A Probabilistic Model of the Bitcoin Blockchain

Marc Jourdan
Ecole Polytechnique
Palaiseau, France
marc.jourdan@polytechnique.edu

Sebastien Blandin, Laura Wynter, Pralhad Deshpande
IBM Research
Singapore
sblandin,lwynter,pralhad@sg.ibm.com

Abstract

The Bitcoin transaction graph is a public data structure organized as transactions between addresses, each associated with a logical entity. In this work, we introduce a complete probabilistic model of the Bitcoin Blockchain, setting the basis for follow-up AI applications on Bitcoin transactions. We first formulate a set of conditional dependencies induced by the Bitcoin protocol at the block level and derive a corresponding fully observed graphical model of a Bitcoin block. We then extend the model to include hidden entity attributes such as the functional category of the associated logical agent and derive asymptotic bounds on the privacy properties implied by this model. At the network level, we show evidence of complex transaction-to-transaction behavior and present a relevant discriminative model of the agent categories. Performance of both the block-based graphical model and the network-level discriminative model are evaluated on a subset of the public Bitcoin Blockchain.

1. Introduction

Analysis of the Bitcoin Blockchain [30] is an area of intense activity [25, 1], and one which has witnessed an explosion of interest as the value of the Bitcoin cryptocurrency has skyrocketed. Research areas include explorations of address clustering techniques to identify logical agents [26, 10], de-anonymization using side-channel attacks [11, 17].

An understanding of the properties of Bitcoin transactions is paramount to the legitimization of the cryptocurrency economy; it constitutes a building block to the conception of adequate regulations [12], and to the design of novel and integrated services benefiting society as a whole.

As of 2018, with more than 500 million address nodes, the Bitcoin graph is comparable in size to a large social network. Yet while probabilistic models of social networks have received considerable attention, from community detection [24] to diffusion models and probabilistic graph modeling [22], probabilistic models of the Bitcoin Blockchain network have not.

Bitcoin transactions are tantamount to a partially observed social network, within which participants can have multiple seemingly independent aliases. This distinguishes our work from classical studies on partially observed social networks, typically focused on partial observations of interactions due to sampling [14], and makes it closer to the vast body of work on entity resolution [36, 5].

A second challenge associated with modeling the Bitcoin Blockchain transaction network consists of capturing the complexity of the hidden structure associated with entity transactions, together with the fine-grained block-level specificities implied by the Bitcoin protocol. In particular, Bitcoin is based on an *unspent transaction output* (UTXO) model, which distinguishes suitable Bitcoin Blockchain models from prior studies on credit card transactions [8, 23], since the proper generative structure needs to account for the underlying UTXO creation and deletion process [9].

In this work, we propose a first attempt at a comprehensive model of the Bitcoin transaction graph using a hybrid generative-discriminative model attempting to draw strengths from both approaches [31]. The generative approach allows defining a model which complies with the abstracted protocol, but which inherits learning capabilities from graphical models, and could be used in practice to support further AI applications on Bitcoin, such as fraud detection, price prediction, policy evaluation.

Our work departs from the body of related work described in Section 6 centered on discriminative models of the Bitcoin Blockchain focused on de-anonymization, but rather focus on generative modeling of the Blockchain capturing fundamental properties of the protocol and hence able to adequately model Bitcoin transactions. As an application, we consider the problem of de-anonymization, and provide both a theoretical and numerical analysis.

We first define pragmatic conditional independence assumptions underlying the Bitcoin protocol, and formulate a generative model of the Bitcoin Blockchain block. In this context, we analyze the revealed entity behavior, both theoretically and from a data perspective. We then turn to network level modeling, present a discriminative model of transaction-transaction behavior, and analyze the associated medium-term categorical agent behavior.

The rest of the article is organized as follows. Section 2 and 4 define the probabilistic graphical model, Section 3 provides results on transaction privacy, Section 5 consists of numerical experiments. Related work is discussed in Section 6 and concluding remarks are provided in Section 7.

2. Probabilistic block model

A Bitcoin transaction consists of a set of input addresses transferring BTC to a set of output addresses. More specifically, in the context of a transaction, each input address contributes a possibly fractional subset of its UTXOs to the creation of the set of UTXOs associated with output addresses, for the same total amount (minus a fee). Each UTXO is associated with an address, and each address is associated with a logical agent, who may hold an arbitrary number of addresses, see Figure 1.

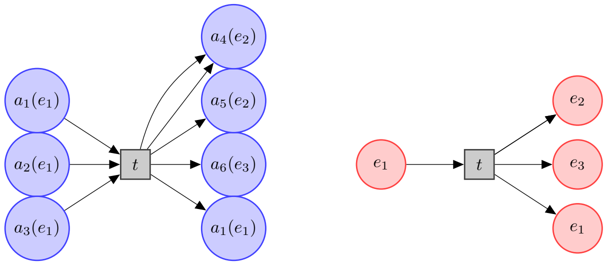

Figure 1. **Bitcoin address-transaction** bipartite graph data structure of visible addresses a_i, (associated with unknown entities e_i), (left) with each arrow corresponding to an *unspent transaction output* (UTXO), and corresponding hidden entity-transaction graph (right). A block consists of many such transactions. A change address, here a_1, may be used to return the remainder of the UTXO.

We embed the Bitcoin Blockchain transaction graph in a directed bipartite graph structure $\mathscr{G} = (\mathscr{A}, \mathscr{T}, \mathscr{E})$, with the following vertex and edge features:

- *address vertex* $a \in \mathscr{A}$: number of UTXO k_a^{UTXO}, and out-degree k_a^{out},
- *transaction vertex* $t \in \mathscr{T}$: transaction value v and fee f,
- *directed address-transaction edge* $\in \mathscr{E}$: outgoing value v from address a via transaction t,
- *directed transaction-address edge* $\in \mathscr{E}$: incoming value v to address a via transaction t.

