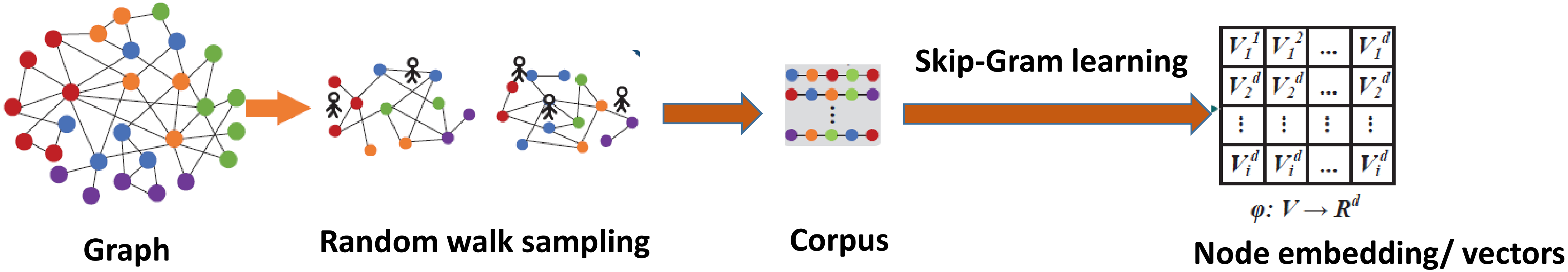
Matrix factorization
Random walk sampling + Skip-Gram learning ✓
Graph convolutional neural networks (GCN) ✓

Machine Learning on Blockchain Graphs

Paper	Embedding Method	Downstream Task
D. Lin, J. Wu, Q. Yuan, and Z. Zheng. ***Modeling and understanding Ethereum transaction records via a complex network approach***. IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II: EXPRESS BRIEFS, VOL. 67, NO. 11, NOVEMBER 2020.	Random walk sampling + Skip-Gram learning	Transaction (link) prediction
D. Lin, J. Wu, Q. Yuan, and Z. Zheng. **T-EDGE: Temporal WEighted MultiDiGraph Embedding for Ethereum transaction network analysis.** Front. Phys., 2020, Sec. Social Physics.	Random walk sampling + Skip-Gram learning	Transaction (link) prediction
F. Poursafaei, R. Rabbany, and Z. Zilic. **SIGTRAN: Signature vectors for detecting illicit activities in Blockchain transaction networks**. PAKDD 2021.	Random walk sampling + Skip-Gram learning + Feature	Detecting illicit activities (node classification)
J. Wu , Q. Yuan, D. Lin , W. You, W. Chen, C. Chen**. Who are the phishers? Phishing scam detection on Ethereum via network embedding.** IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS 2020.	Random walk sampling + Skip-Gram learning	Phishing scams detection (node classification)
L. CHEN, J. PENG, Y. LIU, J. LI, F. XIE, and Z. ZHENG. ***Phishing scams detection in Ethereum transaction network.*** ACM Trans. Internet Technol. 2021.	Graph convolutional neural networks (GCN)	Phishing scams detection (node classification)
T. Yu , X. Chen, Z. Xu, and J. Xu. ***MP-GCN: A phishing nodes detection approach via graph convolution network for Ethereum***. Appl. Sci. 2022.	Graph convolutional neural networks (GCN)	Phishing scams detection (node classification)

Random Walk Sampling + Skip-Gram Learning

○ Transform a graph into a set of random walks through sampling methods, treat each random walk as a sentence, and then adopt word2vec (Skip-Gram) to generate node embeddings from the sampled walks.



$$\underset{\varphi}{\operatorname{argmax}} \frac{1}{|V|} \sum_{j=1}^{|V|} \sum_{-w \le i \le w} \log p(u_{j+i} | u_j)$$

$$\log p(u_j | u_{j+i}) \approx \log \sigma(\varphi_{in}(u_{j+i}) \cdot \varphi_{out}(u_j)) + \sum_{k=1}^{K} \mathbb{E}_{u_k \sim Pn(u)} [\log \sigma(-\varphi_{in}(u_{j+i}) \cdot \varphi_{out}(u_k))]$$

