A Hybrid Deep Learning Model for Dynamic Stock Movement Predictions based on Supply Chain Networks

John Rios
University of Iowa
john-riosrodriguez@uiowa.edu

Kang Zhao
University of Iowa
kang-zhao@uiowa.edu

Nick Street
University of Iowa
nick-street@uiowa.edu

Hu Tian
Chinese Academy of Sciences
tianhu2018@ia.ac.cn

Xiaolong Zheng
Chinese Academy of Sciences
xiaolong.zheng@ia.ac.cn

Abstract

An inter-firm relationship of paramount importance is captured by supply chain networks. Embedded in such a network, a firm's performance is associated with its partners' and peers' performance. This paper proposes an end-to-end predictive framework named Hybrid and Temporal Graph Neural Network (HT-GNN) to predict the dynamic stock price movement of firms. The model learns time-dependent node embeddings by aggregating network neighbors' features and market trends to provide node classifications over time. Experiments on a real-world supply chain network among over 2,700 publicly traded firms show that HT-GNN can improve dynamic stock movement predictions. We define different types of network neighborhoods by identifying firms that contribute to such predictions in different ways and going even beyond immediate ties. Our results would naturally help investors understand stock price movement and managers identify network neighbors with predictive power over its own stock price movement.

1 Introduction

For decades, predicting future stock returns has attracted investors' and researchers' interests. Previous research has approached this challenging problem by analyzing a variety of external information like social media (Luo et al., 2013; Nguyen et al., 2015; Rechenthin et al., 2013) and search logs (Agarwal et al., 2017; Luo et al., 2013), among others. An additional source of firm's information and probably the most important relationship between firms is its supply chain, which has been shown to be associated with the firm's performance (Bellamy et al., 2014; Feng et al., 2019; Merschmann & Thonemann, 2011; Seiler et al., 2020; Vanichchinchai & Igel, 2011). However, we found that there is no exiting literature on the association of stock returns for companies that are connected in the supply chain beyond immediate neighbors. Since these relationships are better modeled as a graph, a network approach, as opposed to a linear or hierarchical model, provides a unique perspective to depict the characteristics of a supply chain (Kim et al., 2011) and to identify relevant findings to make better managerial decisions. At the same time, such partnerships reflected in a supply chain network are more intuitive and interpretable to managers and investors than other inter-firm relationships that have been used to study stock prices, such as co-search in search engines and co-appearance in news.

An effective tool that is growing in popularity to produce network inferences is Graph Neural Networks (GNNs). GNNs are a set of models that adapt conventional deep learning methods to graphs in order to learn meaningful node representations (embeddings). The key contribution of these methods lies in learning how to aggregate information from the neighbors of a central node in a graph, since this is far more challenging task for graphs than for other types of data with more rigid structure like texts or images. These learned representations can then be articulated to perform traditional network inferences like node classification, link prediction or graph classification, among others.

Motivated by these findings, this paper **attempts to investigate if a GNN model can leverage signals from neighboring firms in a supply chain network to predict a focal firm's future stock price movement.** To answer this question, we propose an end-to-end supervised learning model named Hybrid and Temporal Graph Neural Network (HT-GNN). Our HT-GNN learns market and neighbor information aggregation for node classification over time. Before providing stock price predictions, the model also learns dynamic node embeddings for each firm, so that predictions become more explainable. Since these embeddings are learned in a supervised fashion, they incorporate information not only about the market and stock movement but also about the network structure. Finally, we propose different definitions of network neighborhood for our focal firms to predict future stock price movements. The results show that peer companies in neighborhoods defined beyond direct ties improve the prediction performance of stock price movement.

The remainder of this paper is organized as follows. Section 2 reviews exiting literature in the field relevant to this work, while Section 3 describes the data collection and network construction. In Section 4 we present the proposed D-GNN along with its results. Section 5 presents our conclusions and future work.

2 Literature Review

Networks (also referred as graphs) are collections of nodes and edges (i.e., links between nodes) that can be used to represent a variety of complex interconnected systems. We can identify three classical machine learning tasks in networks: community detection, node classification, and link prediction. In a general sense, communities can be understood as subgraphs, in which nodes are expected to be more connected with each other than with nodes outside the community. The link prediction problem is mainly focused on the identification of missing or unobserved information (links) and on the prediction of future connections in evolving networks based on the currently observed data. Similarly, the node classification problem focuses on predicting the labels on a semi-complete graph. This means that labels are known for a portion of the nodes and the goal is to predict the label of the unknown ones. Another version of this problem, and one of the goals for this paper, is about predicting node labels through time. In this setting the models learn relevant patterns from previous times and use it to predict node labels in the future.

A key challenge for network inference is that to perform these tasks, raw graph data needs to be transformed into structured data in order to implement traditional machine learning algorithms. GNNs are efficient techniques that automatically learn the representation of a node and are presented in Section 2.1. These representations can later be used to make inferences about the nodes or graphs. In Section 2.2, we introduce the concept of supply chain network, while in Section 2.3, we review how supply chains and additional information have been leveraged for stock price prediction.

2.1 Graph Neural Networks

With graphs being non-Euclidean domains, many traditional machine learning algorithms cannot be directly implemented on them. Among the main challenges associated with graph data, we find that graphs can have different number of unordered nodes and nodes can have different number of neighbors (Wu et al., 2020). To tackle these problems, GNNs learn node representations (embeddings) by aggregating the message passed from the node neighbors. These representations can then be used as inputs for tasks like node prediction, link prediction, or graph classification. Very complete reviews on GNNs can be found on Wu et al. (2020) and Zhou et al (2018).