Since the Bitcoin protocol specifies that transactions should be validated in blocks and the *proof-of-work* consensus protocol incentivizes validators to agree on a single blockchain, we ignore transient disagreements and assume a discrete-time *simple path* structure of blocks. We refer the interested reader to [7] for a more thorough study of relevant Blockchain graph semantics, in particular in the context of *Hyperledger Fabric* [3].

We propose a stationary graphical model [18] of a Bitcoin Blockchain block. First we develop a fully observable *block-transaction, address* (BT-A) model, illustrated in Figure 2, that we then augment with entity attributes into a *block-transaction, entity-address* (BT-EA) model with more complex structure.

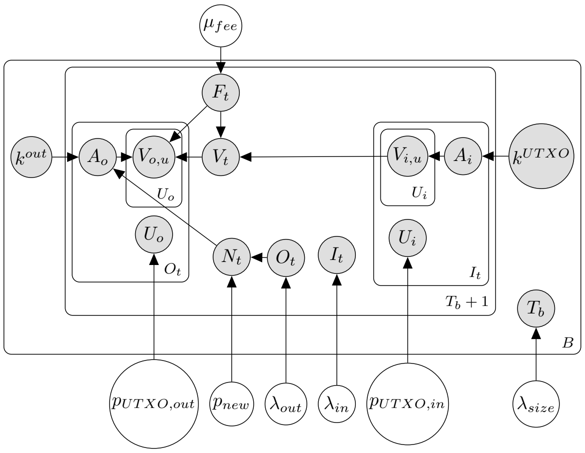

Figure 2. **Block-transaction address** model, plate notation. Observed random variables are shaded while non-observed variables are plain.

2.1. Block-transaction address model

A block b is composed of the set of transactions t validated by the peer node who solved the cryptographic challenge the fastest. With the approximation of stationary inter-block time, and assuming independence between the ability of solving the cryptographic challenge and the selection of transactions, we model the number T_b of transactions per block as a Poisson distribution. Similarly assuming stationary and independent address usage, we model the number of input addresses I_t and output addresses O_t per transaction as a Poisson random variable. The Poisson distribution is chosen because it corresponds to a number of events within a fixed time period.

Definition 1 (Transactions, input and output addresses)**.**

$$\begin{aligned} &\forall b \in \mathscr{B}, \quad T_b \sim \mathscr{P}(\lambda_{size}) \\ &\forall t \in \mathscr{T}, \quad I_t \sim \mathscr{P}_n(\lambda_{in}) \quad \textit{and} \quad O_t \sim \mathscr{P}_n(\lambda_{out}), \end{aligned} \tag{1}$$

where $\mathscr{P}_n$ is the normalized Poisson distribution on $\mathbb{N}^*$.

On the receiving end of a transaction, it is possible to generate a new address. In the Bitcoin pseudonymous context, this reduces the traceability of the full set of transactions associated with an entity. Considering the set of output addresses as a whole, we model the conditional distribution of the number of new addresses given the number of output addresses as a Binomial random variable.

Definition 2 (New address distribution)**.**

$$\forall t \in \mathscr{T}, \quad N_t | O_t \sim \mathscr{B}(O_t, p_{new}). \tag{2}$$

In the interest of a tractable inference procedure, and in the absence of an informative prior, in this work we focus our efforts on maximum-likelihood estimation, and assume uniform prior $\lambda_{size}, \lambda_{in}, \lambda_{out}, p_{new}$.

We now proceed to describe the generative model of the input and output addresses. A natural choice for the generative hierarchical model is the LDA or Dirichlet-multinomial model used in topic modeling [6]. Here, given the full observability of the model variables and decomposability of the likelihood, motivated by topological social network analysis, we use the Albert-Barabasi *preferential attachment* model [4, 2], which can be seen as the posterior probability of an LDA model in the appropriate feature space.

Specifically, we consider that the probability of the i^{th} address A_i to be a given address a is proportional to the number of available UTXO of the address. The model reads as follows.

Definition 3 (Input addresses)**.** $\forall i \in \{1, \ldots, I_t\}$

$$\mathbb{P}(A_i = a | k^{UTXO}) = \frac{k_a^{UTXO} + 1}{\sum_{a' \in \mathscr{A}} (k_{a'}^{UTXO} + 1)} \tag{3}$$

where $\mathscr{A}$ is the set of available addresses, k_a^{UTXO} is the number of unspent outputs of the address.

The output address model is similar, except that the attachment model is now considered a function of the out-degree of the address, i.e. while the inclination of the address to be part of the inputs (i.e. to spend) is considered to be a function of the number of UTXOs it has still available, the inclination of the address to be part of the outputs (i.e. to accumulate) is considered to be a function of the number of distinct UTXOs it has already spent.

Definition 4 (Output addresses)**.** $\forall o \in \{1, \ldots, O_t\}$

$$\begin{aligned} \mathbb{P}(A_o = a | N_t, k^{out}) \sim & \mathbb{1}(o \leq N_t; a = a_0) \\ & + \mathbb{1}(o > N_t) \frac{k_a^{out} + 1}{\sum_{a' \in \mathscr{A}} (k_{a'}^{out} + 1)} \end{aligned} \tag{4}$$

where a_0 denotes a new address.

For each input address, since empirically we observe that the $UTXO$ distribution is concentrated around 1, we model the conditional distribution of the number U_i of UTXOs used given the number of UTXOs available as a geometric random variable with uniform prior. We then draw the UTXOs uniformly from the available set.

