STOCK RECOMMENDATIONS USING INFORMATION CONNECTIONS FROM FINANCIAL BIPARTITE GRAPH

Rick Melucci
rmelucci@stanford.edu

Yuichi Kajiura
yuichik@stanford.edu

Jiushuang Guo
jguo18@stanford.edu

ABSTRACT

Finding how to make a consistent profit in stocks has been a long-standing theme for all investors. There are groups of successful investors who focus on finding secure stocks that are undervalued compared to their intrinsic price: those investors are known as "value investors". In this study we propose an approach for individual investor to easily follow the value investors' strategies based on the network analysis and machine learning on the bipartite investor-stock network constructed from a dataset of 13-F filings: quarterly investment information collected by SEC from institutional investors. We convert the bipartite graph into a directed investor-investor network, apply several analyses relating to Motif, PageRank and HITs, and detect secure investor communities by Louvain algorithm. We also apply JODIE, a temporal graph neural network model, on the original bipartite investor-stock graph to predict future investments of those investor communities. We then use these predictions to create a portfolio recommendation for individual investors.

1 Introduction

As the world becomes more volatile, uncertain and complex, managing personal assets is a great concern for individuals with little financial knowledge or little time.

On one hand, there has been rising interest among professional investors to apply machine learning for stock trading, usually in combination with high-frequency-trading. For such investors, the first movers with the fastest machines or fastest algorithms typically enjoy the profit. This requires a never ending cycle of development; as the popularity of a winning investment pattern performed by a machine grows, the market dynamics change as more people begin to follow that investing pattern. Consequently, the pattern quickly becomes obsolete. Due to this complex environment, it is difficult for an individual investor to navigate their way through a sea of large investors that each greatly influence the market.

On the other hand, there are groups of investors that focus on finding secure stocks by inspecting companies and finding ones that are undervalued compared to their intrinsic value (price): these investors, like Warren Buffett for example, are known as *value investors*. Value investing is a strategy that individuals can follow, and many who have enough financial knowledge to do so actually use this technique. It's a technique that would be useful for individuals who want to handle their investments less often, as they likely do not have the need to double or triple their assets quickly with high risk; they'd rather enjoy a secure method of positively profiting from their investments in the long run. However, it is time-consuming to inspect each asset, and it is difficult to pin down how to evaluate a company's intrinsic value without some background in finance.

In order to tackle this problem environment, we start with network analysis and finally build an algorithm that creates portfolio suggestions based on value investors, tailored for individuals who will not be able to check back often. In the network analysis, we convert the investor-stock dataset into an investor-investor directed network using the time order of investment as direction of information flow between investment. We then identify the information communities of value investors and their role inside the communities, and extract the key financial factors that the members of each community use for valuation. Next, we move back to original investor-stock bipartite network and finally predict a suggested set of stocks that value investor communities are likely to invest in soon. We calculate this prediction by applying graph neural networks to the bipartite investor-stocks networks to capture structure, combined with temporal financial features of nodes.

2 Previous Work

Since the financial crisis of 2007-2008, many studies have been published focusing on demystifying different aspects of financial networks and their systemic risk. However, there are relatively few studies focusing on the analysis of institutional investor networks in stock market, and only a few studies use both machine learning and network analysis to identify information connection between investor or a portfolio of stock suggestions. Among those studies, we picked 3 related works which take different approaches to analyze investor and/or stock networks. [1] provides a method of centrality analysis on stock networks constructed based on correlation between stocks; [2] proposes a method for motif analysis on shareholder network; [3] applies node embeddings on investor-stock bipartite network in order to perform clustering and classification. None of those studies directly approaching our problem, but we start our work from extending those studies.

2.1 Correlation based recommendation

F. Pozzi, T. Di Matteo & T.Aste[1] propose an approach for choosing well-diversified portfolios by choosing stocks that are peripheral in financial networks. These financial networks used in the study are transformations of stocks' correlation matrix into simple network models, using methods such as Minimum Spanning Trees[4] and Planar Maximally Filtered Graphs[5]. The researchers concluded that investing in subsets of central highly connected stocks is characterized by greater risk and worse performance. To do so, they calculated two hybrid centrality measures per node based on Degree, Betweenness Centrality (BC), Eccentricity (E), Closeness (C) and Eigenvector Centrality (EC). Then they showed that stock portfolios with the lowest centrality measures have good 'signal-to-noise ratio', also known as 'information ratio'[6] as a proxy for good performance of a portfolio. This signal-to-noise ratio is defined as

$$\frac{\bar{r}(\tau)}{s(\tau)} \tag{1}$$

where $\bar{r}(\tau)$ is the average and $s(\tau)$ is the standard deviation of the returns over 7071 investments. As they admitted that correlation between stocks is evolving over time and changes markedly during crisis, there is no reliable reason that the price correlation from the past will manifest itself again in the future. On the other hand, our method capture similarity of stocks based on network information as well as static financial features, such as price to earnings ratio and price to book ratio[7] so that it can learn consistent and intrinsic representation of stocks.

2.2 Motif based network analysis

Q. Guan et al.[2] provides a motif-based analysis on an investor network of China's energy stock market. They examine the differences in information connections of three types of investors - companies, funds and individuals, and analyze how these connection evolved over the 10 years time frame. To do so, they construct an undirected investors' network based on co-sharing information, and count the occurrences of 3-motifs with 2 type of edges (weak and strong). In their results, Q.Guan et al. found a rise in connections across different types of investors, as well as the consistent existence of homophily among individuals. Because they ignored the time-order of investments and constructed undirected graphs, they scrapped the possibility of determining temporal information relationships among investors, such as who is a leader and who is a follower. Our model instead utilize information regarding investment period to construct directed network, which allow as further analysis among investor relationships.