○ DeepWalk (KDD 2014)
○ LINE (WWW 2015)
○ Node2vec (KDD 2016)
○ HuGE (ICDE 2021)

T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. 2013. **Distributed representations of words and phrases and their compositionality**. In NeurIPS
A. Grover and J. Leskovec. 2016. **Node2vec: scalable feature learning for networks**. In KDD.

Random Walk Sampling + Skip-Gram Learning on Blockchain Graphs

○ **Challenges**

- Dynamic/ temporal
- Multi-graph
- Value on edges
- Other node and edge features

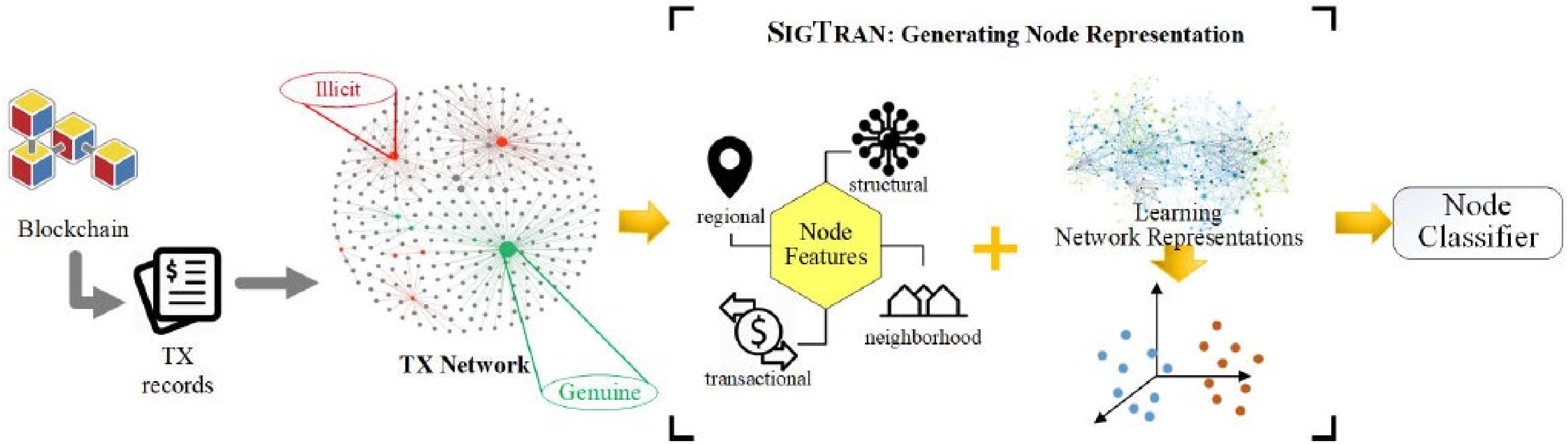
○ **L-length temporal walk:** A sequence of l nodes together with a sequence of (L-1) edges traversed in non-decreasing timestamps

○ **Temporal Biased Sampling (TBS):** Sampling method biases the selection towards edges that are closer (or later) in time to the previous edge.

○ **Weighted Biased Sampling (WBS):** Sampling method biases the selection towards edges with a higher value of transaction amount, implying a larger similarity between the two accounts.

D. Lin, J. Wu, Q. Yuan, and Z. Zheng. **Modeling and understanding Ethereum transaction records via a complex network approach**. IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II: EXPRESS BRIEFS, VOL. 67, NO. 11, NOVEMBER 2020.
D. Lin, J. Wu, Q. Yuan, and Z. Zheng. **T-EDGE: Temporal WEighted MultiDiGraph Embedding for Ethereum transaction network analysis**. Front. Phys., 2020, Sec. Social Physics.
J. Wu , Q. Yuan, D. Lin , W. You, W. Chen, C. Chen. **Who are the phishers? Phishing scam detection on Ethereum via network embedding**. IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS 2020.