Definition 5 (Input UTXOs)**.**

$$\begin{aligned} &\forall i \in \{1, \ldots, I_t\}, U_i | k_{A_i}^{UTXO} \sim \mathscr{G}_{[1,\ldots,k_{A_i}^{UTXO}]}(p_{UTXO,in}) \\ &\forall u \in \{1, \ldots, U_i\}, V_{i,u} | U_i \sim \mathscr{U}_{\{1,\ldots,k_{A_i}^{UTXO}\}} \end{aligned} \tag{5}$$

where $\mathscr{G}_{[1,\ldots,k_{A_i}^{UTXO}]}$ is the normalized geometric distribution with support $[1, \ldots, k_{A_i}^{UTXO}]$, and where $\mathscr{U}_{\{1,\ldots,k_{A_i}^{UTXO}\}}$ is the uniform distribution over the set $\{1, \ldots, k_{A_i}^{UTXO}\}$.

We obtain the total transaction value V_t as the sum of the input UTXOs.

Definition 6 (Transaction value)**.**

$$V_t | I_t, U_i, V_{i,u} = \sum_{1 \leq i \leq I_t} \sum_{1 \leq u \leq U_i} V_{i,u}. \tag{6}$$

A fee is paid to the miners to reward their validation work and higher fees may nudge their selection of transactions when creating blocks. We thus model the fee associated with a Bitcoin transaction as a normalized Gaussian distribution. The number of output UTXOs and their values is modeled similarly to the input UTXOs.

Definition 7 (Fee value, output UTXOs)**.**

$$\begin{aligned} &\forall t \in \mathscr{T}, \quad F_t | V_t = \mathscr{N}_{[0,V_t]}(\mu_{fee}, \sigma_{fee}) \\ &\forall o \in \{1, \ldots, O_t\}, \quad U_o \sim \mathscr{G}(p_{UTXO,out}) \\ &\forall u \in \{1, \ldots, U_o\}, \quad V_{o,u} | V_t, F_t \sim \mathscr{U}_{[1,\ldots,V_t - F_t]} \end{aligned} \tag{7}$$

where $\mathscr{N}_{[a,b]}$ denotes the Gaussian distribution normalized over the interval $[a, b]$, and where $\mathscr{U}$ denotes the normalized uniform distribution (the $V_{o,u}$ are also normalized in order to sum to $V(t) - F(t)$).

The resulting block-transaction address model (1)-(2)-(3)-(4)-(5)-(6)-(7) is presented in Figure 2.

We now turn to a more complex variant of the proposed model meant to capture categorical behavior of the unobserved entities transacting on the Blockchain.

2.2. Block-transaction entity-address model

An entity e is associated with a Bitcoin user and fully characterized by a set of addresses $A(e) = \{a_i^{(e)}\}_i$. In this section we extend the BT-A model to take into account categorical entity behavior. We assume that entities belong to different categories $c \in \mathscr{C}$, with potentially different behaviors.

We first model the fact that the hyper-parameters λ_{in} and λ_{out} associated with the number of input and output addresses, depend on the category c of the associated entity, and are noted $\lambda_{in,c}$ and $\lambda_{out,c}$. Similarly the parameter associated with the number of new addresses in the output

$p_{new,c}$, and the number of UTXO in the input $p_{UTXO,in,c}$ and output $p_{UTXO,out,c}$ are category-dependent.

Second we update the conditional independence structure of the generative model to reflect the fact that address selection (3)-(4) is now also conditioned on entities.

Definition 8 (Input and output entities and addresses)**.**

$$
\begin{aligned}
&\mathbb{P}(E_t = e | k^{UTXO}) = \frac{k_e^{UTXO} + 1}{\sum_{e' \in \mathscr{E}} (k_{e'}^{UTXO} + 1)} \\
&\mathbb{P}(A_i = a | k^{UTXO}, E_t) = \frac{\mathbb{1}(a \in A(E_t))\,(k_a^{UTXO} + 1)}{\sum_{a' \in \mathscr{A}, a \in A(E_t)} (k_{a'}^{UTXO} + 1)} \\
&\mathbb{P}(E_o = e | k^{out}) = \frac{k_e^{out} + 1}{\sum_{e' \in \mathscr{A}} (k_{e'}^{out} + 1)} \qquad (8) \\
&\mathbb{P}(A_o = a | k^{out}, E_o) = \mathbb{1}(o \leq N_t; a = a_0) \\
&\qquad + \mathbb{1}(o > N_t) \frac{\mathbb{1}(a \in A(E_o))\,(k_a^{out} + 1)}{\sum_{a' \in \mathscr{A}, a \in A(E_o)} (k_{a'}^{out} + 1)}.
\end{aligned}
$$

This dependency structure intending to capture the behavior of distinct categories of entities is illustrated in Figure 3.

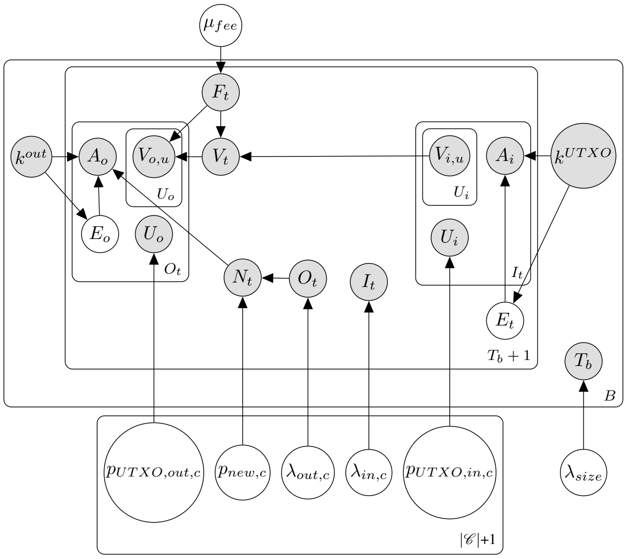

Figure 3. **Block-transaction entity-address** model, including category-specific variables, as well as hidden entities. As per the protocol, all input addresses are associated with only 1 entity, while output addresses are generally associated with multiple entities.