2.3 Node embedding based method

Zhang et al.[3] study the problem of node embedding in bipartite graph with dynamic and attributed edges. They propose the method called IGE(Interaction Graph Embedding). This method uses 3 different embeddings: for source nodes, target nodes, and edge attributes, constructs 3 neural networks, and combines them to represent probability that an edge exists given a set of a source node and its edges. Zhang et al. applied IGE on 4 real world datasets of investor-stock data which include timestamp, share price, and amount as edge attributes, to perform clustering and classification. The model has some drawbacks such that it did not use any structural information, and that they only use time stamp as a proximity measure of two nodes so time order is not used in the model. We incorporate both network and time-order information into our model so that it achieves better results.

3 Dataset

Institutional investors are required by law to disclose their investments every quarter in 13F forms filed with the SEC, and such information is collected by a third party. We use the dataset sourced from 13F filing aggregated by Sharadar on the Quandl platform[8] as our main dataset. This contains 6 years of filings from Jan. 2013. There are 23M reports from 6,454 investors who hold 13,499 stocks in total, where each report consisted of a ticker, investor name, reporting date (quarterly), value, units and price. 2.8M unique (investor, stock) pairs exist. By eliminating 0.8M (investor, stock) pairs reported in 2013Q1 for which we don't know when the investment initially happened, there are 2.0M new pairs from 2013Q2 to 2019Q2. This data can be naturally represented as a bipartite graph with edges between investor nodes and stock nodes.

To supplement the main dataset above, we also utilize fundamental stock data from same publisher[8] in order to construct features for stock nodes for link prediction. This consists of 150 fundamental indicators and financial ratios of 14,000 companies from 1997. We selected several features which particularly relate to value investors.

We also prepared a name list of 74 value investors, where 41 are sourced from web[9-14] and 33 are extracted from the main dataset by filtering the investor names who contain "value" in their name then later confirmed that they actually do value investments through their website. This list is for validation purposes on our community detection or other network analysis.

4 Methodology

4.1 Network analysis on directed investor-investor graph

4.1.1 Graph construction

We start by modeling the data as a bipartite investor-stock graph. From this bipartite graph, we then create a directed investor graph using time order of the investment. For example, if an investor i invested in a stock s in quarter t and then an another investor j invested to the same stock s in the next quarter $t+1$, we create a directed edge from i to j. The weight of an edge (i, j) is

$$w_{d,\{i,j\}} = \sum_{t=1}^{T} \frac{E_{ij}^{t}}{(n_i^{t-1}+1)(n_j^{t}+1)} \tag{2}$$

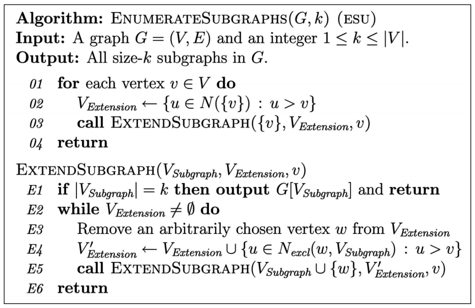
where n_i^{t-1} and n_j^t are number of total 'new investments' from investor i in quarter $t-1$ and j in t respectively, E_{ij}^t is number of stocks which were invested by i at $t-1$ and then by j at t, and T is a set of time period. Note that 'new investment' here means new edge (i, j) appeared at quarter t. $(\cdot + 1)$ is for smoothing purpose. We calculate the weight this way because investor j and i sometimes hold the same investments just because j and/or i do invest many assets. If this is the case, the information connection between j and i should not be strong. Therefore, we introduced the weight in order to penalize the number of investments.

Because each investor node is densely connected, we filtered out less important edges after the weight calculation by (1) setting minimum threshold of weight 0.1 and (2) choosing the 15 in-degree edges with highest weights for each node. (This also aligns with our intuition, since it is not reasonable that each investor directly connects to other several thousands of investors. Rather, an investor will typically refer to at most 10-20 other investors who have similar investment strategies.)

4.1.2 Motif analysis

Motif is a subgraph in a large network that reflects the functional property of the network. Different motifs represent different connection patterns and interactions between the vertices. Therefore, motif analysis is important on our constructed investor network, since it can help us understand how investors interact with each other in stock investment. For example, we can verify whether the co-holding and following behavior (where some investors follow other investors' investment behavior) exist in our investor network.