One widely adopted type of GNNs is Convolutional GNNs (ConvGNNs) since they can also be used to construct other more complex types of GNNs. The unstructured nature of node neighborhoods makes it infeasible to apply regular deep learning convolution operations, so ConvGNNs try to generalize this concept from grid data to graph data. The key idea of a ConvGNN layer is to produce a representation of node u that aggregates u's own features along with the features of (message from) u's neighbors. ConvGNN layers can also be stacked to include nodes that are k hops away from the central node. ConvGNNs can be categorized in spectral-based and spatial-based. In spectral-based ConvGNNs, the convolution depends on the eigendecomposition of the graph Laplacian. Different versions of ConvGNNs have been proposed (Bruna et al., 2014; Defferrard et al., 2016) reducing the computational complexity and being Graph Convolutional Networks (Kipf & Welling, 2017) the most popular. However, these methods are trained based on a fixed graph structure, which means that they cannot be applied to new graphs with a different structure.

Spatial-based ConvGNNs define convolutions directly on the graph, operating locally on each node. Similar to conventional convolutional neural networks on an image, spatial-based approaches convolve the representation of node u with its neighbors' representations to derive the updated representation of node u. The convolution operators are jointly trained for all the central nodes sharing the same convolution parameters. This approach allows an inductive framework, which means that the model can be applied to nodes and graphs with different characteristics or even unseen ones. As mentioned before, the key challenge for the convolution operators is to handle neighborhoods of different sizes and to be invariant to permutations of the neighbors ordering. GraphSAGE (Hamilton et al., 2017) proposed a set of aggregator architectures to combine the neighbors' representations of a node and achieved improvement in performance with respect to previous GNN models. The representation of node u when stacking k GraphSAGE layers ($\boldsymbol{h}_u^{(k)}$) would be produced by the convolution in Equation 1

$$\boldsymbol{h}_u^{(k)} = \sigma\left(\boldsymbol{W}^{(\boldsymbol{k})}\left[\boldsymbol{h}_u^{(k-1)}, f(\boldsymbol{h}_n^{(k-1)}, \forall n \in S_{N_u})\right]\right) \tag{1}$$

where $[\cdot]$ means concatenation, $f(\cdot)$ is the aggregator architecture, $\boldsymbol{W}^{(\boldsymbol{k})}$ is a learnable parameter matrix, S_{Nu} is a random sample of the u's neighborhood (N_u), and $\sigma(\cdot)$ is an activation function. The initial representation of node i ($\boldsymbol{h}_i^{(0)}$) would be its input feature vector ($\boldsymbol{x}_i$). Randomly sampling N_u is another of the contributions of GraphSAGE. It allows GraphSAGE to be applied to large graphs and in context where the neighborhood size varies considerably, while still obtaining a high performance.

One of the basic but powerful aggregator architectures used in spatial-based ConvGNNs is mean. However, Velickovic et al. (2018) proposed an attention mechanism that learns a weight α_{uv}^{k} used to represent the importance of the neighbor v for the central node u. The authors also incorporated multihead attention to increase the model's ability to learn, showing an improvement over GraphSAGE.

2.2 Supply Chain Networks and Firm Performance

A network approach to analyze supply chains has provided insightful findings that other approaches would not identify. Based on this approach, researchers started to adopt network analysis tools to study supply chains. For example, Kim et al. (2011) related key social network analysis metrics to the three automotive supply chain networks reported in Choi & Hong (2002), showing how to apply social network analysis to investigate the structural characteristics of supply chain networks. In addition, Hearnshaw & Wilson (2013) mirrored the properties of complex network models with real-world supply chain networks. Previous literature has also studied disruptions in supply chain networks from a topological perspective (Zhao et al., 2011).

Not only firms but also their performance is linked in a supply chain network. The stock price of a firm is related to its own supply chain performance (Hendricks & Singhal, 2005) and to the performance of the firm's business partners (Rios et al., 2019). Cohen & Frazzini (2008) found that future monthly return of a focal company is correlated with lagged returns of its customers. Also, according to Menzly & Ozbas (2010), at the sector level, the lagged performance of a firm's supplier and customer sectors correlates with the future stock return of the focal firm. Later, Agarwal et al. (2017) measured the weekly online co-search attention that investors pay to a focal firm and its business partners and found that partners that have a low co-search attention explain future weekly stock returns of the focal company. Finally, additional performance metrics of a firm such as innovation, ROA, and asset turnover are also shown to be associated to its supply chain network (Bellamy et al., 2014; Seiler et al., 2020).

2.3 Stock Return Prediction

Predicting stock movements has led to the generation of numerous theories and factors associated with the Capital Asset Price Model (CAPM) (Lintner, 1965; Sharpe, 1964). While CAPM explains the relationship between systematic risk and expected return for stocks, anomalies in this model offer an opportunity to make money. The Fama-French model, an extension of CAPM, provides three factors (market, small minus big, and high minus low) from Fama & French (1996) and a fourth factor (momentum) from Carhart (1997), which explain the anomalies in the CAPM model. To show the predictive power of new features, these factors are usually included as control variables in regressions for cross-sectional stock return prediction. Other variables like average daily return or price in previous time periods are also included as control variables or baseline models for stock return prediction in previous literature (Goyal & Welch, 2003; Nguyen et al., 2015).