2.3. Model inference

We assume a known dependency structure as described above, hence we do not require learning the structure, and we simply estimate the model parameters. Since the prior is decomposable over nodes, and since all variables are observed in the BT-A model, the MLE inference amounts to local computation over each node and its parents.

Regarding the BT-EA model, while the hidden entity variables make the inference more complex in general, here we assume that a separate heuristic such as the multi-input heuristic [10] allows associating each address with an entity, hence the inference process over the labeled set reduces to the scalable process used for the BT-A model.

3. Block-level privacy analysis

In this section we present an analysis of address re-use behavior in the context of the probabilistic model introduced in the previous section, as well as implications of these results for Bitcoin transaction anonymity.

3.1. Attacker model

We model an attacker, attempting to identify the full set of addresses $A(e)$ associated with an entity e. We assume that the attacker uses the standard multi-input heuristic [10], which associates the full set of address inputs for each transaction to a single entity and applies transitive closure. From the perspective of the external attacker, the true set of addresses $A(e)$ of an entity e is partitioned into A_e aliases, a-priori seen as distinct entities;

$$A(e) = \bigcup_{1 \leq i \leq A_e} A(e)_i,$$

where $A(e)_i$ denotes the address set associated with alias i of entity e. In this setting, when participating in a transaction t on the input side, we consider that the targeted entity e selects $\{N_{in,i}\}_{1 \leq i \leq A_e}$ addresses from its available set following a generic multinomial distribution with parameters $\{p_i\}_{1 \leq i \leq A_e}$, which includes the special case for which the alias distribution is a linear function of k^{UTXO}.

This models the typical Bitcoin user who, while being concerned by his privacy, is not particularly careful about address selection, and uses multiple distinct aliases with distinct address sets, but sometimes mixes these address in the same transaction input, leading to a privacy collapse.

Given the multi-input heuristic, it is indeed sufficient for an attacker to observe two addresses from distinct aliases and to associate these two aliases to the same entity using the multi-input heuristic. Formally, upon observing the input addresses from a transaction t associated with input entity e, the attacker is able to associate the following address set with entity e:

$$\{a \in A(e)_i \cup A(e)_j | \mathbb{1}(N_{in,i} > 0;\, N_{in,j} > 0), 1 \leq i, j \leq A_e\}.$$

3.2. Privacy analysis

In the following for simplicity we consider a one-step iteration and assume that the attacker is only aware of the set of addresses associated with alias $A(e)_1$. In this sense the control parameter p_1 plays the role of 1-p_{new} from the BT-EA model. We analyze the number $D_{e,t}$ of addresses from entity e that the attacker is able to discover after seeing the addresses involved in 1 transaction, expressed as:

$$D_{e,t} = \sum_{i=2}^{A_e} |A(e)_i| \mathbb{1}(N_{in,i} > 0;\ N_{in,1} > 0).$$

We can express the number of discovered addresses $D_{e,t}$ as a function of the alias addresses selection probabilities p_i.

Proposition 1 (Privacy loss from address re-use)**.**

$$\mathbb{E}[D_{e,t}] = \frac{1 - \exp(-\lambda_{in,c} p_1)}{1 - \exp(-\lambda_{in,c})} \sum_{i=2}^{A_e} |A(e)_i|(1 - \exp(-\lambda_{in,c} p_i)) \tag{9}$$

Proof. By definition of D_e, t, we have

$$\mathbb{E}[D_{e,t}] = \sum_{i=2}^{A_e} |A(e)_i| \mathbb{P}(N_{in,i} > 0 \wedge N_{in,1} > 0 | E_t = e).$$

Let B be the second factor in the summation term, by marginalizing over I_t and using the chain rule, we can write:

$$\begin{aligned} B &= \mathbb{P}(N_{in,i} > 0 \wedge N_{in,1} > 0 | E_t = e) \\ &= \sum_{n \geq 1} \mathbb{P}(N_{in,i} > 0 \wedge N_{in,1} > 0 | E_t = e, I_t = n) \\ &\qquad\qquad\qquad\qquad \mathbb{P}(I_t = n | E_t = e). \end{aligned}$$

Letting C denote the first factor in the summation term above, we have:

$$\begin{aligned} C &= \mathbb{P}(N_{in,i} > 0 \wedge N_{in,1} > 0 | E_t = e, I_t = n) \\ &= \sum_{n_i > 0, n_1 > 0, \sum_{j=1}^{A_e} n_j = n} \mathbb{P}(\{N_{in,j} = n_j\}_j | E_t = e, I_t = n) \\ &= \sum_{n_i > 0, n_1 > 0, \sum_{j=1}^{A_e} n_j = n} \frac{n!}{\prod_{j=1}^{A_e} n_j!} \prod_{j=1}^{A_e} p_j^{n_j} \end{aligned}$$

where the last equality is obtained by definition of the multinomial distribution. Similarly since the number of input addresses I_t follows a binomial distribution we have:

$$\mathbb{P}(I_t = n | E_t = e) = \frac{\lambda_{in,c}^n}{n!(\exp(\lambda_{in,c}) - 1)},$$

and combining this expression with the expression of C, we can simplify the expression of B to finally obtain equation (9), which concludes the proof. □

With p_1 as the control parameter, the expression states that the attacker information gain is an exponential function of the probability of using addresses already identified (i.e. address re-use). The asymptotic behavior of a privacy-conscious user is described next.