In 2.2, Q. Guan et al.[2] examines relation between three types of investors by using motif analysis on an undirected investor network. Since we don't have investor types in our dataset, we instead focused on following behavior in the investor network. In our proposal, we critiqued that Q.Guan et al.[2] ignores the time relation in the investment, so we improved the analysis by using directed edges as stated in 4.1.1.; therefore, our motif analysis can better capture following behavior by including the investment time information as the direction of edge.



```
Algorithm: ENUMERATESUBGRAPHS$(G, k)$ (ESU)
Input: A graph $G = (V, E)$ and an integer $1 \leq k \leq |V|$.
Output: All size-$k$ subgraphs in $G$.

for each vertex $v \in V$ do
    $V_{Extension} \leftarrow \{u \in N(\{v\}) : u > v\}$
    call EXTENDSUBGRAPH$(\{v\}, V_{Extension}, v)$
return

EXTENDSUBGRAPH$(V_{Subgraph}, V_{Extension}, v)$
if $|V_{Subgraph}| = k$ then output $G[V_{Subgraph}]$ and return
while $V_{Extension} \neq \emptyset$ do
    Remove an arbitrarily chosen vertex $w$ from $V_{Extension}$
    $V'_{Extension} \leftarrow V_{Extension} \cup \{u \in N_{excl}(w, V_{Subgraph}) : u > v\}$
    call EXTENDSUBGRAPH$(V_{Subgraph} \cup \{w\}, V'_{Extension}, v)$
return
```

Figure 1: ESU Algorithm in Wernicke, Sebastian

In order to do motif analysis, we first conducted counting on occurrence of 3-nodes motifs by the Exact Subgraph Enumeration Algorithm(ESU) proposed in Wernicke, Sebastian[18]. The details of the ESU algorithm are stated in fig 1. Since our investor graph is densely connected, we are not able to do motif counting on the entire graph in feasible time. Also, we want to focus more on exploring the interactive investing behavior among value investors and other investors, so we only conducted motif counting on the 74 value investors stated in section 3.

After motif counting using ESU algorithm, we measured significance of motifs by z score test. We want to know if a certain motif is significant(either over-represented or under-represented) in our investor graph, compared with random graph with same in-degree and out-degree sequences. Configuration model is used to construct 20 different random graphs, and ESU is then conducted on these 20 graphs to approximate the mean and standard error of motifs counting on the random graph. Finally, a z score is calculated for each subgraph(motif) of type i, by the formula $Z_i = \frac{(N_i^{real} - \bar{N}_i^{rand})}{std(N_i^{rand})}$, where N_i^{real} is the count on the subgraph of type i in investor network, and N_i^{rand} is count on the subgraphs of type i in random network generated by configuration model. For an easier further comparison, network significance profile(normalized z score) is also computed by the formula $SP_i = \frac{Z_i}{\sqrt{\sum_j Z_j^2}}$.

4.1.3 HITs analysis

Since our newly constructed directed investor-investor graph should capture the cascading of buying, we performed HIT analysis on the investor-investor graph to investigate which investor nodes seem to have an "early-investor" role, where others followed their decisions. Such "early-investors" manifest themselves as hubs in our directed graph. On the other hand, node with high authority can be considered as a "follower".

The HITs algorithm calculates two scores, a hub score and an authority score. First, all nodes' hub and authority scores are initialized to 1. It performs a series of iterations, each one made up of two steps: first we update the authority score of each node to the sum of the hub scores of each node that points to it, then we update each node's hub score to the sum of the authority scores of the nodes it points to. In each iteration, the scores are scaled and normalized. After the process, a node with a high hub score points to many other authorities, and a node with a high authority score is linked to by many different hubs. To calculate hub and authority scores, we use SNAP's HITs function.

4.1.4 PageRank Analysis

We also calculate PageRank score to investigate the influence of each investor node in the investor-investor graph. The PageRank algorithm was first introduced by Larry Page in order to measure the importance of webpages, and can more generally measure the importance of nodes in a network. The algorithm models a random walk through the graph, with a random chance of teleportation, and it relies on the idea that nodes with in-links from high-scoring neighbors must be important. To perform the algorithm, we calculate the following equation until convergence:

$$r_j = \sum_{i \to j} \beta \frac{r_i}{d_i} + (1 - \beta) \frac{1}{N} \tag{3}$$

where j is the rank of the current node, $\sum_{i \to j}$ is the sum over all i's that have a link pointing to j, r_i is the rank of node i, d_i is the degree of node i, and β is a constant.

4.2 Community detection on undirected investor-investor graph

4.2.1 Graph construction

We constructed an undirected investor graph, where two investors share an edge if they invest to a same stock in a same quarter. Similarly to the directed investor graph on 4.1.1, we introduced weight of an edge (i, j) as

$$w_{u,\{i,j\}} = \sum_{t=1}^{T} \frac{E_{ij}^t}{(n_i^t + 1)(n_j^t + 1)} \tag{4}$$

After the weight calculation, we filtered out less important edges by setting minimum threshold of weight 0.1. The purpose to construct another graph from 4.1 is that (1) we do not need directed edge to perform community detection and (2) investors who invested in a same quarter potentially have stronger connection than investors who invested in a subsequent quarter.

4.2.2 Louvain Algorithm

We run the Louvain algorithm on undirected investor graph to detect value investor community from the network. The Louvain algorithm is a greedy algorithm proposed in Blondel, Vincent D., et al[19], using 2-steps strategy to maximizes modularity. The Louvain algorithm, which supports weighted hierarchical communities detection, is a good method for us, since our graph is weighted with potential hierarchical communities (low-level communities can be formed by investors who invest only on certain areas of stocks or investors who follow a single leading investor, and high-level communities can be consist of investors from smaller communities which share similar investing strategies). The first step of the Louvain algorithm is partitioning, which is initialized with each node in a different community, then moves each node to another community by maximizing the modularity delta ΔQ among all the potential moves to different community C. We define

$$\Delta Q = \Delta Q(i \to C) - \Delta Q(D \to i) \tag{5}$$

where

$$\Delta Q(i \to C) = [\frac{\sum_{in} + K_{i,in}}{2m} - (\frac{\sum_{tot} + K_i}{2m})^2] - [\frac{\sum_{in}}{2m} - (\frac{\sum_{tot}}{2m})^2 - (\frac{K_i}{2m})] \tag{6}$$

with $\sum_{in}$ the sum of link weights between nodes in C, $\sum_{tot}$ the sum of all link weights of nodes in C, $K_{i,in}$ the sum of link weights between node i and C, and K_i the sum of all link weights of node i; $\Delta Q(D \to i)$ is defined similarly. The second step is contracting communities into super-nodes. Then the algorithm will be run recurrently on the super-node network.

