Motifs and Community Detection in the Venmo Transaction Graph

Aarush Selvan
Symbolic Systems
Stanford University
aselvan@stanford.edu

Dhruv Kedia
Computer Science
Stanford University
dkedia@stanford.edu

Rohan Sampath
Computer Science
Stanford University
rsampath@stanford.edu

December 11, 2019

1 Introduction

Mobile phones and innovation in financial technology have transformed the manner in which people now make payments. Mobile payments are becoming a popular form of payment and are seen as acceptable substitutes to cash, credit cards, and bank cheques. In the past couple of years, we have seen the rise of Venmo, Paypal, Zelle, ApplePay, Google Wallet, Square Cash, Visa Checkout, Stripe and several other mobile payment providers in the United States. If you look at Asian economies, especially China, mobile payments have exploded. China has leapfrogged from being a cash based economy to a mobile payments based economy, skipping the credit card phase all together. In China, people use mobile payments to pay for restaurants, street food, shops, train tickets, utility bills, and even to pay amongst one another. However, this scale of mobile payments usage has not been seen in the United States - yet. All forecasts point towards a rapid expansion in mobile payments in the United States over the next couple of decades as well.

Therefore, for this project, we examine the payment graph for one such mobile payment service that is popular in the United States - Venmo. Venmo is primarily a peer to peer payment platform, but also supports peer to vendor payments. Venmo currently has about 40million users and saw $24 billion in transaction volume in the quarter ending September 30, 2019. Venmo has made it extremely convenient for people to transfer money and split bills with one another. Often when friends dine with each other or apartment mates pay rent, they use Venmo to split and collect the large payment amongst them. Beyond the convenience, what makes Venmo attractive to users is its social component. Friends can caption transactions, like transactions, and comment on transactions. Our goal for this project is to identify group payment characteristics using Venmo payment transaction graphs. This is a particularly interesting problem because it will help us identify social relations in this financial network. You often engage in financial activities with those you share some social relationship with. For example you tend to go for dinners with friends, movie dates with your partner, happy hour with your colleagues, etc. Beyond this, it is also an interesting to problem to analyze how people are using the Venmo, and generate insights as to how one can increase the usage of the platform. Specifically, for this project, we hope to achieve 3 principal tasks:

- Examine basic graph characteristics to learn about the structure of Venmo transaction networks and how the platform is being used.
- Understand how people are using Venmo. We plan on identifying if there exists specific payment patterns within the graph or in other words we will be looking at the existence of different kinds of motifs, including star shaped motifs and temporal motifs.
- Identify the the communities and clusters that are formed in the Venmo payment network, particularly using the Louvain and related clustering algorithms. This will help us understand "who" is using Venmo. For example, we expect there to be distinct communities such as students, niche users, businesses, apartment mates, etc.

2 Related Work

There has been significant previous work on network analysis and community detection in mobile payment graphs. A seminal paper in this respect is Zhang et al. [2017]; the key motivation of this paper is to study the role of social relationships in the adoption of Venmo mobile payments. Through their study, the authors have found that Venmo communities are very densely connected compared to other interaction networks. The paper identifies that there are two distinct types of communities - user-user transaction communities and user-business transaction communities. In order to identify these communities, the paper uses the Divisive Hierarchical Clustering algorithm that leverages the concept of modularity. Other papers, such as Traag et al. [2019], introduce community detection algorithms such as the Leiden algorithm to extend the Louvain algorithm.

A key component of analyzing graph structure is evaluating the presence of motifs. Throughout this class, we have mainly focused on learning how to analyze motifs in static graphs that describe relationships between objects (nodes) and links between the objects (edges). What is critical to understanding graph structure in payment networks, however, is the time at which each payment was done; such temporal networks can be represented by a series of timestamped, or temporal, edges. Paranjape et al. [2017] introduce the temporal network motif as an elementary unit of temporal networks and provide a general methodology for counting such motifs, including some fast counting algorithms. We build on this analysis to count temporal motifs for our Venmo transaction dataset.

Finally, given that a payment graph is a directed one, it is important to extend useful concepts from undirected to directed. Onnela et al. [2005] is an example of work in this domain; the authors introduce a series of concepts - the most major ones being subgraph intensity and subgraph coherence - to extend these useful concepts from unweighted networks to weighted networks. Relying upon their definition of subgraph intensity, the authors define the "total intensity" of a motif as the sum of the intensities of all subgraphs that constitute a particular motif.