Proposition 2 (Privacy-conscious asymptotics)**.** *If* $p_1 \ll pi$ *we have:*

$$\mathbb{E}[D_{e,t}] \sim \lambda_{in}\, p_1(|A(e)| - |A(e)_1|).$$

This result shows that the one-step information gain from the attacker is a linear function of the probability of using already-used addresses, and also linear in the number of addresses typically used as input. This result at the transaction level can be readily extended to a chain-length estimate by accounting for the probability of an entity to transact, as provided explicitly in equation (8) of the BT-EA model. We also highlight that while a low p_1 models a privacy-conscious user, the user strategy is non-adaptive, in the sense that the user does not try to adjust his strategy based on the attacker strategy.

4. Probabilistic transaction graph model

We now consider the behavior of entities across transactions, and assume that entity categories exhibit different behaviors. Given the lack of a-priori underlying modeling structure to this behavior, and given the combinatorial nature of such behavior, we propose a discriminative framework in which model selection can be carried out more efficiently based on a possibly large set of relevant features. We rely on the classical multi-input heuristic [10] for defining entities, and formulate a decision-tree based classification problem in the following feature space.

We consider the following five feature classes, and for continuous features explicitly consider the feature mean and standard deviation; address features, entity features, temporal features, graph centrality metric features, motif features.

Address-specific features include attributes such as the total BTC received, the total BTC balance, the number of input/output transactions, etc. Analogous features are defined at the entity level as well as the number and proportion of Coinbase transactions (indicative of BTC creations).

Temporal features are those such as the number of weeks, months, years of activity. the number of entity traded with per week, month, year, the number of receiving/sending/receiving sending days, the activity period duration, and the active day ratio. Graph centrality metric features are standard features.

Motif features are presented in Figure 4. Here we consider 1, 2, and 3 motifs, extending the 2-motifs from [33]. Motifs are a comprehensive description of the transaction-to-transaction properties.

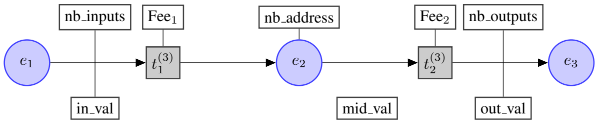

Figure 4. **2-motif features** (rectangular white boxes) annotated over a 2-motif. A N-motif is a path of length $2N$ on the bipartite entity-transaction graph. We distinguish Direct motifs from Loop motifs, the latter indicating that an entity is transacting with itself using distinct addresses.

5. Experiments

In this section we present numerical results of our probabilistic Bitcoin Blockchain model. We first describe the training procedure for the generative block model and discuss obtained model parameters. We then turn to the transaction-to-transaction discriminative model results and analyze the properties revealed by the joint analysis.

5.1. Dataset

We consider the set of blocks of height inferior or equal to $514,971$, corresponding to blocks created before March 24th 2018, 15:19:02, which contains about $500,000,000$ addresses. Address labels, revealing entity identifiers, are obtained from WalletExplorer `https://www.walletexplorer.com/`. The set of address entity label pairs used has been made available at `https://github.com/Maru92/EntityAddressBitcoin`.

We interact with the Blockchain via the BlockSci toolbox v.0.4.5 released on March 16th 2018 [20], on a 64 GB machine. The final labeled dataset used in numerical experiments consists of $28,353,493$ addresses, associated with $|\mathscr{E}_{known}| = 260$ entities representing 4 entity categories in the following proportions:

- *Exchange* (E): 108 entities, 7,892,587 addresses,
- *Service* (S): 68 entities, 17,606,608 addresses,
- *Gambling* (G): 65 entities, 2,775,810 addresses,
- *Mining Pool* (M): 19 entities, 78,488 addresses.

When training the probabilistic model, we restrict ourselves to the period from January 1st 2016 to March 16th 2018, where overall patterns are relatively stationary. Indeed since the proposed model is static we do not attempt to study its ability to model transient regimes. We observe $UTXO$ statistics in Table 1 and $UTXO$ distribution in Figure 5, showing wide variability across multiple scales.

5.2. Transaction subset modeling

Since we consider a subset of the transaction graph, we need to model transactions originating from our subset and directed outside it, or vice-versa. We follow the proposed model structure and model the number of external output addresses as a Poisson distribution $\mathscr{P}(\lambda_{sub})$. Transactions from unknown addresses towards known input addresses are modeled with no known input and a number of transactions per block $T_{b,incoming}$ following a Poisson distribution $\mathscr{P}(\lambda_{size,sub})$. Coinbase transactions are created in a similar manner: no inputs, number of addresses in the outputs drawn following a Poisson distribution of parameter $\lambda_{out,sub}$, with new addresses, $p_{new,sub}$, and several UTXOs created per addresses, $p_{UTXO,out,sub}$.

Quantity	E	S	G	M	All
mean $\mu(V_{u,o})$	8.62	0.53	0.11	1.27	4.39
std $\sigma(V_{u,o})$	93.1	41.6	0.81	4.25	70.0

Table 1. **UTXO empirical statistics in BTC:** the UTXO output values have a large standard deviation compared to their mean, and vary significantly across entity categories.