In addition to historical financial data, researchers have discovered that information outside a focal firm, such as social media and investor attention, can improve the prediction of stock returns. Luo et al. (2013) found that social media metrics (Web blogs and consumer ratings) were significant at explaining future equity values, while conventional online behavioral metrics (Google searches and Web traffic) have weaker effects. However, their model is explanatory, instead of predictive, in nature. Rechenthin et al. (2013) used supervised machine learning algorithms to predict future stock price direction and found slight predictability in the sentiment of user-generated posts about a stock in Yahoo Finance. Additionally, Nguyen et al. (2015) took a step further by including not only the overall sentiments of a stock in message boards, but also topic-sentiment features, which represent sentiments about specific topics of a company. Nevertheless, while social media and user-generated data represent a rich source of information for the prediction of stock prices as they are based on wisdom of the crowd, they are also more subject to manipulations (Rechenthin et al., 2013) compared to financial data of a focal firm and its neighbors. Also, such data often lacks interpretabilities (e.g., why do users talk about two companies together in a post?), which are important to convince stakeholders to adopt a predictive model.

Researchers have also employed GNN models in the stock prediction problem. Feng et al. (2019) proposed a temporal graph convolution to rank company stocks based on their expected return for the next day. The authors built a network by connecting companies in the same industry sector and company interdependencies obtained from Wikidata (Vrandečić & Krötzsch, 2014). However, even when they included some supply chain information, their connections are most likely not complete. Additionally, their model was only aggregating companies that were direct neighbors in the graphs, without including any information about the network structure or market trends. Signals from other firms beyond a firm's immediate neighborhood have not been explored for actual predictions of firm's performance.

3 Data Collection and Network Construction

Our data collection started with the firms in Standard and Poor (S&P) 1500 during 2019. Using these firms as seeds for a snowball sampling approach, we retrieved information of up to 2-hop neighbors (including both customers and suppliers) for each seed company. The supply chain information was collected from Mergent Horizon (http://www.mergentonline.com) in July 2019. This database provides companies' basic information (e.g., Ticker Symbol, Industry Sector, and Country), lists of suppliers, and lists of customers, among others. We started by retrieving information about customers and suppliers of each seed company. Then, we parsed the web page of each of the companies identified as suppliers (1-hop suppliers) of the seed companies in order to include their customers and suppliers (2-hop suppliers of the seed companies) into the analysis. Similarly, for each of the customers (1-hop customers) of the seed companies, we included its suppliers and customers (2-hop suppliers of the seed companies). Figure 1 presents the structure of the web scraping process for the supply chain network.

The resulting super network combines the supply chains of all the seed companies. The set V of nodes represents firms that are connected by edges indicating directed supply relationship. An edge $e_{i,j}$ means that v_i is a supplier of v_j. For each company in the dataset, we collected its stock prices and additional financial data for three years (January 1st, 2016 and December 31st, 2018). Such information was collected from both the Center for Research in Security Prices (CRSP). Then, we excluded firms whose stock data was unavailable, including private firms and government agencies. Our predictive models were built over Period 2 (last two years. January 2017 - December 2018), while Period 1 (first 12 months. January 2016 - December 2016) was used for parameter estimation. After excluding the companies with incomplete financial information, we obtained a supply chain network with 2,731 nodes (firms) and 14,194 edges. Figure 2 visualizes the network (a) and its degree distribution (b).

Figure 1. Structure of the complete web-scraping process for the network collection

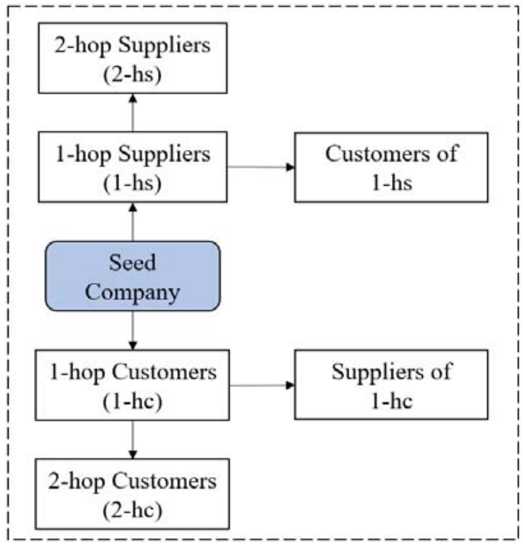
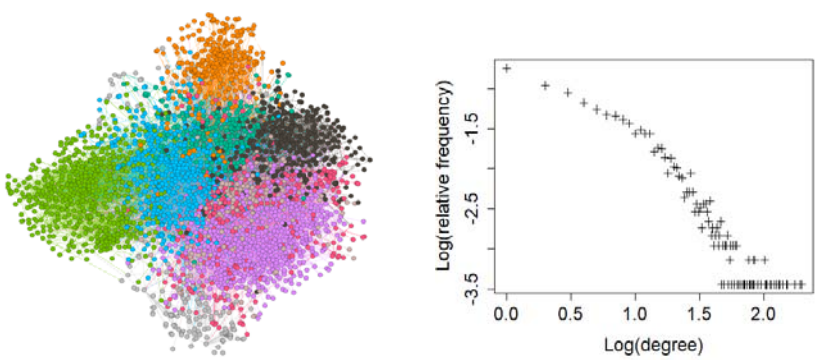


Figure 2. Supply chain network and its degree distribution. Node colors represent network communities



(a) Supply Chain Network (b) Degree Distribution

Table 1 presents a summary of network statistics including number of nodes, number of edges, average degree, diameter, and size of the largest connected component (LCC).