3 Data

This data was collected as part of a data analysis project by Dan Salmon. Yanhong Wu, an employee at Visa Research directed us to this dataset on Github. Our dataset consists of $7,076,585$ Venmo transactions, scraped from publicly available data. [1] By default, all transactions in Venmo are public unless users choose to set their privacy settings to either (a) Friends only, or (b) Private. The data was scraped for the following date ranges: (a) July 2018 - October 2018, and (b) Jan 2019 - Feb 2019. Each transaction consists of the following relevant information - date of transaction, payer information, receiver information, caption of transaction, social activities information (comments, likes, and mentions), and device information. Within the payer and receiver information, we have access to their userId, user type, personal details, friends count, and friend status. Most notably, the data does **not** specify the amount of each transaction, since Venmo does not make this data public.

We represent this dataset as a directed multigraph network, with a node representing a user, and a directed edge from the payor to the payee. The graph is represented using the TNEANet built-in class in Snap.py. Thanks to the availability of 100+ GB of RAM on our new GCP virtual machines, we were able to build the full graph of 7M transactions available.

3.1 Summary Statistics and Degree Distribution

Table 1 presents a few summary statistics for the transaction graph. The average degree per node is just 1.95, which demonstrates that this is a relatively sparse graph. The tables and figures that follow plot the degree distribution of the graph.

[1] Source: https://github.com/sa7mon/venmo-data

Statistic	Value
Number of users (nodes)	$7,178,381$
Number of transactions (edges)	$7,024,837$
Average transactions/user (i.e., avg. degree)	1.9572
Max. total transactions (max total-degree)	359

Table 1: Summary statistics for the transaction sample sub-graph

		Out-degree	
		Zero	One or more
In-degree	Zero	0 (0.0%)	$2,875,588$ (40.1%)
	One or more	$2,562,031$ (35.7%)	$1,740,762$ (24.3%)

Table 2: Distribution of graph nodes by in-degree and out-degree.

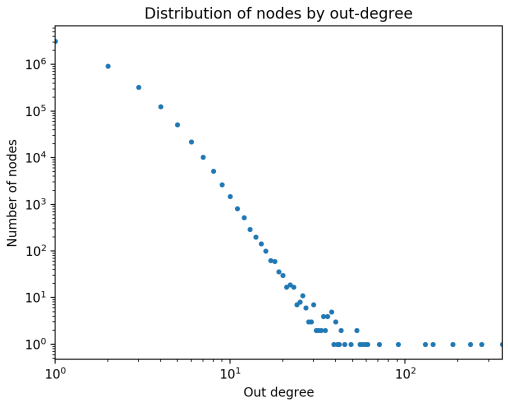

Figure 1: Distribution of nodes by out-degree. Nodes with zero out-degree are excluded.

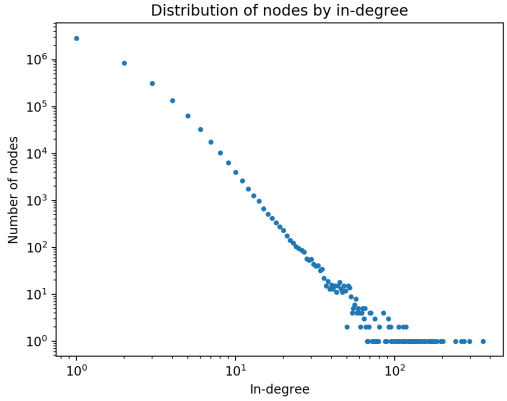

Figure 2: Distribution of nodes by in-degree. Nodes with zero out-degree are excluded.

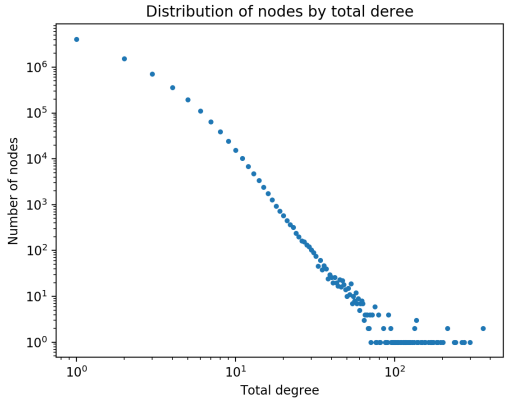

Figure 3: Distribution of nodes by total degree. Nodes with zero out-degree are excluded.