Random Walk Sampling + Skip-Gram Learning on Blockchain Graphs



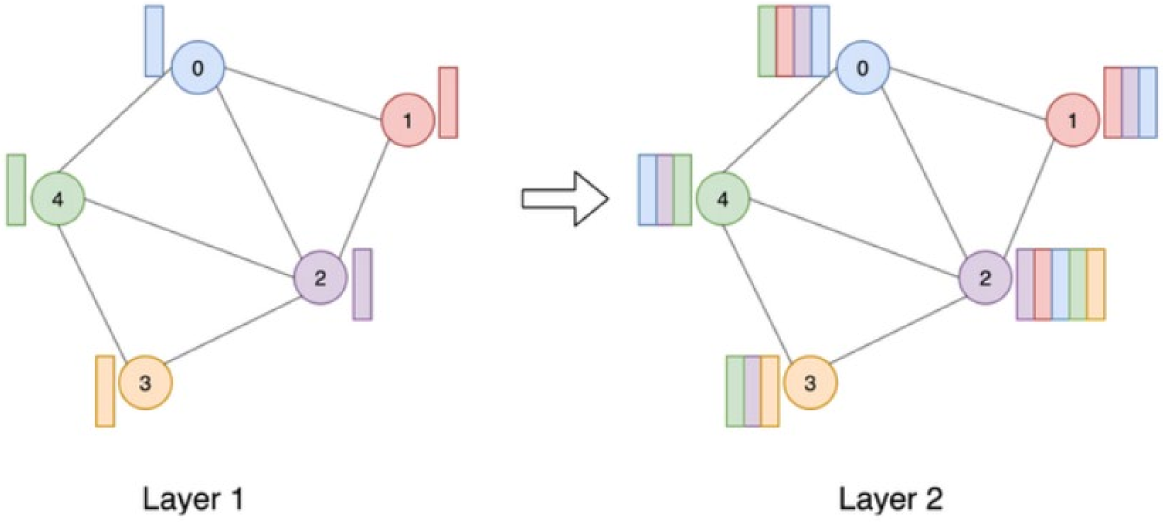
SIGTRAN embedding to detect illicit nodes on a blockchain network

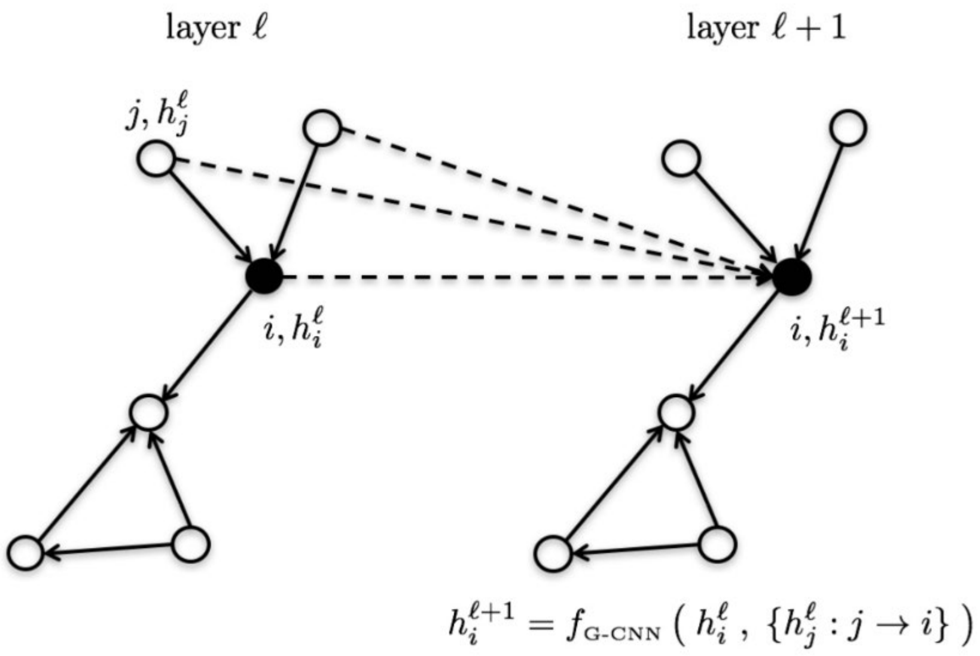
○ **SIGTRAN** extracts a set of useful features which are fused with the corresponding node representations produced by a node embedding method