Table 1. Summary of basic network metrics.

	Number of Nodes	Number of Edges	Average Degree	Diameter	Size of LCC (%)
Supply Chain Network	2,731	14,194	10.39	17	100

For a second neighborhood definition, we identified communities in the supply chain network by maximizing modularity (Newman & Girvan, 2004). Correlated behaviors are well-known in a network context—connected nodes are similar to each other and tend to behave similarly (McPherson et al., 2001; Zhao et al., 2010). Based on this concept, we are using network communities as a second approach to identify similar peer companies that can provide relevant information about a focal firm for our predictive models.

4 Predictive Model

For predictive experiments, we identified a group of 463 *focal* companies. These are the group of companies that are in the S&P500 and have available information about their supply chain partners. Stock prediction can be considered as a dynamic node classification problem, where labels are assigned to each node over time (e.g., on daily or hourly bases). Therefore, traditional GNN models (Hamilton et al., 2017; Veličković et al., 2018) need to be adapted to incorporate characteristics of a node and of its neighbors that change over time to produce time-dependent predictions. In this context, previous research has applied GNNs to inter-firm networks to predict stock performance over time (Feng et al., 2019). However, unlike previous research, our approach includes market information and nodes (peer companies) beyond 1-hop neighbors in order to leverage information about the network structure of the supply chain network and improve prediction performance.

Relevant peer companies beyond direct ties are identified by network communities previously built using modularity maximization (Newman & Girvan, 2004). We propose a second neighborhood definition that includes the nodes (peer companies) in the same network community of the focal firm. In summary, our approach includes two neighborhood definitions: *i*) Direct Neighbors (N_1) and *ii*) Network Communities (N_2). Finally, we built a GNN model with these two neighborhood definitions to predict the stock price movement of our 463 focal firms in a daily basis.

4.1 Model Setup

Our model predicts stock price movement of the focal firms from one day to the next one. However, predicting whether a stock price will move up or down by a tiny amount (e.g., 0.001%) is extremely challenging and yet not very helpful for investors or stakeholders. For this reason, we built a three-class categorical target variable indicating if the price is going up, going down, or staying the same from day *t* to day *t+1*. For each firm, we first calculated the standard deviation (*sd*) of the daily stock returns of the firm during Period 1. Then, we used those estimates to discretize the observed numeric daily stock returns for this firm during Period 2 into the three classes. If the daily stock return of a company from day *t* to day *t+1* is within its ±0.25*sd*, that instance was labeled as *Class 0* (i.e., Neutral). This means that the stock price does not change much from day *t* to day *t+1*. By contrast, if the change in stock return is lower than -0.25*sd*, that instance was labeled as *Class -1*, which means that the stock price goes down from day *t* to day *t+1*. Similarly, if the stock return change is higher than 0.25*sd*, that instance was labeled as *Class 1*, which means that the stock price is going up. This discretization produced a balanced distribution of 0.325, 0.320, and 0.355 for *Class -1*, *Class 0*, and *Class 1*, respectively.

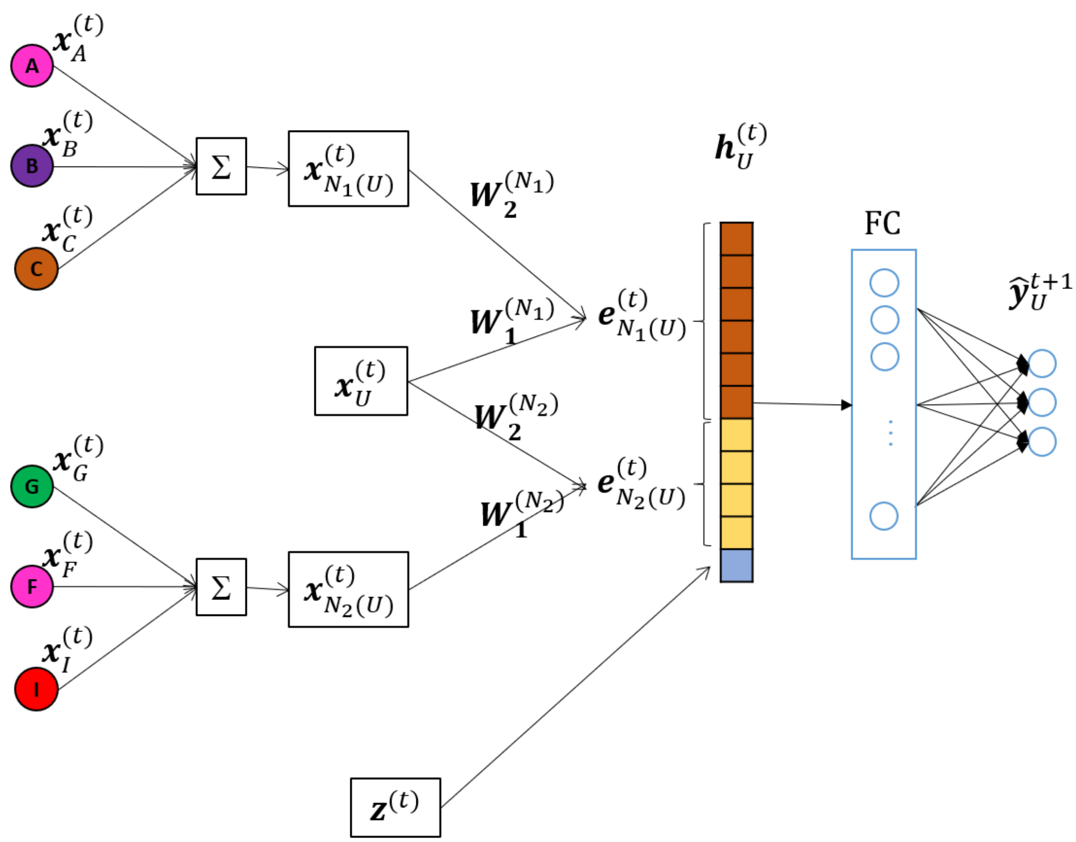
4.2 The Hybrid and Temporal GNN Model (HT-GNN)