4.3 Link prediction in investor-stock bipartite graph

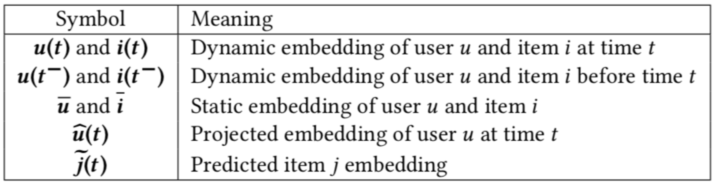
We would like to predict which stocks each investor, especially value investors will buy in the future; therefore, a link prediction algorithm for investor-stock bipartite graph is needed in our case. Most of the algorithms so far learn a static embedding for each user and item, which is problematic in our case. Since stocks can change rapidly and dramatic within a short time, and since investors might have different investing strategy at different time period, the embeddings for each stock and each investor are time dependent. Therefore, a temporal link prediction with dynamic embedding is desired for our task.

4.3.1 JODIE

JODIE [20], which was proposed this year 2019 by Kumar, Srijan, Xikun Zhang, and Jure Leskovec, is a dynamic embedding prediction trajectory algorithm, which can efficiently predict embedding of user/item at future timestamp, and use the predicted embedding to further forecast link/interaction between nodes.

JODIE trains two types of embeddings: a static embedding as well as a dynamic embedding for each user and each item(in our case, a user is an investor and an item is a stock ticker). The static embedding represents the long-term stationary characteristic of each user or item, while the dynamic embedding captures the temporal characteristic of each user or item. Two Recurrent Neural networks are used to train the embeddings: one for user embedding and one for item embedding. To update the embedding for each user at time t, denoted $u(t)$, the embedding i^-, item i's embedding after its previous interaction with any user, is used. The same update rule applies for item embedding by using u^-. This update rule is stated as formula:

$$u(t) = \sigma(W_1^u\, u(t^-) + W_2^u i^- + W_u^3 f + W_u^4 \Delta u)$$



Symbol	Meaning
$\boldsymbol{u(t)}$ and $\boldsymbol{i(t)}$	Dynamic embedding of user u and item i at time t
$\boldsymbol{u(t^-)}$ and $\boldsymbol{i(t^-)}$	Dynamic embedding of user u and item i before time t
$\bar{\boldsymbol{u}}$ and $\bar{\boldsymbol{i}}$	Static embedding of user u and item i
$\widehat{\boldsymbol{u}}\boldsymbol{(t)}$	Projected embedding of user u at time t
$\widetilde{\boldsymbol{j}}\boldsymbol{(t)}$	Predicted item j embedding

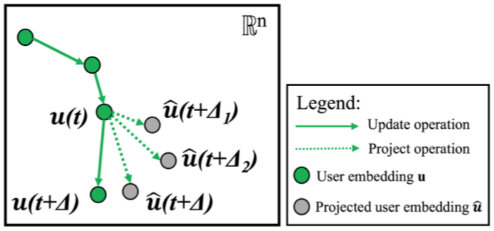


Figure 2: Projection Operation in JODIE [20]

$$i(t) = \sigma(W_1^i u(t^-) + W_2^i u^- + W_i^3 f + W_i^4 \Delta u)$$

where Δu is the time elapsed after u's last interaction with any item; Δi is the time elapsed after i's last interaction with any user; f denotes the interaction feature vector.

To predict future embedding, a projection operation is used. In JODIE, after a time interval Δ since previous interaction, a temporal attention layer is trained to project the embeddings of users, which will be used to predict which items this user will most likely to interact with. The fig 2 shows the projection operation process in JODIE: algorithm predicts the embedding for user u at different time in the future $t + \Delta_1$,$t + \Delta_2$,$t + \Delta$ (here $\Delta_1 < \Delta_2 < \Delta$) to get projected embedding as $\hat{u}(t + \Delta_1)$,$\hat{u}(t + \Delta_2)$,$\hat{u}_(t + \Delta)$. When the the interaction is observed at time $t + \Delta$, the embedding is updated again to $u(t + \Delta)$.

The JODIE algorithm uses the following loss function to train the model, in order to predict the item embedding that user most likely will interact with.

$$Loss = \sum_{(u,j,t,f)\in S} \|\widetilde{j}(t) - [\bar{j}, j(t^-)]_{concate}\|_2 + \lambda_U \|u(t) - u(t^-)\|_2 + \lambda_I \|i(t) - i(t^-)\|_2$$

where $\widetilde{j}(t)$ is the predicted embedding, and $\bar{j}, j(t^-)$ denoted the static embedding and the dynamic embedding immediately before the predicted time.