4 Technical Approach

4.1 Clustering Coefficients

We calculate the clustering coefficient of the Venmo transaction sample sub-graph. The global clustering coefficient C for the graph with set of nodes V is defined as follows:

$$C = \frac{1}{|V|} \sum_{i \in V} C_i \tag{1}$$

where C_i, the local clustering coefficient for each node $i \in V$, is defined as follows:

$$C_i = \begin{cases} \frac{2|e_i|}{k_i(k_i-1)} & k_i \geq 2 \\ 0 & k_i < 2 \end{cases} \quad (2)$$

where k_i is the degree of node v_i and e_i is the number of edges between the neighbors of v_i.

In addition, we compare this clustering coefficient with two other randomly generated directed multigraphs with the same number of nodes and edges as the Venmo transaction sample sub-graph (i.e., $1,896,974$ nodes and $1,191,575$ edges):

(i) **Random Erdos-Renyi graph**: edges in the random Erdos-Renyi graph are sampled *with* replacement, since the Venmo transaction dataset that we are comparing with is a directed multigraph.
(ii) **"Pair+" graph**: this recreates a graph that is, in principle similar to the Venmo transaction graph. . Therefore, the 'Pair+' graph is created as follows: pair-up all nodes with an edge between them (i.e., $N/2$ "pair" edges), (ii) distribute the other edges randomly (i.e., the other $E - N/2$ edges are generated randomly between nodes, with replacement).

The results of this comparison are presented in Table 3.

4.2 Motifs

4.2.1 Star-shaped Motifs

Next we tried to look at prevalence of group payments. One of the use cases of Venmo is that it allows a person to pay for a bill and then charge individuals in the group their share (e.g., when going out for brunch, one person may pay the bill and then everyone else would pay that person for their share of the meal). In our graph, this would appear as star-shaped motifs, with the number of vertices indicating the size of the group, and the direction of the edges going from the vertices to the central node. We wanted to understand how frequent these star-motifs were according to the size of the start (i.e., how often group payments happen given the size of the group)

We used a slightly different algorithm to count the number of these star-shaped motifs:

Algorithm 1 count N-pointed star motifs

```
Initialize N
Initialize Counter
for Nodes in Graph do
    Initialize empty set S
    for Neighbors in Node.GetInEdges() do
        Start node = Node.GetId()
        End node = Neighbor
        S.add(Start Node, End Node)
    end for
    if S.len() == N then
        Counter +=1
    end if
end for
Return Counter
```

4.2.2 Temporal Motifs

We adapt our definition of temporal motifs from Paranjape et al. [2017] as follows:

A k-node, l-edge, δ-temporal motif is a sequence of l edges, $M = (u_1, v_1, t_1), (u_2, v_2, t_2)..., (u_l, v_l, t_l)$ that are time- ordered within a δ duration, i.e., $t_1 < t_2... < t_l$ and $t_l - t_l \leq \delta$, such that the induced static graph from the edges is connected and has k nodes.

For example, in figure 4, we see the definition of temporal motifs for $\delta = 10$ seconds. The crossed-out patterns are not instances of motif M because either the edge sequence is out of order (for the first pattern crossed out) or the edges do not all occur within the time window $\delta = 10$ sec (for the second pattern crossed out).

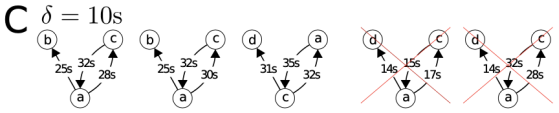

Figure 4: Examples of temporal motifs

We define δ as number of days for our analysis.

Since our graph is relatively sparse, we have few temporal motifs for $k \geq 3$. Therefore, we define temporal motifs for $k = 2$ (i.e., two users), and various values of l and δ, where δ is defined as the number of days from the first to the last transaction in the motif.

4.3 Community Detection Algorithms

As mentioned a critical part of the project is identifying the communities and clusters in the graph. This will help us understand "who" is using Venmo and understand the structure of the network. We expect there to be distinct communities such as students, niche users, businesses, apartment mates. In order to identify these communities, we will be implementing the Leiden Community Detection Algorithm.

4.3.1 Leiden Algorithm Traag et al. [2019]

Initially for our milestone we experimented with the Louvain algorithm Traag [2015] but that lead to badly connected communities that did not capture the structure of the graph or make intuitive sense. Based on online literature we decided to experiment with the Leiden Algorithm that yield communities that are better connected.