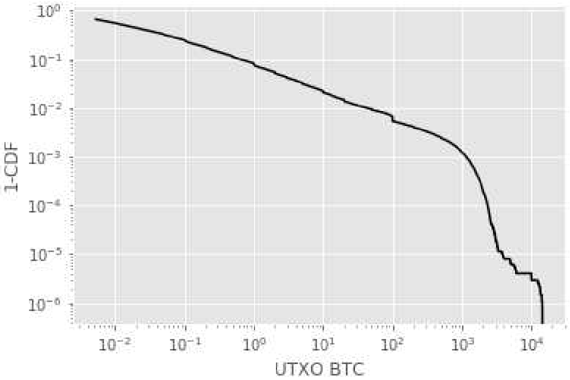

Figure 5. **BTC UTXO distribution :** 1 minus the cdf of the UTXO represented in log log coordinates, with 99.9% of the distribution qualitatively following a power law on the interval $[10^{-3}, 10^3]$.

5.3. Block model training

We train the model using data from the period January 1st 2016 to March 16th 2018 consisting of about 10 million addresses. We first verify the main independence assumption, between the number of input addresses and the number of output addresses. Since $\rho_{pearson}(I_t, O_t) = 0.015$, we consider the marginal independence hypothesis validated.

The inference produces a value $\lambda_{size} = 65.6$ for both models. In Table 2 we present the model parameter results from the model training for the BT-A and BT-EA models.

The results reflect the idiosyncratic properties of Bitcoin Blockchain transactions, with for instance the need to gather UTXOs from various addresses, which is illustrated by the fact that $\lambda_{in} > \lambda_{out}$. It is also clear from the UTXO parameters that the input parameters are more discriminative than the output parameters, which reflect transfers from other parties from the perspective of the entity concerned.

Parameter	BT-EA				BT-A
	E	S	G	M	All
$P(E_t = e)$	0.33	0.55	0.09	0.03	1
λ_{in}	3.79	2.58	1.98	21.2	2.99
λ_{out}	0.68	1.96	0.21	7.04	1.21
$p_{UTXO,in}$	0.95	0.92	0.84	0.67	0.92
$p_{UTXO,out}$	1.00	1.00	1.00	1.00	1.00
p_{new}	0.23	0.20	0.47	0.55	0.26

Table 2. **Model parameters** from calibration on the period from January 1st 2016 to March 16th 2018, for the Exchange, Services, Gambling, Mining Pool categories.

Lastly we observe significant address generation distinctions across entity categories, with Gambling and Mining Pools seemingly more privacy-conscious given their higher probability of generating new addresses. They also transact less frequently, using more input addresses. Detailed impact of entity behavior on privacy properties is analyzed subsequently.

5.4. Block model testing

In order to assess the model performance, we now evaluate out-of-sample model accuracy. Starting from scratch, we train the model on 4911 blocks corresponding to the period from January 1st 2017 to January 31st, 2017, and evaluate the model on 2150 blocks associated with the period from February 1st, 2017, to February 14th, 2017.

Metric	BT-EA				BT-A
	E	S	G	M	All
MSE	1.22	-0.30	-0.02	0.06	1.12
RMSE	125	53.3	1.15	5.19	90.5
MAE	15.6	0.94	0.20	2.42	7.47
RMAE	1.82	1.74	1.86	1.93	1.69
NRMSE	1.34	1.28	1.42	1.22	1.29

Table 3. **Error statistics in BTC for UTXO output values $V_{u,o}$:** from the BT-A level overall value, as well as per category from the BT-EA model, for the Mean Signed Error (MSE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Relative Mean Absolute Error (RMAE) and Normalized Root Mean Squared Error (N-RMSE) expressed as RMSE divided by $\sigma(V_{u,o})$.

The results from Table 3 illustrate that given the multi-scale nature of the underlying distributions, the model estimates are relatively close on average, i.e. well within an order of magnitude. Furthermore, the BT-EA model significantly reduces the bias (MSE) as well as the variance (RMSE) for most categories. The Exchange category is the only one for which both bias and variance increase, suggesting a fundamental modeling limitation.

The error terms are relatively large in absolute terms for both models, which is largely explained by the inherent variance in the data, both at the population level and at the class level. Indeed, the bias is low and most of the data variance is explained, with a N-RMSE ranging between 1.22 and 1.34.

5.5. Privacy analysis validation

Given the calibrated model parameters, we now validate experimentally the theoretical privacy properties of Bitcoin Blockchain transactions expressed by equation (9). We leverage the generative model and attacker model described above to simulate transaction traces and evaluate the proportion of the addresses that are re-identified for distinct categories, as a function of the number of transactions.

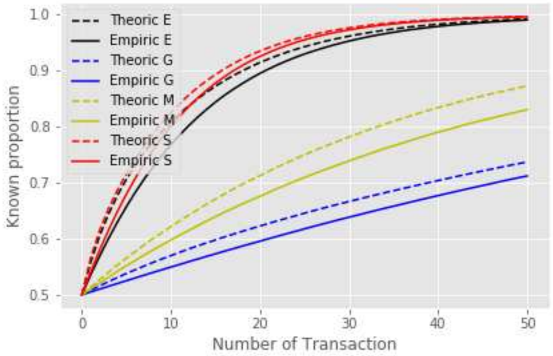

Figure 6. **Proportion of identified addresses:** by category, as a function of the number of transactions.

Figure 6 shows agreement between the analytics results and the simulation of the block model. The figure also illustrates that Exchanges and Services typically are less privacy-conscious (lower probability of generating new addresses, frequent transactions), and hence for an equivalent number of Blockchain transactions, typically reveal a greater proportion of their address set.