○ **SIGTRAN** features: structural features (in-degree, out-degree, total-degree); transactional features (amount and time interval of the transactions); regional and neighborhood features (number of edges, features in the egonet)

F. Poursafaei, R. Rabbany, and Z. Zilic. **SIGTRAN: Signature vectors for detecting illicit activities in blockchain transaction netwo**rks. PAKDD 2021.

Graph Convolutional Neural Networks (GCN)



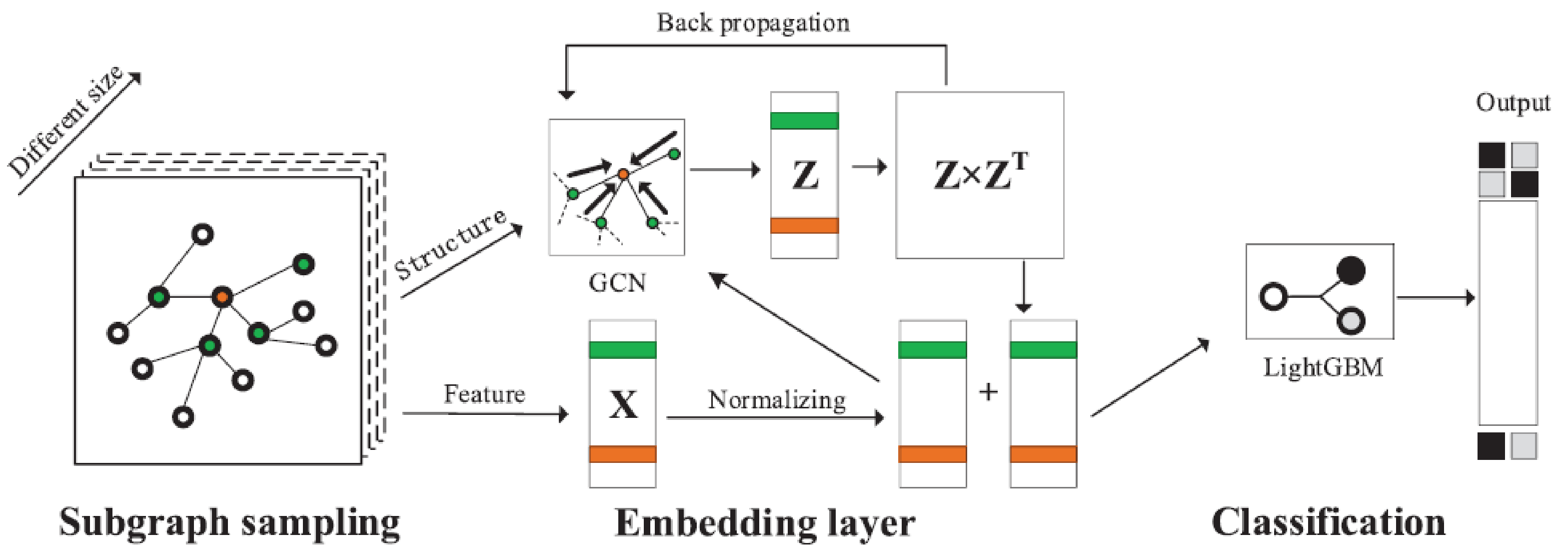


Source: https://graphdeeplearning.github.io/project/spatial-convnets/

$$\mathbf{F}^l(\mathbf{X},\mathbf{A}) = \sigma(\widetilde{\mathbf{D}}^{-1/2}\widetilde{\mathbf{A}}\widetilde{\mathbf{D}}^{-1/2}\mathbf{F}^{l-1}(\mathbf{X},\mathbf{A})\mathbf{W}^l)$$

