Bitcoin Change Detection with Deep Learning

Peter Wang (pwang01@stanford.edu)

June 10, 2018

Introduction

Bitcoin, in its euphoric rise, has captivated millions with its ability to ascribe value to virtual bits in a consensus protocol. However, criminals and those inclined to hide their identities (including Silk Road) have flocked to Bitcoin because of widespread misconceptions about its anonymity and security. Fortunately, clustering and embedding techniques in deep learning can hopefully cluster addresses into singular identities; the revelation of identity of one of the addresses, then, reveals the owner of all of them. The question, then arises: given the graph structure of Bitcoin transactions, can one cluster addresses into belonging to the same owner? A successful model could undermine the promises of anonymity and assist authorities in prosecuting those engaging in illicit activities, as well as augment companies' risk management strategies. This particular model aims to take Bitcoin transactions and identify the "change addresses," which are outputs still controlled by the owner of the inputs.

Related Work

Current work around change address detection involves the use of several human-generated heuristics. For example, a change address is unlikely to be larger than the smallest input to the transaction (otherwise that input would be obviated). Additionally, because most transactions are initiated by only one owner, it is a reasonable assumption to cluster all of the inputs together as one owner. Then, transactions in the future can be used to indicate that certain output addresses are of the same owner as input addresses, and that gives a change address dataset. However, I was unable to find research done in the particular problem that incorporated deep learning.

References:
1. http://www.atmos.umd.edu/~ide/data/teaching/amsc663/14fall/amsc663_14proposal_stefan_poikonen.pdf 2. https://jonasnick.github.io/papers/thesis.pdf 3. http://www.livebitcoinnews.com/bitcoin-address-clustering-new-heuristics-part-1/ 4. https://fc17.ifca.ai/bitcoin/papers/bitcoin17-final11.pdf 5. https://jonasnick.github.io/slides/2016-zurich-meetup.pdf

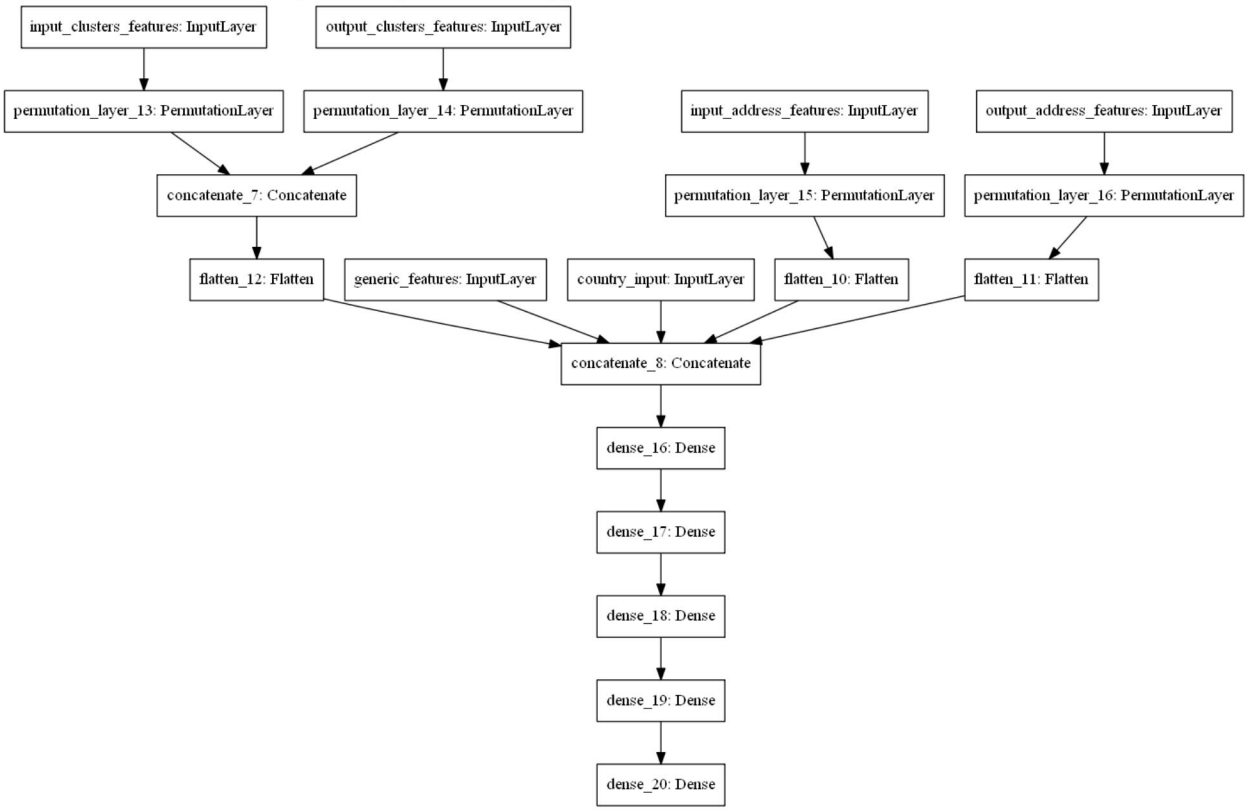
Dataset

The dataset is a raw MySQL database of lightly-annotated Bitcoin transactions on the blockchain. It includes the inputs and outputs of every transaction, as well as some basic information about the amounts transferred, and the past transaction history per address (created by MySQL through queries). Additionally, the dataset contains some basic heuristic clustering information.

Approach

I first decided to try to solve the simpler problem of detecting the change address from the outputs of a transaction. (In Bitcoin, all of the input Bitcoin to a transaction must be put into an output, with each transaction having multiple outputs.) The neural network is a standard feedforward neural network, with features such as input addresses, input amounts, input clustering information, output addresses, output amounts, and whether the output addresses being used for the first time. However, the ordering of the clustering information will affect the network; additionally, the output of the network is a sigmoid-activated binary vector that corresponds to the input order of the output address features. This requires a degree

of permutation equivariance: swapping the order of the output address features (since this is arbitrarily chosen), should swap the order of the output. Therefore, I decided to write a permutational layer based on the idea in Guttenberg et al (https://arxiv.org/pdf/1612.04530.pdf), which involves parameter sharing. Any arbitrarily-ordered input features are first fed into a permutation layer and then into fully-connected layers. The loss function is a simple binary cross-entropy.



Results

From a naive counting of the correctly predicted zeroes and ones of the output, the network produces a validation accuracy of 97%. Of course, because of the extremely large dataset and its dirtiness, this number may not be representative of the change detection algorithm's performance. Additionally, some of the cluster information may actually give away the change addresses without having use in practice; this happens when the change addresses are used in future transactions. Unfortunately, because of the massive amounts of data involved, the database can only record the state of the transaction graph at a single, recent point in time. However, the provider of the dataset, Blockseer, may be able to attempt to use the model for future predictions, which could demonstrate its (in)efficacy.

Work to be Completed

Because most of the clustering so far is still produced by human-designed heuristics (and indeed, the change detection model relies on some heuristic clustering to start with – to improve that clustering), the clustering is time-consuming and inflexible. I plan to use GraphSAGE, an inductive representation learning algorithm, on a subset of the Bitcoin transaction graph and use some sort of autoencoder to produce embeddings that I can use to directly cluster transaction nodes in the graph, obviating the need for change detection and other human-generated heuristics.