The Leiden algorithm is also an iterative algorithm, that converges to a partition in which all subsets of all communities are locally optimally assigned. The algorithm is also a simple, efficient, and scalable algorithm for identifying communities in large networks. The algorithm takes advantage of speeding up the local moving of nodes by moving nodes to random neighbours.

It is also a greedy optimization algorithm with the goal of optimizing the modularity of a partition of the network. Modularity is a measure that provides a numerical value to the quality of an assignment of nodes to particular communities. It evaluates how much more densely nodes are connected within a community versus how they are connected in a random network.

The mathematical equation for modularity (Q) is as follows: $Q = \frac{1}{2m} \sum_i \sum_j \left[A_{ij} - \frac{d_i d_j}{2m} \right] I_{y_i = y_j}$ where

A_{ij} is the edge weight between nodes i and j
d_i and d_j are sum of weights of edges attached to nodes i and j
$2m$ is the sum of all edge weights in the graph y_i and y_j are the communities of the nodes i and j

The detailed description of the Leiden algorithm Traag et al. [2019] (adapted from this paper) is as follows:

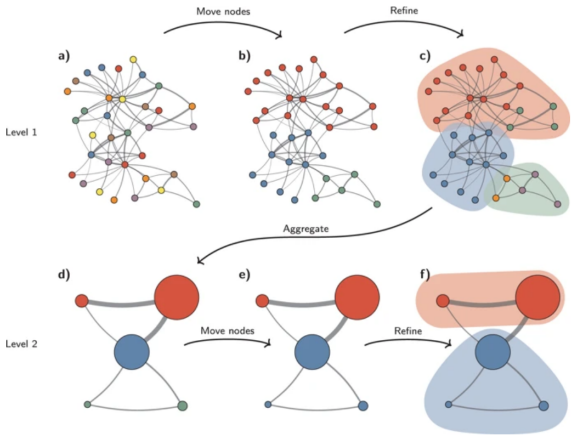

Figure 5: Leiden Algorithm visuals

The Leiden algorithm starts from a singleton partition in step A. The algorithm moves nodes from one community to another to find a partition in step B, which is then refined in step C. An aggregate network in step D is created based on the refined partition, using the non-refined partition to create an initial partition for the aggregate network. For example, the red community in step B is refined into two subcommunities in step C, which after aggregation become two separate nodes in step D, both belonging to the same community. The algorithm then moves individual nodes in the aggregate network in step E. In this case, refinement does not change the partition in step F. These steps are repeated until no further improvements can be made.

5 Results and Findings

5.1 Clustering Coefficients

The results for the clustering coefficient comparisons are presented in Table 1.

The clustering coefficients for both the Random Erdos-Renyi graph and the "Pair+" graph are close to zero, and

Graph	Cluster coefficient score
Venmo transaction sample sub-graph	1.2862×10^{-2}
Erdos-Renyi graph	5.2716×10^{-7}
"Pair+" graph	5.2716×10^{-7}

Table 3: Comparison of clustering coefficients between the Venmo transaction graph and the other randomly generated graphs.

orders of magnitude lower than the Venmo transaction graph. This makes intuitive sense. As shown in the summary statistics above, the average degree for nodes is 2.0, and 81.8% of nodes have a degree of 1. For all nodes with degree 1, the local clustering coefficient is zero, as shown in equation (2) above. Further, for each node i with degree greater than 1, you must have that there is one or more edges between the neighbors of node i - which, given the small number of edges, is highly unlikely.

On the other hand, because friends of friends often happen to be friends with each other, a real world transaction graph based on social connection, such as Venmo, is expected to have a positive clustering coefficient. More specifically, for every node i in the transaction graph with degree > 1, there is a non-trivial possibility that the neighbors of i have an edge between them too.

5.2 Motifs

5.2.1 Star-shaped Motifs

The count of star-shaped motifs we found in the dataset is presented in Table 4. As we can see, there is a large concentration of star motifs with 3 nodes and 4 nodes, but a sharp drop off on 5 and 6 nodes. This would make sense - common real-life applications of 3 and 4 node star-shaped motifs would be dinners or excursions consisting of 3 and 4 guests respectively. In this case, one person often pays and charges the others (hence a star-shaped motif).