T. N. Kipf and M. Welling, “**Semi-supervised classification with graph convolutional networks**”, ICLR, 2017.

Graph Convolutional Neural Networks (GCN) on Blockchain



Node embedding and classification based on graph convolutional network and autoencoder

○ In the first step, apply a random walk to sample the subgraph. The orange dots are randomly selected and represent the starting point for the walk.

○ For the obtained subgraphs, features (degree, transaction amount and frequency, no of neighbors, etc.) are extracted and min-max normalized as the feature matrix X.

○ The adjacency matrix and X are fed into GCN with encoder and decoder stage for embedding.

○ As the output of GCN, Z and features' matrix X are concatenated to get the final result for classification.

L. CHEN, J. PENG, Y. LIU, J. LI, F. XIE, and Z. ZHENG. **Phishing scams detection in Ethereum transaction network**. ACM Trans. Internet Technol. 2021.

Applications of Blockchain Data Analytics and Open Problems

Target Applications

- Bulk of the works conducted graph analysis to gain insights into transaction and token transfers.

- Some of them considered downstream tasks, e.g., node classification, link prediction, anomaly detection, token price prediction.

- Most tools for blockchain data are related to e-crime or financial (e.g., price, investor) analytics.

- From ransomware payment detection to sextortion discovery, transaction graph analysis has proven useful to study blockchain address importance and to cluster them.

Oggier, F., Datta, A. and Phetsouvanh, S., 2020. **An ego network analysis of sextortionists**. *Social Network Analysis and Mining*, *10*(1), pp.1-14.

Bistarelli, S., Mercanti, I. and Santini, F., 2018, August. **A suite of tools for the forensic analysis of bitcoin transactions: Preliminary report**. In *European Conference on Parallel Processing* (pp. 329-341). Springer, Cham.

Applications

- Price prediction
  - Cryptocurrencies, tokens, NFTs.
- Unsupervised learning
  - Address clustering: detecting influential investors, exchange addresses.
  - Transaction clustering: linking transactions to an entity (P2P network solutions).
- Supervised learning
  - Address type detection: ransom receiving, money laundering addresses.
  - Transaction type detection: pump and dump, darknet market transactions.
  - Smart contract type prediction: Ponzi schemes.

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

Price, Risk, and Volatility

- Relationship between transaction networks of multiple cryptocurrencies and health of crypto eco-system.

- Network features of cryptocurrencies transactions as a proxy for market sensing.

- Ensemble forecasting of fiat currencies with cryptocurrencies features.

Baur, D.G., Hoang, L.T. and Hossain, M.Z., 2022. **Is Bitcoin a hedge? How extreme volatility can destroy the hedge property**. *Finance Research Letters*, p.102655.

Mokni, K., 2021. **When, where, and how economic policy uncertainty predicts Bitcoin returns and volatility? A quantiles-based analysis**. *The Quarterly Review of Economics and Finance*, *80*, pp.65-73.

Learning and Labels

- Supervised learning: we have external labels on nodes or edges

- What are our node labels:
  - known ransomware coin receiving/forwarding addresses
    - http://chartalist.org/btc/TaskTypePrediction.html
    - How do we know these addresses? Some companies release them when ransomed.
  - potential darknet market addresses
    - https://www.gwern.net/DNM-archives#gramsd2l
    - How do we identify these addresses? We match market item price amounts of a day to output amounts in btc transactions of the day.

https://www.gwern.net/DNM-archives

Darknet Market Archives (2013–2015)

Site
Me

Changes
News

support on
Patreon

Mirrors of ~89 Tor-Bitcoin darknet markets & forums 2011–2015, and related material.