In general, our three-class stock movement prediction for the firm U can be expressed as $\hat{\boldsymbol{y}}_U^{(t+1)} = f\left(\boldsymbol{x}_U^{(t)}\right)$. We want to learn a map function $f(\cdot)$ that receives a vector of input features for company U at time t ($\boldsymbol{x}_U^{(t)}$) and produces a likelihood vector for the potential price movements (up, down, neutral) at time $t+1$ ($\hat{\boldsymbol{y}}_U^{(t+1)}$). The proposed model is hybrid since it combines three sources of information: i) U's own characteristics at time t ($\boldsymbol{x}_U^{(t)}$); ii) the characteristics of every neighbor j of U at time t ($\boldsymbol{x}_j^{(t)}$); and iii) overall market trend at time t ($\boldsymbol{z}^{(t)}$), which is not directly associated to a specific firm.

Our model starts with an embedding layer that applies GraphSAGE to each of our two neighborhood definitions independently ($N_1(U)$ and $N_2(U)$). Each convolution operator produces a firm embedding for node U at time t ($\boldsymbol{e}_{N_1(U)}^{(t)}$ and $\boldsymbol{e}_{N_2(U)}^{(t)}$), which are later concatenated with market information $\left(\boldsymbol{z}^{(t)}\right)$ to produce a node embedding for firm U at time t ($\boldsymbol{h}_U^{(t)}$). $\boldsymbol{h}_U^{(t)}$ is then used as the input of a fully connected layer to predict the target variable stock movement of firm U at time $t+1$.

Figure 3 depicts our model architecture.

Figure 3. Representation of the Hybrid and Temporal GNN Model (HT-GNN)



The first neighborhood of the focal firm U ($N_1(U)$) comprises its business partners (direct neighbors in the supply chain network), which in this example are firms *A*, *B*, and *C*. The second neighborhood ($N_2(U)$)) consists of the companies in the same network community the firm U is, which in this example would be firms *G*, *F*, and *I*. $\sum$ represents the GraphSAGE aggregator for the neighbors' features ($\boldsymbol{x}_j^{(t)}$), which in our experiments is a mean aggregator.

Following equations 2 and 3, we apply GraphSAGE to each neighborhood and obtain the node embeddings $\boldsymbol{e}_{N_1(U)}^{(t)}$ from the Direct Neighbors and $\boldsymbol{e}_{N_2(U)}^{(t)}$ from the Community Neighborhood. Equation 4 introduces our final embedding for the company U ($\boldsymbol{h}_U^{(t)}$)

$$\boldsymbol{e}_{N_1(U)}^{(t)} = \sigma\left(\boldsymbol{W}_1^{(N_1)}\boldsymbol{x}_U^{(t)} + \boldsymbol{W}_2^{(N_1)}\, mean_{j\in N_1(U)}\left(\boldsymbol{x}_j^{(t)}\right)\right) \tag{2}$$

$$\boldsymbol{e}_{N_2(U)}^{(t)} = \sigma\left(\boldsymbol{W}_1^{(N_2)}\boldsymbol{x}_U^{(t)} + \boldsymbol{W}_2^{(N_2)}\, mean_{j\in N_2(U)}\left(\boldsymbol{x}_j^{(t)}\right)\right) \tag{3}$$

$$\boldsymbol{h}_U^{(t)} = \left[\boldsymbol{e}_{N_1(U)}^{(t)}, \boldsymbol{e}_{N_2(U)}^{(t)}, \boldsymbol{z}^{(t)}\right] \tag{4}$$

From equations 2-4, $\boldsymbol{W}_1^{(N_1)}$, $\boldsymbol{W}_2^{(N_1)}$, $\boldsymbol{W}_1^{(N_2)}$, $\boldsymbol{W}_2^{(N_2)}$ are trainable weight parameters, σ is a ReLU activation function, $\boldsymbol{N}_1(U)$ is the set direct neighbors of a focal firm, $\boldsymbol{N}_2(U)$ is the set of neighbors in the same network community of the focal firm, and $[\cdot]$ is used for concatenation.