4.3.2 Data pre-processing and Feature selection

Before applying JODIE, we perform pre-processing of the raw data and performed feature selection. First, we eliminate reports from investors which did not increase its units of holdings from the prior quarter. This gives us 4M new/increased investments, considered as edges between investor and stocks. Next, we select several temporal financial indicators from the supplemental dataset as edge features and prune data which contains missing features, leaving approximately 3M edges in the graph. Then we calculate weights of each edge from investor i to stock s, defined as

$$w_b = \frac{\log V_{is}^t}{\sqrt{n_i^t \cdot n_s^t}} \tag{7}$$

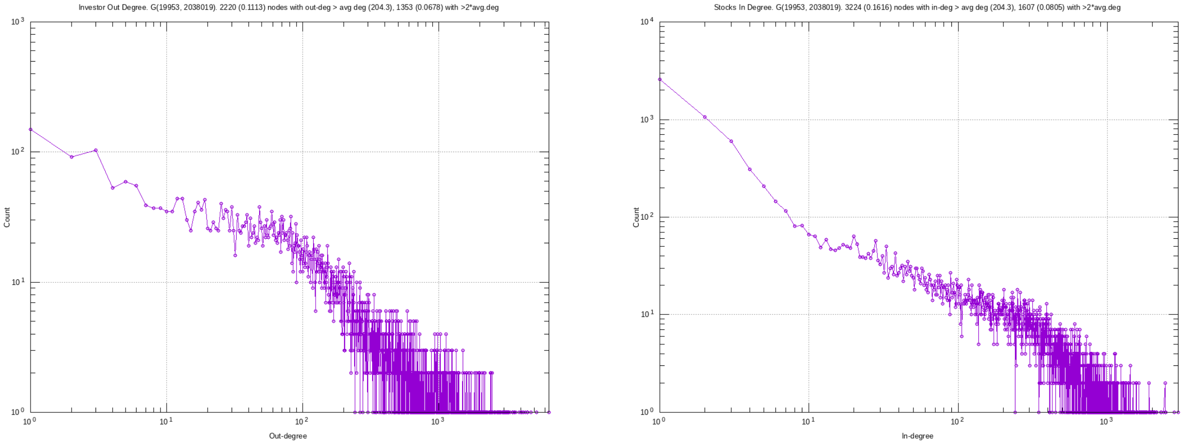
where n_i^t and n_s^t are number of total new/increased investments from investor i and to stock s in quarter t respectively, and V_{is}^t is the value of a new/increased investment from i to s in t. The purpose of this weight is to penalize investors who invest to every stocks, stocks which are invested by everyone, and investments with tiny amount. We prune all the data for which weight is less than the threshold of 0.2.

5 Results

5.1 Network analysis on directed investor-investor graph

5.1.1 Statistics of investor-stock bipartite network

After eliminating pairs reported on 2013Q1 and extracted new unique pairs, there are 7,744 investor nodes and 5,957 stock nodes, with 2,029,442 edges. Its weakly connected component size is 1.0 and approx full diameter is 5 (2.9 for 90% effective diameter). Figure 2 shows degree distribution of both investor and stock nodes.



(a) outDegree of investor nodes

(b) inDegree of stocks nodes

Figure 3: Degree distribution of investor-stock bipartite graph

5.1.2 Statistics of directed investor-investor network

After constructing the investor-investor network, we filtered out edges based on their weights by setting a threshold of 0.03 and then selecting top 15 in-edges for every node. This reduced the edge number from 16.6M to 75,893, with investor nodes of 5,705 (after eliminating zero-degree investor). There are 73 zero in-degree nodes and 1,115 zero out-degree nodes. Interestingly, there are only 1108 bidirectional edge and 1570 closed triangle, which implies one-directional flow of information. Its WCC size is 0.9996 and SCC size is 0.764, and the approximate diameter is 7 (3.44 for 90% effective diameter). The degree distribution of investors is shown in figure 3.

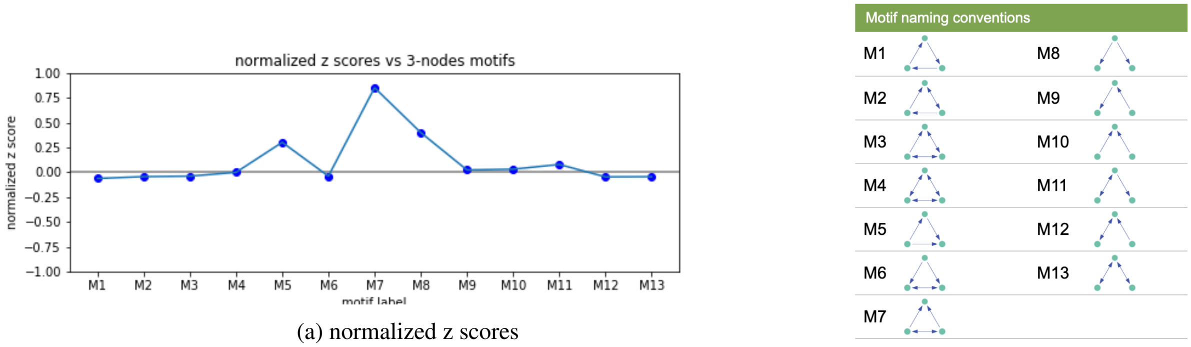
5.1.3 Motif analysis

We counted motifs by ESU algorithm on 13 types of 3-nodes subgraphs listed in fig 4b, and the corresponding result of normalized z score is showed in 4a. Z test concludes that subgraphs M5 and M7 are over-represented in our investor network.