Nodes	Count
S_3	$241,811$
S_4	$89,827$
S_5	$37,342$
S_6	$16,996$

Table 4: Count of star-shaped motifs

5.2.2 Temporal Motifs

The count of motifs for $k = 2$ (i.e., two users), and various values of l (i.e., number of edges - number of payments) and δ, where δ is defined as the number of days from the first to the last transaction in the motif. An interesting observation is that there are very few counts of motifs with $\delta > 1$, which seems to suggest that most payments between two users happen within a single day. The counts of motifs are presented below in 5.

l	δ = **0**	δ = **1**	δ = **2+**
l = 2	$134,189$	$12,790$	205
l = 3	$18,907$	$1,243$	74
l = 4+	$8,784$	$1,192$	24

Table 5: Count of temporal motifs

5.3 Community Detection

In this networks of 7,178,381 nodes we find that there are 1,667,620 communities. Below is a distribution of the number of nodes that belong in each of the different communities.

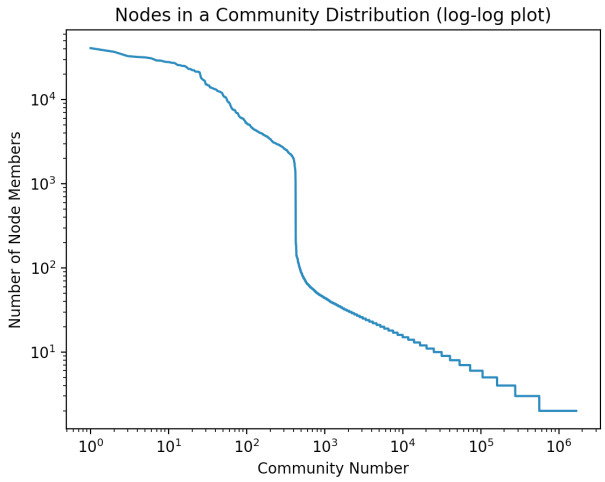

Figure 6: Number of Nodes in Different Communities

We are guaranteed that there are no completely disconnected nodes because of the nature of a payment transaction graph. Each node included in the graph must have at least 1 edge, since it is either the sender or receiver and hence the smallest community size is 2. We can see that at the tail end of the above plot the graph plateau's out. Since there are about 7million nodes, from the above plot we can see that the majority of them are of community size less than 10 (in the order of hundreds of thousands). This confirmed our initial intuition that majority of the communities we will find are extremely small. This is because the Venmo payment transaction network is fairly sparse. This is because it only includes data from accounts that are public. Further we also expect this small community number to include niche users and infrequent Venmo users. Niche users are those who use Venmo amongst their small circle, for example a group of individuals that play Poker.

Next, we noticed that in the order of thousands, there are communities of size 100 nodes. These are larger communities that are indicative of communities such as college friends, work friends, friends who play sports together. These are communities we interact with a lot in our daily lives. It is expected that these communities are in the order of 100 nodes because we are only involved in financial transactions with those that are 2 hops away from us.

Lastly, we noticed there are very few communities (less than 100) that are extremely large in the order of thousands. This is indicative of small businesses, student club organizations at colleges, charity organizations, and rotary clubs that use Venmo to facilitate their daily running. It is expected that these are fewer in number due to sheer reasons behind scale. (We expect there to be fewer organizations than friend groups in reality).

6 Future Work

We were fundamentally limited by the sparseness of our dataset (average node degree of only 1.95), which is largely a consequence of the fact that publicly available Venmo transactions are only a small subset of all Venmo transactions. Future work would therefore include procuring a more complete Venmo dataset (or other payment transaction dataset), although this may be challenging due to data privacy issues. Once this is done, future work would include: (a) analysis of more granular temporal networks (i.e., a more granular time scale than number of days), (b) application of other community detection algorithms (e.g., spectral clustering), (c) identifying motifs of more complex nature.

References

Xinyi Zhang, Shiliang Tang, Yun Zhao, Gang Wang, Haitao Zheng, and Ben Y Zhao. Cold hard e-cash: Friends and vendors in the venmo digital payments system. 2017.

Vincent A Traag, Ludo Waltman, and Nees Jan van Eck. From louvain to leiden: guaranteeing well-connected communities. *Scientific reports*, 9, 2019.

Ashwin Paranjape, Austin R Benson, and Jure Leskovec. Motifs in temporal networks. pages 601–610, 2017.

Jukka-Pekka Onnela, Jari Saramäki, János Kertész, and Kimmo Kaski. Intensity and coherence of motifs in weighted complex networks. *Physical Review E*, 71 (6):065103, 2005.

Vincent A Traag. Faster unfolding of communities: Speeding up the louvain algorithm. *Physical Review E*, 92(3):032801, 2015.