Then, $\boldsymbol{h}_U^{(t)}$ is passed to the fully connected layer. Since this is three-class prediction, we apply SoftMax to obtain the stock movement prediction at time *t+1* ($\hat{\boldsymbol{y}}_U^{(t+1)}$) and cross-entropy to compare with the ground truth. The whole model is jointly trained using historical stock data. As a result, these firm embeddings are learned in a supervised way, so that they not only help stock movement prediction, but also capture information about the market, characteristics of a focal firm, and its supply network neighbors (beyond immediate ties) over time in a low-dimensional space.

4.3 Feature Definition

The features included by our predictive model ($\boldsymbol{x}_U^{(t)}$) are related to a focal firm's characteristics and its financial performance. The set of features is presented below:

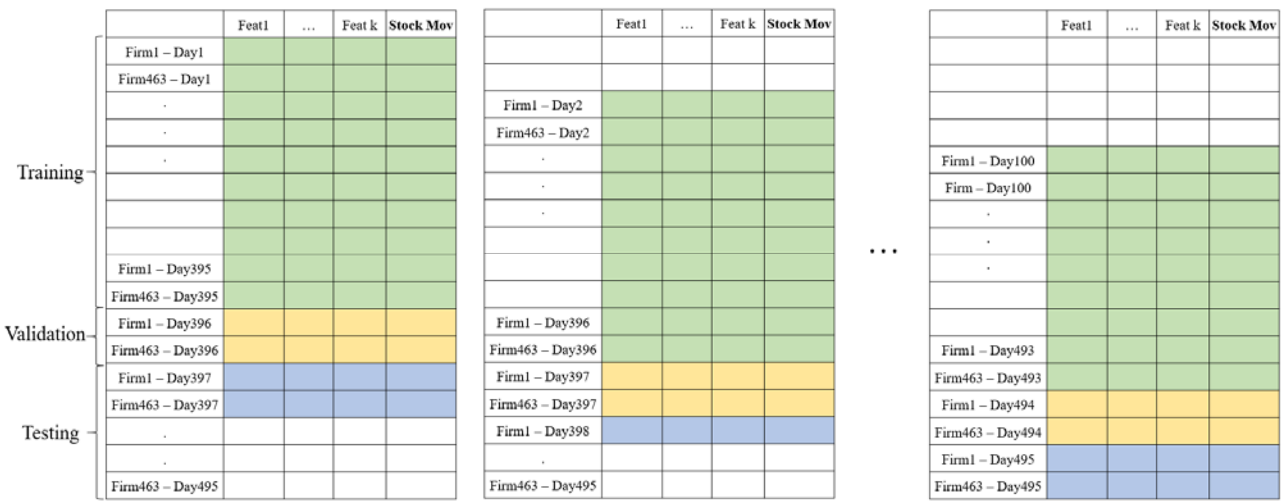
- Stock Price time *t*: the adjusted close price of stock *i* at day *t*.
- Stock Price time *t-1*: the adjusted close price of stock *i* at day *t-1*.
- Stock Return time *t*: the percentage of stock *i*'s price change from day *t-1* to *t*.
- Stock Return time *t-1*: the percentage of stock *i*'s price change from day *t-2* to *t-1*.
- Lowest Stock Price *t*: the lowest price of stock *i* at day *t*.
- Highest Stock Price *t*: highest price of stock *i* at day *t*.
- Trading Volume *t*: total number of shares of stock *i* traded at day *t*.
- Sector (NAICS): first digit of the NAICS sector code of company *i*. Since this variable is categorical, it was binarized into 8 dummy variables.

Additionally, market information ($\boldsymbol{z}^{(t)}$) is included through market return at time *t*, which is the return of S&P Composite Index (provided by CRSP) from day *t-1* to day *t*.

4.4 Experiments and Results

To train the proposed HT-GNN, we used a sliding time window of size 395 days including all our 463 focal firms. The model is then validated using one day of our focal firms' stock movement after the training period (t=396). Finally, we tested the model on the day after that (t=397). Then, we restart the model and move the time window one day. It means that we train from t = 2 to t = 396, validate on t = 397 and test on t = 398. We iterate over the complete dataset obtaining a resulting testing set of 100 days. The structure of our sliding time window evaluation is presented in Figure 4.

Figure 4 Sliding time window evaluation



To measure the model performance, we used area under the curve ROC (AUC). Since we defined this prediction as a three-class classification problem, we calculated the macro-average of the AUC (macro AUC), which averages the three ROC curves and calculates the AUC for the resulting curve.

We performed experiments to evaluate the relevance of the different parts of our model. The evaluated models are described below:

- **Baseline**: this basic model only includes market and the focal company's own information ($\boldsymbol{z}^{(t)}$ and $\boldsymbol{x}_U^{(t)}$) for its node representation $\boldsymbol{h}_U^{(t)}$. This means that it does not consider signals from any of the identified network neighborhoods.
- **Direct Neighbors**: in addition to the company's own information, this model also includes information about the company's direct neighbors to perform GraphSAGE. The resulting embedding $\boldsymbol{e}_{N_1(U)}^{(t)}$ and the market information ($\boldsymbol{z}^{(t)}$) are concatenated to produce $\boldsymbol{h}_U^{(t)}$, which is then passed to the fully connected layer when predicting for the focal company U. Different embedding sizes were evaluated on the validation set to pick the best value for this parameter.
- **Network Communities**: in addition to the company's own information, this model also includes information about all the company's neighbors in the same network community to perform GraphSAGE. The resulting embedding $\boldsymbol{e}_{N_2(U)}^{(t)}$ and the market information ($\boldsymbol{z}^{(t)}$) are

concatenated to produce $\boldsymbol{h}_U^{(t)}$, which is then passed to the fully connected layer when predicting for the focal company U. Different embedding sizes were evaluated on the validation set to pick the best value for this parameter.