M5 is a 3-nodes motif with two nodes point to the third node, while the two nodes also have one a way connection. M7 is a 3-node motif with two nodes point to each other, and both point to the third node. The two motifs both represent a following behavior: different investors follow one single investor, while these followers also share information of stocks. This result shows that there exist mutually connected communities who are following similar leading investor, and there are also some tail follower who follows such mutually connected communities. It verified our initial expectation on following behavior in stock investment. It also provides the evidence that many people do follow value investors' investment, which gives a theoretical reasoning on why we should further analyze value investors' investment strategies and forecast their future investment by link prediction.

5.1.4 HITs/PageRank analysis

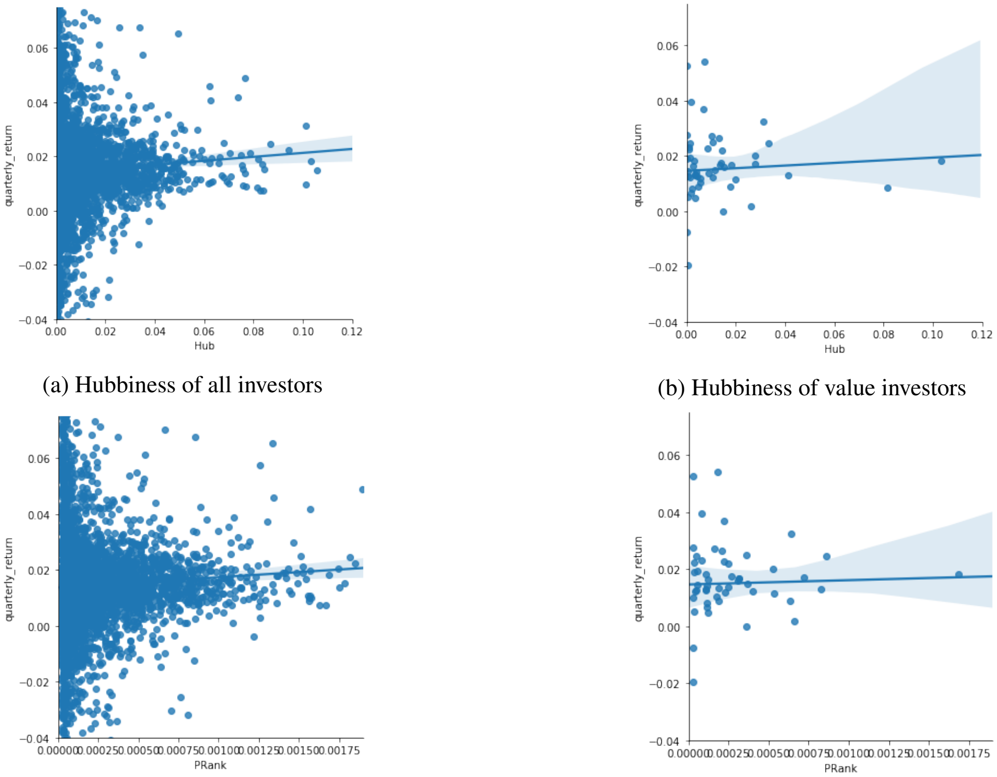
We calculated Hubbiness and PageRank of all investors and compared it with average quarterly return of investments. The results in Figure 5 show that there are slight positive correlations between hubbiness/PageRank and quarterly return, which imply an "early mover" makes more profit than "followers". However, the signals are not strong enough to use as a detector of profitable investors.



(a) normalized z scores

(b) 3 nodes motifs naming conventions

Figure 4: normalized z scores(on value investors' nodes) on all types of 3-nodes motifs