Transaction anonymity however depends also on the transaction-to-transaction behavior. Indeed, it is conceivable that certain entities, while not following best block-level practices on address re-use, hence easily identifiable as entities, could be transacting in a way that little information is gathered from their network level transaction structure. In order to assess the latter, we now turn to the numerical results of our proposed network transaction model.

5.6. Transaction network model

We use the Python LightGBM implementation of the gradient boosted decision tree model [21] with a $70/30$ training/test partition of our dataset. A Gaussian Process (GP)-based optimization procedure for hyper-parameter optimization is implemented using the Python *skopt* library `https://scikit-optimize.github.io/` with initial parameter values obtained from a coarse random search. The learning rate hyper-parameter is optimized over the interval $[0.01, 0.5]$ with early stopping after having done a random search over $[0, 2]$; the resulting value is 0.18. The GP procedure is used with 50 iterations.

We make use in total of 10 address features, 8 entity features, 16 temporal features, 42 centrality features, 44 1-motif features, 81 2-motif features, and 114 3-motif features. We present in Table 4 the F1, Accuracy and Precision results over the entire dataset and for each category.

Category	Accuracy	F_1	Precision
Exchange	0.94	0.92	0.91
Gambling	0.95	0.97	1.00
Mining	0.50	0.67	1.00
Service	0.95	0.88	0.83
Overall	0.92	0.91	0.92

Table 4. **Classification performance:** for the 4 entity categories considered, and overall.

The results illustrate that the model is able to very well capture the behavior of most entity categories. Furthermore, the network-level privacy analysis confirms the prior block-level analysis, with Mining Pools being the most privacy-conscious. Indeed, considering the most relevant features of the LightGBM model, in a 1 vs. all setting, it appears that for most categories except the Mining Pool, motif features are the most informative, indicating that the LightGBM model is not able to leverage the transaction subgraph for identification of the Mining Pool category.

6. Related work

A number of studies on the graph properties of the Bitcoin Blockchain transaction graphs have analyzed statistics and structure of vertices and edges [35, 13].

Heuristics for clustering multiple addresses to an entity have been studied in [26] and consistent address re-use patterns have been shown in [15].

Analysis of the Bitcoin protocol in the context of attacks have been proposed, for instance inference of peer-to-peer communication structure, in [11], statistical analysis of bloom filters in [32], and analysis of Bitcoin minting patterns in [27] with application to de-anonymization. Flow-based address-transaction graph studies can be found in [28, 16]. The obfuscation of Bitcoin transactions traceability has been considered in [29]. Reference to using side informations can be found in [34].

Several studies have applied discriminative models to the problem of de-anonymizing Bitcoin transactions, with for instance the use of transaction-specific features in [37], able to achieve 70% accuracy for classifying entities into several types. In [33], the authors introduce transactions paths with application to the detection of Bitcoin exchanges, and achieve greater than 80% accuracy. Similar transactions paths features are used in [19] for a 5-class classification problem with above 90% accuracy results.

7. Conclusion

In this work, we proposed a probabilistic model of the Bitcoin Blockchain which accounts for the complex Bitcoin protocol features. The model consists of a hierarchical structure from unspent transaction output (UTXO), to address, transaction, and block. We take into account entity modeling, including features relevant for robustness to de-anonymization attacks, namely address re-use patterns. We also propose a discriminative model of transaction-to-transaction behavior and show its effectiveness in practice.

We analyzed the accuracy of the generative model using a large Bitcoin dataset of more than 10 million address vertices, discussed the significant block-level heterogeneity of the model parameters across entity categories, and provide a complementary analysis of transaction-to-transaction behavior using the discriminative model. We consider in particular the de-anonymization properties of certain behaviors, which is one of the main focus areas of Bitcoin studies.

Extensions of this work may include the design of more complex graphical models including latent variables for modeling transaction intent, and shared side-information across entities, inducing multivariate preferential attachment. A significant challenge for such models with more complex dependency structure and hidden variables is the design of a tractable training and inference procedure given the large-scale nature of such public cryptocurrency transaction graphs.

References

[1] Cuneyt Gurcan Akcora, Yulia R. Gel, and Murat Kantarcioglu. Blockchain: A graph primer. *CoRR*, abs/1708.08749, 2017.

[2] R Albert and AL Barabasi. Topology of evolving networks: Local events and universality. *Physical Review Letters*, 2000.

[3] Elli Androulaki, Artem Barger, Vita Bortnikov, Christian Cachin, Konstantinos Christidis, Angelo De Caro, David Enyeart, Christopher Ferris, Gennady Laventman, Yacov Manevich, Srinivasan Muralidharan, Chet Murthy, Binh Nguyen, Manish Sethi, Gari Singh, and Keith Smith. Hyperledger fabric: a distributed operating system for permissioned blockchains. In *EuroSys*, pages 1–15. ACM, 2018.

[4] AL Barabasi and R Albert. Emergence of scaling in random networks. *Science*, 1999.

[5] Indrajit Bhattacharya and Lise Getoor. A latent dirichlet model for unsupervised entity resolution. In *Proceedings of the 2006 SIAM International Conference on Data Mining*, pages 47–58. SIAM, 2006.

[6] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3(Jan):993–1022, 2003.

[7] Christian Cachin, Angelo De Caro, Pedro Moreno-Sanchez, Björn Tackmann, and Marko Vukolic. The transaction graph for modeling blockchain semantics. *IACR Cryptology ePrint Archive*, 2017:1070, 2017.

[8] Andrea Dal Pozzolo, Giacomo Boracchi, Olivier Caelen, Cesare Alippi, and Gianluca Bontempi. Credit card fraud detection and concept-drift adaptation with delayed supervised information. In *Neural Networks (IJCNN), 2015 International Joint Conference on*, pages 1–8. IEEE, 2015.