Bitcoin, *Silk-Road*, *shell*, *R*, *dataset*

2013-12-01–2021-03-20 · *finished* · certainty: *highly likely* · importance: *9* · backlinks

Dark Net Markets (DNM) are online markets typically hosted as Tor hidden services providing escrow services between buyers & sellers transacting in Bitcoin or other cryptocoins, usually for drugs or other illegal/regulated goods; the most famous DNM was Silk Road 1, which pioneered the business model in 2011.

From 2013–2015, I scraped/mirrored on a weekly or daily basis all existing English-language DNMs as part of my research into their usage, lifetimes/characteristics, & legal riskiness; these scrapes covered vendor pages, feedback, images, etc. In addition, I made or obtained copies of as many other datasets & documents related to the DNMs as I could.

This uniquely comprehensive collection is now publicly released as a 50GB (~1.6TB uncompressed) collection covering 89 DNMs & 37+ related forums, representing <4,438 mirrors, and is available for any research.

Open Problems

- Investigating graph properties, embeddings, and anomalous patterns.
  - Stablecoins' price stabilization mechanisms (Luna Terra).

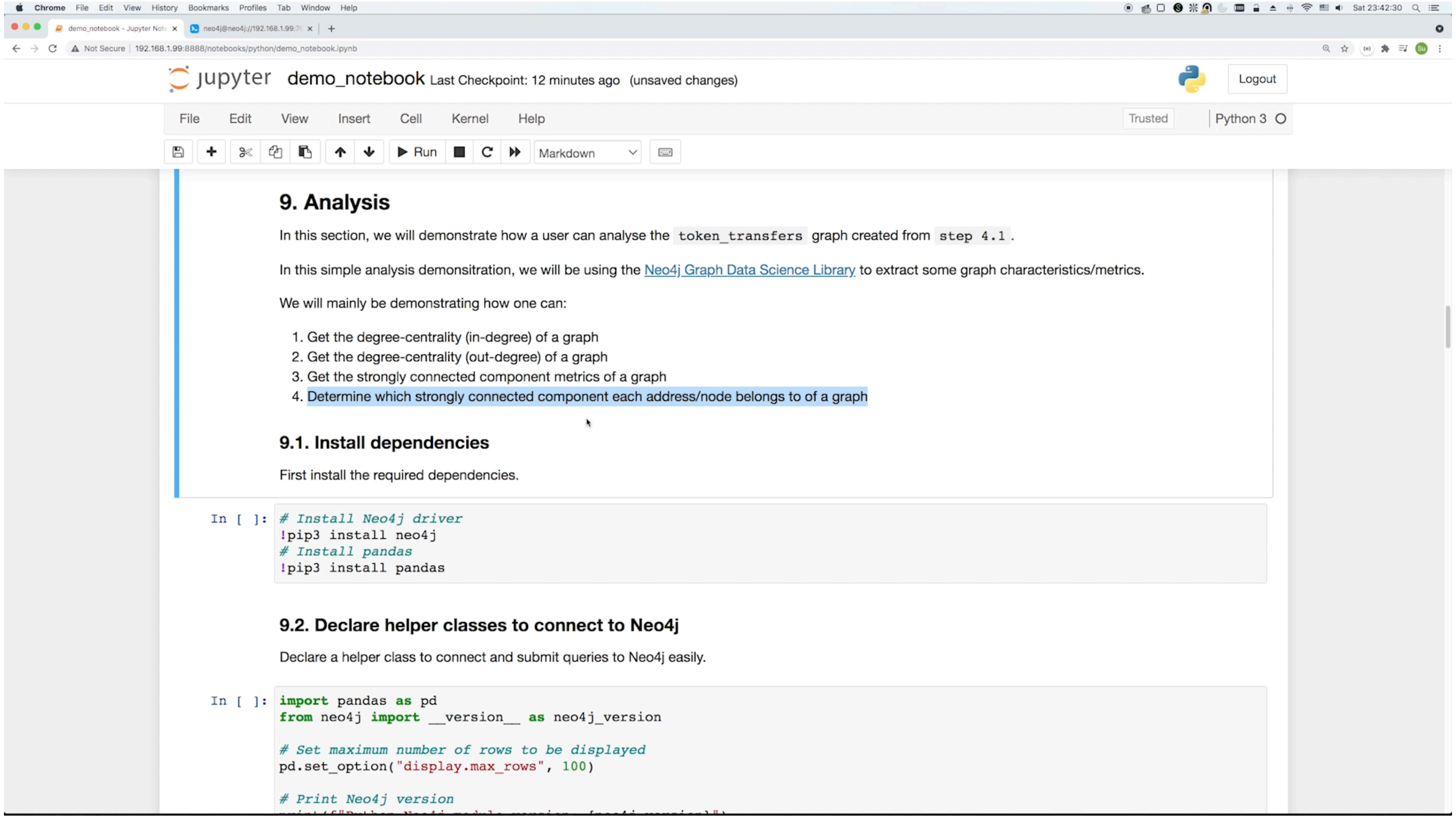
- Multilayer graphs would be an expressive model of real-world activities such as external and internal transactions, token transfers, dApps and DeFi usage.