(a) Hubbiness of all investors

(b) Hubbiness of value investors

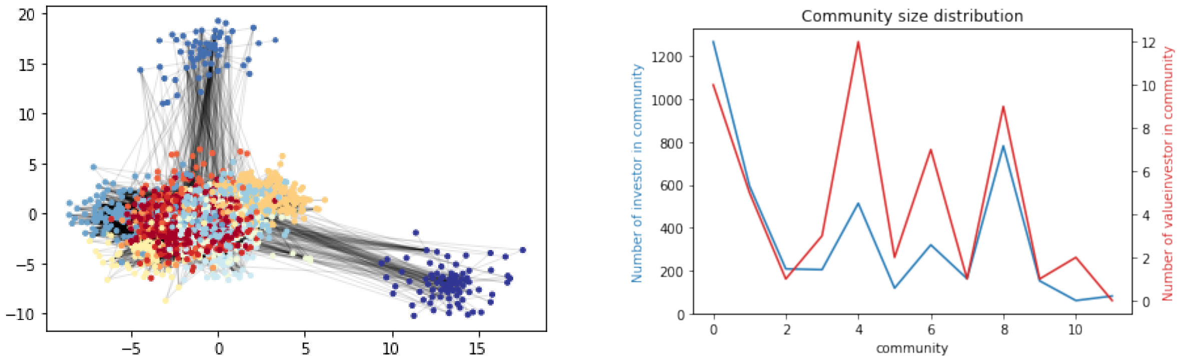
(c) PRanks of all investors

(d) PRanks of value investors

Figure 5: Hubbiness and PageRank versus average return on investments

5.2 Community Detection by Louvain Algorithm on undirected investor-investor graph

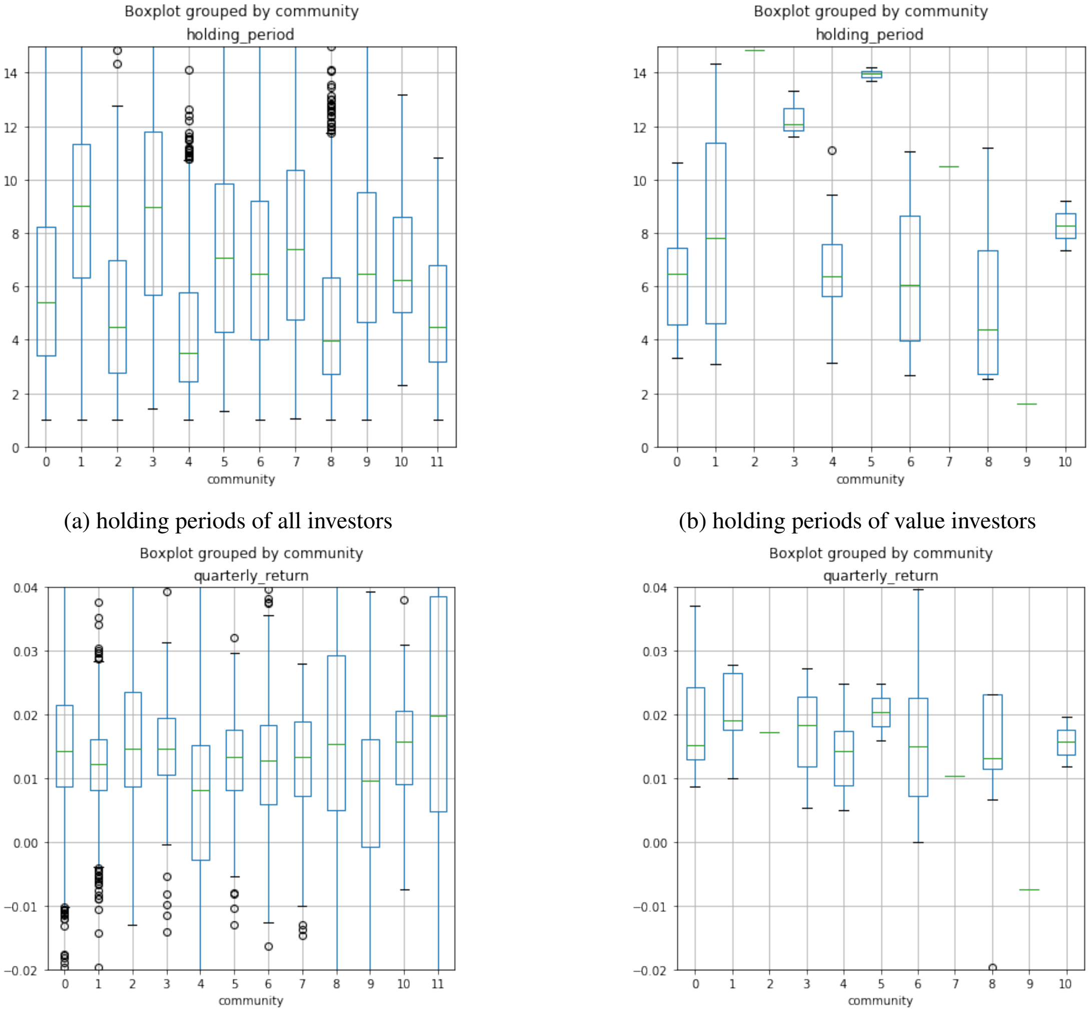
We performed Louvain Algorithm on the weighted and undirected investor graph. After pruning communities with less than 3 nodes, 12 communities were left as shown in Figure 6. Among those communities, community 3, 4, 5, 6 and 10 have relatively higher ratios of value investors than other communities. Next, we calculated average holding periods and average quarterly return on investments (ROI) for each investor in each communities as shown in Figure 7. The results show that value investors in community 3 and 5 play long term investments and enjoy moderate performance, implying potential investors that an individual investors should follow.



(a) 2D plot of communities

(b) Size of each community

Figure 6: Results of Louvain algorithm



(a) holding periods of all investors

(b) holding periods of value investors

(c) quarterly ROI of all investors

(d) quarterly ROI value investors

Figure 7: Holding periods / Return on investments(ROI) of each community

5.3 Link Prediction

5.3.1 Evaluation Methods

The evaluation methods that we used are the same as those used in the JODIE [20] paper - Mean Reciprocal Rank(MRR) and Recall@10. Mean Reciprocal Rank is measured by the formula $\text{MRR} = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{\text{rank}_i}, where \text{rank}_i$ denotes the rank position of the first relevant item for the i-th query. Recall@10 is the fraction of interactions in which ground truth item is ranked within the top 10 predicted items. Recall@10 = $\frac{\sum_i^{|predicted\,set|} \mathbb{1}\{GroundTruth \in top10\}}{|predicted\,set|}$, where $|predicted\,set|$ denotes the size of the predicted set.