[9] Sergi Delgado-Segura, Cristina Pérez-Sola, Guillermo Navarro-Arribas, and Jordi Herrera-Joancomartı. Analysis of the bitcoin utxo set. In *Proceedings of the 5^{th} Workshop on Bitcoin and Blockchain Research Research, Lecture Notes in Computer Science*, 2018.

[10] Dmitry Ermilov, Maxim Panov, and Yury Yanovich. Automatic bitcoin address clustering. In *Machine Learning and Applications (ICMLA), 2017 16th IEEE International Conference on*, pages 461–466. IEEE, 2017.

[11] Giulia Fanti and Pramod Viswanath. Deanonymization in the bitcoin P2P network. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1364–1373, 2017.

[12] Y.J. Fanusie and T. Robinson. Bitcoin laundering: An analysis of illicit flows into digital currency services. *Center on Sanctions and Illicit Finance, Elliptic*, 2018.

[13] Michael Fleder, Michael S. Kester, and Sudeep Pillai. Bitcoin transaction graph analysis. *ArXiv e-prints*, arxiv:1502.01657, 2015.

[14] Mark S Handcock and Krista J Gile. Modeling social networks from sampled data. *The Annals of Applied Statistics*, 4(1):5, 2010.

[15] Martin Harrigan and Christoph Fretter. The unreasonable effectiveness of address clustering. *IEEE UIC/ATC/ScalCom/CBDCom/IoP*, pages 368–373, 2016.

[16] Danny Yuxing Huang, Maxwell Matthaios Aliapoulios, Vector Guo Li, Luca Invernizzi, Elie Bursztein, Kylie McRoberts, Jonathan Levin, Kirill Levchenko, Alex C Snoeren, and Damon McCoy. Tracking ransomware end-to-end. In *2018 IEEE Symposium on Security and Privacy (SP)*, pages 618–631. IEEE, 2018.

[17] Husam Al Jawaheri, Mashael Al Sabah, Yazan Boshmaf, and Aiman Erbad. When A small leak sinks A great ship: Deanonymizing tor hidden service users through bitcoin transactions analysis. *CoRR*, abs/1801.07501, 2018.

[18] Michael I. Jordan. Graphical models. *Statist. Sci.*, 19(1):140–155, Feb. 2004.

[19] Marc Jourdan, Sebastien Blandin, Laura Wynter, and Pralhad Deshpande. Characterizing entities in the bitcoin blockchain. In *International Conference on Data Mining*. IEEE, 2018.

[20] Harry A. Kalodner, Steven Goldfeder, Alishah Chator, Malte Möser, and Arvind Narayanan. Blocksci: Design and applications of a blockchain analysis platform. *CoRR*, abs/1709.02489, 2017.

[21] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems (NIPS)*, pages 3146–3154, 2017.

[22] Ugur Kuter and Jennifer Golbeck. Sunny: A new algorithm for trust inference in social networks using probabilistic confidence models. In *AAAI*, volume 7, pages 1377–1382, 2007.

[23] Bertrand Lebichot, Fabian Braun, Olivier Caelen, and Marco Saerens. A graph-based, semi-supervised, credit card fraud detection system. In *International Workshop on Complex Networks*, pages 721–733. Springer, 2016.

[24] Jure Leskovec, Kevin J Lang, Anirban Dasgupta, and Michael W Mahoney. Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. *Internet Mathematics*, 6(1):29–123, 2009.

[25] M. Lischke and B. Fabian. Analyzing the bitcoin network: The first four years. *Future Internet*, 8(1):7, 2016.

[26] D. D. F. Maesa, A. Marino, and L. Ricci. Uncovering the bitcoin blockchain: An analysis of the full users graph. In *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pages 537–546, Oct. 2016.

[27] D. McGinn, D. McIlwraith, and Y. Guo. Toward open data blockchain analytics: A bitcoin perspective. *CoRR*, abs/1802.07523, 2018.

[28] Sarah Meiklejohn, 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*, pages 127–140. ACM, 2013.

[29] Malte Möser and Rainer Böhme. The price of anonymity: empirical evidence from a market for bitcoin anonymization. *J. Cybersecurity*, 3(2):127–135, 2017.

[30] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system, 2008.

[31] Andrew Y Ng and Michael I Jordan. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In *Advances in neural information processing systems (NIPS)*, pages 841–848, 2002.

[32] Jonas David Nick. Data-driven de-anonymization in bitcoin. Master's thesis, ETH-Zürich, 2015.

[33] Stephen Ranshous, Cliff A Joslyn, Sean Kreyling, Kathleen Nowak, Nagiza F Samatova, Curtis L West, and Samuel Winters. Exchange pattern mining in the bitcoin transaction directed hypergraph. In *International Conference on Financial Cryptography and Data Security*, pages 248–263. Springer, 2017.

[34] Fergal Reid and Martin Harrigan. An analysis of anonymity in the bitcoin system. In *SocialCom/PASSAT*, pages 1318–1326. IEEE, 2011.

[35] Dorit Ron and Adi Shamir. Quantitative analysis of the full bitcoin transaction graph. In *Financial Cryptography and Data Security*, pages 6–24. Springer Berlin, 2013.

[36] Parag Singla and Pedro Domingos. Entity resolution with markov logic. In *International Conference on Data Mining (ICDM)*, pages 572–582. IEEE, 2006.

[37] Kentaroh Toyoda, Tomoaki Othsuki, and P. Takis Mathiopoulos. Multi class bitcoin-enabled service identification based on transaction history summarization. In *IEEE Conference on IoT, GCC, CPSC, SD, B, CIT, Congress on Cybermatics*, 2018.