- **Direct Neighbors and Network Communities**: this is the complete model we propose combining all the previously described components. The model concatenates $\boldsymbol{e}_{N_1(U)}^{(t)}$, $\boldsymbol{e}_{N_2(U)}^{(t)}$, and the market information ($\boldsymbol{z}^{(t)}$) to obtain $\boldsymbol{h}_U^{(t)}$ and pass it to the fully connected layer when predicting for the focal company U. Different combinations of embedding sizes for the two neighborhood definitions were evaluated on the validation set to pick the best one. One particular edit to the previous version of the neighborhoods is that in these models, companies in both sets are excluded from its community neighborhood since they are already included as a direct neighbor.
- **Two-hop neighbors**: this model is an extension of the Direct Neighbors model. As the network communities, this model aims to include information beyond immediate ties of a focal firm. In this version, the GraphSAGE aggregator is propagated until the 2-hop neighbors. The resulting embedding $\boldsymbol{e}_{N_1(U)}^{(t)}$ and the market information ($\boldsymbol{z}^{(t)}$) are concatenated to produce $\boldsymbol{h}_U^{(t)}$, which is then passed to the fully connected layer when predicting for the focal company U. Different embedding sizes were evaluated on the validation set to pick the best value for this parameter.

We compare the five models in the testing set of 100 days. The results were evaluated with 10 different random seeds to reduce the results variability and increase its robustness. We grouped the results by focal company (463 firms) and calculated a macro-AUC per company to evaluate the models' performance. A summary of the results is presented on Table 2 and **Error! Not a valid bookmark self-reference.**. To check if differences in performance between the proposed models and the baseline were statistically significant, we conducted paired t-tests to compare the macro-AUC means. These results are indicated by stars in Table 2. The comparison revealed that adding network neighbors' information helps to better predict a focal firm's future stock movement, since all the models including any kind of neighbor outperformed the baseline.

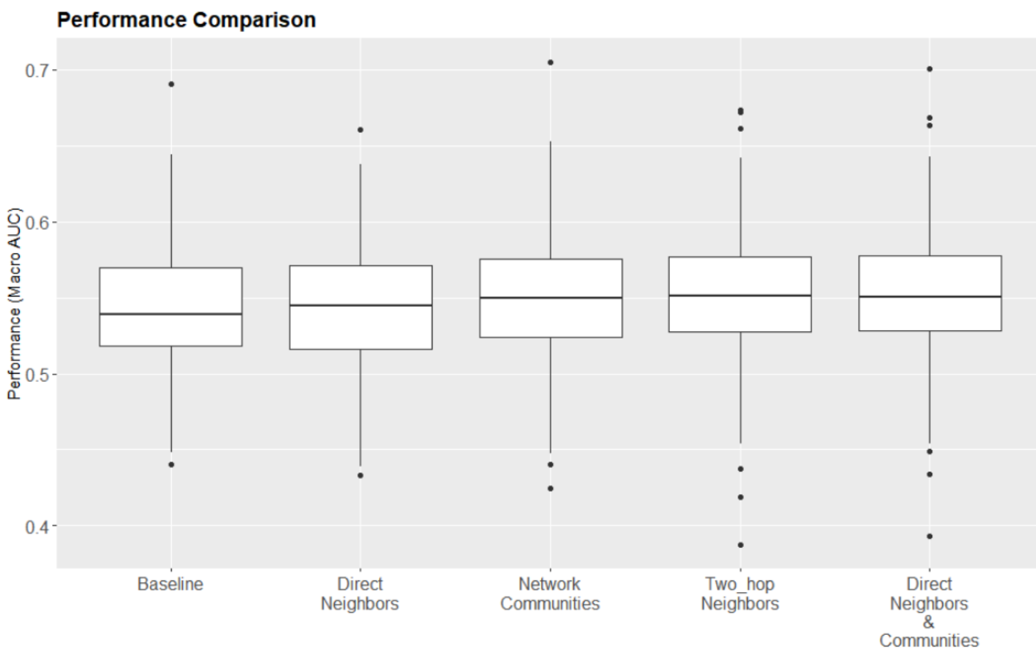
In an additional comparison, we found that both Network Communities and Two-hop Neighbors outperform the Direct Neighbors model with a p-value<0.001, highlighting the importance of including companies beyond immediate ties. Additionally, we found that the complete model (Direct Neighbors plus Network Communities) is the best one, outperforming the other four when comparing their means.

The results show that our HT-GNN can leverage signals from neighboring firms in a supply chain network to predict a focal firm's future stock movement and that the identified peer companies beyond direct neighbors can improve the prediction.

Table 2. Numerical comparison of the models' performance.

Macro AUC	**Baseline**	**Direct Neighbors**	**Network Communities**	**Two-hop Neighbors**	**Direct Neighbors and Network Communities**
Average	0.541	0.543*	0.549***	0.550***	**0.552*** **
Min	0.440	0.433	0.425	0.388	0.393
Max	0.690	0.660	0.705	0.673	0.700
Std Dev	0.040	0.038	0.039	0.040	0.040
*p-value<0.05, **p-value<0.01, ***p-value<0.001					

Figure 5. Graphical comparison of the models' performance.