5.3.2 Results for JODIE

We found that the naive implementation of JODIE (with no financial features) performed badly for our problem, with values of about 0.003-0.006 for both MRR and Recall@10. We believe this may be because JODIE treats all interactions as equally important. After pruning interactions with low weights, the performance increased an order of magnitude as shown in Table 1. We also performed JODIE with 6 features related to value investments: price earnings, price to earnings ratio, price to book ratio, current ratio, dividend yield, and payout ratio. Running JODIE with the features showed a 5% improvement in MRR and a 10% improvement in Recall@10 respectively. The result of 0.096 for Recall@10 may sounds relatively low compared to the results on other datasets used in original paper, but we think it is actually a quite good result considering that our dataset only consists of new/additional investments and much fewer repeated interactions between same investor-stock pairs (which makes prediction difficult). Note that we keep all the parameter setting as default, except embedding dimension which we chose the value 64. Other than that, we also tried modification of JODIE such as learning by weighted MSE to penalize investor/stock which appear everywhere or treating a quarter as a batch; however, those did not help to achieve higher performance, convincing us that the regular JODIE algorithm is suitable for our problem.

Architecture	Dataset Size(interactions)	Feature	MRR	Recall@10
JODIE	0.6million	No	0.040	0.087
JODIE	0.6million	Yes	0.042	0.096

Table 1: Link Prediction Result Table
MRR: Mean Reciprocal Rank

6 Conclusion

By analyzing motifs in the directed investor-investor graph we created, we identified that there are investors who connect to many other investors, potentially implying leaders to follow, and that those leaders can be interpret as nodes with high hubiness and PageRank but that those indicators have only slight correlation with investment performance. On the other hand, community detection on the undirected investor-investor graph we created shows clear differences among investor communities, especially in terms of holding periods. These results are useful for individual investors who do not have enough time to replace their portfolio often and prefer to follow long term investors, or value investors. We also found that graph neural networks, such as JODIE trained on preprocessed data with indicative financial features, have a power to predict future investments from professional investor's past holdings. Combining the link prediction from JODIE with the community detection above, we can predict future investments of any particular community. Performing prediction on long-term/value investor community which have consistent investment performance will help individual investors to greatly narrow down potential stocks to research before investing.

As a suggestion for future works, community detection will be improved by applying a method which allows overlapping of communities, such as AGM, since we noticed that investors are densely connected. Some investor also own several fund with different or mixed strategies. Those facts will create many overlaps of communities. Although naive HITs/PageRank analysis on whole dataset shows limited insights possibly because of heavily densed network, performing weighted PageRank on particular community detected by Louvain or other methods may also show interesting correlations with investment performance. Performance of JODIE could also be enhanced by either incorporating more features such as PageRank, HIT scores, and other financial indicators from the supplementary table, or by tuning parameters such as embedding dimension, since we only had a chance to test an embedding dimension of 64 due to time limitations.

References

[1] F. Pozzi, T. Di Matteo T. Aste. Spread of risk across financial markets; better to invest in the peripheries. In *Scientific Reports volume3, Article number: 1665*, 2013.

[2] Q. Guan, H. An, N. Liu, F. An & M. Jiang. Information Connections among Multiple Investors: Evolutionary Local Patterns Revealed by Motifs. In *Scientific Reports volume 7, Article number: 14034*, 2017.

[3] Y. Zhang, Y. Xiong, X. Kong, and Y. Zhu. Learning node embeddings in interaction graphs. In *CIKM*, 2017.

[4] Mantegna, R. N. Hierarchical structure in financial markets. In *European Physical Journal B 11, 193–197*, 1999.

[5] Tumminello, M., Aste, T., Di Matteo, T. Mantegna, R. N. A tool for filtering information in complex systems. In *Proceedings of the National Academy of Sciences 102/30, 10421–10426*, 2005.

[6] Thomas H. Goodwin. The Information Ratio. In *Financial Analysts Journal Vol. 54, No. 4, pp. 34-43*, 1998.

[7] Graham, B., Dodd, D. Security Analysis. In *McGraw-Hill, New York*, 1934.

[8] https://www.quandl.com/databases/SFA/data

[9] https://www.kiplinger.com/slideshow/investing/T041-S002-6-great-mutual-funds-for-value-investors/index.html

[10] https://www.investopedia.com/articles/investing/071415/five-wildly-successful-value-investors.asp

[11] https://mobile.reuters.com/article/amp/idUSKBN1W20E3

[12] https://www.wsj.com/articles/a-value-investor-defends-value-investing-despite-its-recent-track-record-11570414080

[13] https://einvestingforbeginners.com/deep-value-investing-quotes-ashul/

[14] https://www.vintagevalueinvesting.com/the-complete-list-of-q2-2019-hedge-fund-letters-to-investors/

[15] H. Dai, Y. Wang, R. Trivedi, and L. Song. Deep coevolutionary network: Embedding user and item features for recommendation. In *arXiv:1609.03675*, 2016.

[16] S. Kumar, X. Zhang, and J. Leskovec. Predicting dynamic embedding trajectory in temporal interaction networks. In *Proceedings of the 25th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2019.

[17] L. Lu and T. Zhou Link prediction in complex networks: A survey. In *Physica A: Statistical Mechanics and its Applications, 390(6):1150–1170*, 2011.

[18] Wernicke, Sebastian. Efficient detection of network motifs. In *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB) 3.4* , 2006

[19] Blondel, Vincent D., et al. Fast unfolding of communities in large networks In *Journal of statistical mechanics: theory and experiment* , 2008

[20] Kumar, Srijan, Xikun Zhang, and Jure Leskovec. Predicting dynamic embedding trajectory in temporal interaction networks. IN*Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery Data Mining. ACM* 2019.