- Conducting graph analysis in an OLAP (online analytical processing) manner for accounts
  - miners, mining pools, mixers, exchanges, phishing accounts, ICO contracts, gambling games.

- Due to highly dynamic nature of accounts and transactions, employed ML models must deal with data and model drifts.
  - Drift detection, incremental learning, machine unlearning and continuous learning would be useful.

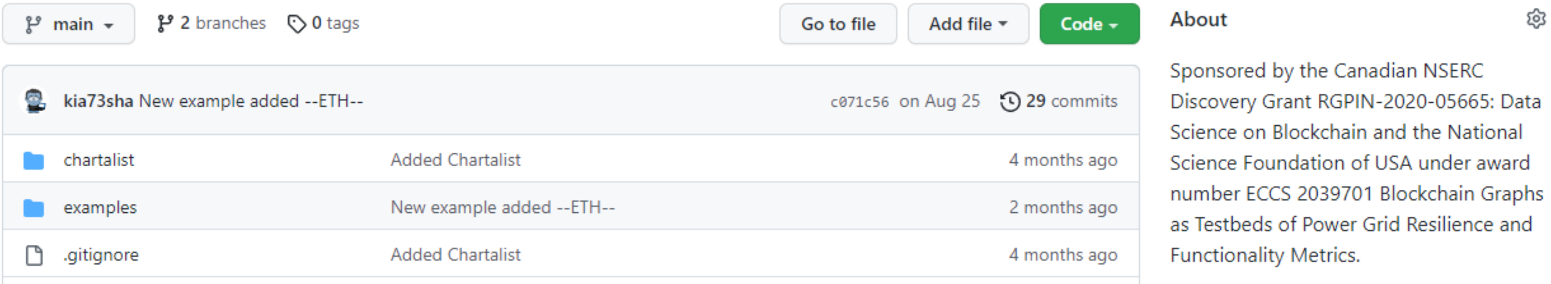
Check out the Ethereum toolbox – open-sourced at:

https://github.com/voonhousntu/ethernet



V. H. Su, S. S. Gupta, A. Khan. ***Automating ETL and mining of Ethereum blockchain network***, WSDM 2022.

https://github.com/cakcora/Chartalist



Chartalist

Chartalist is the first blockchain machine learning ready dataset platform from unspent transaction output and account-based blockchains.

Thanks for attending!

Reach us at

arijitk@cs.aau.dk
cuneyt.akcora@umanitoba.ca

https://twitter.com/cuneytgurcan

https://twitter.com/rijitk