5 Conclusions and Future Work

This paper contributes to the literature on stock price prediction by leveraging information from supply chain networks neighbors of a focal firm and combining that with market performance. Our main thesis states that analyzing the performance of a focal firm's neighbors in its supply chain network improves the prediction of its stock price movement. We propose an end-to-end deep learning model that learns how to aggregate network neighbors of a focal firm along with the market movement to predict the firm's stock movements over time. By defining different types of neighborhoods, we use experiments based on a large-scale supply chain network to show that including neighboring firms, especially those beyond immediate neighboring, in a supply chain network improves the prediction of stock movement. The model also produces dynamic firm representations (i.e., embeddings) that combine market movements, supply chain network structures, as well as characteristics and performance of the focal firm and its neighboring firms, making managerial decisions more interpretable. The results produced by these models would naturally help investors understand stock price movement. In addition, the models can help a firm identify, at a more granular level, network neighbors that have predictive power over its own stock price movement.

We have identified different potential directions for future work. The first one (and probably the most natural extension) would be to incorporate sequential learning into our prediction architecture. This would aim to learn any relevant temporal pattern from historical performance of individual firms and the market. A second addition would be to include an attention mechanism to combine the resulting embeddings to assign weights to the different neighborhood definitions. Finally, we can include sample mechanisms to the neighbor aggregations for the model to focus on those that might be more relevant.

References

Agarwal, A., Leung, A. C. M., Konana, P., & Kumar, A. (2017). Cosearch attention and stock return predictability in supply chains. *Information Systems Research*, 28(2), 265–288. https://doi.org/10.1287/isre.2016.0656

Bellamy, M. A., Ghosh, S., & Hora, M. (2014). The influence of supply network structure on firm innovation. *Journal of Operations Management*, 32(6), 357–373. https://doi.org/10.1016/j.jom.2014.06.004

Bruna, J., Zaremba, W., Szlam, A., & LeCun, Y. (2014). Spectral networks and deep locally connected networks on graphs. *2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings*.

Carhart, M. M. (1997). On persistence in mutual fund performance. *The Journal of Finance*, 52(1), 57–82.

Choi, T. Y., & Hong, Y. (2002). Unveiling the structure of supply networks: Case studies in Honda, Acura, and DaimlerChrysler. *Journal of Operations Management*, 20(5), 469–493. https://doi.org/10.1016/S0272-6963(02)00025-6

Cohen, L., & Frazzini, A. (2008). Economic links and predictable returns. *Journal of Finance*, 63(4), 1977–2011. https://doi.org/10.1111/j.1540-6261.2008.01379.x

Defferrard, M., Bresson, X., & Vandergheynst, P. (2016). Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in Neural Information Processing Systems*, 3844–3852. Retrieved from https://github.com/mdeff/cnn_graph

Fama, E. F., & French, K. R. (1996). Multifactor explanations of asset pricing anomalies. *The Journal of Finance*, 51(1), 55–84.

Feng, F., He, X., Wang, X., Luo, C., Liu, Y., & Chua, T. S. (2019). Temporal relational ranking for stock prediction. *ACM Transactions on Information Systems*, 37(2). https://doi.org/10.1145/3309547

Goyal, A., & Welch, I. (2003). Predicting the equity premium with dividend ratios. *Management Science*, 49(5), 639–654. https://doi.org/10.1287/mnsc.49.5.639.15149

Hamilton, W. L., Ying, R., & Leskovec, J. (2017). Inductive representation learning on large graphs. *Advances in Neural Information Processing Systems*, 1025–1035.

Hearnshaw, E. J. S., & Wilson, M. M. J. (2013). A complex network approach to supply chain network theory. *International Journal of Operations and Production Management*, 33(4), 442–469. https://doi.org/10.1108/01443571311307343

Hendricks, K. B., & Singhal, V. R. (2005). An empirical analysis of the effect of supply chain disruptions on long-run stock price performance and equity risk of the firm. *Production and Operations Management*, 14(1), 35–52.

Kim, Y., Choi, T. Y., Yan, T., & Dooley, K. (2011). Structural investigation of supply networks: A social network analysis approach. *Journal of Operations Management*, 29(3), 194–211. https://doi.org/10.1016/j.jom.2010.11.001

Kipf, T. N., & Welling, M. (2017). Semi-supervised classification with graph convolutional networks. *5th International Conference on Learning Representations, ICLR 2017 - Conference Track Proceedings*. Retrieved from https://arxiv.org/pdf/1609.02907.pdf

Lintner, J. (1965). The valuation of risk assets and the selection of risky investments in stock portfolios and capital budgets. *Review of Economics and Statistics*, 47(1), 13–37.

Luo, X., Zhang, J., & Duan, W. (2013). Social media and firm equity value. *Information Systems Research*, 24(1), 146–163. https://doi.org/10.1287/isre.1120.0462

Menzly, L., & Ozbas, O. (2010). Market segmentation and cross-predictability of returns. *Journal of Finance*, 65(4), 1555–1580. https://doi.org/10.1111/j.1540-6261.2010.01578.x

McPherson, M., Smith-Lovin, L., & Cook, J. M. (2001). Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, *27*(1), 415–444. https://doi.org/10.1146/annurev.soc.27.1.415

Merschmann, U., & Thonemann, U. W. (2011). Supply chain flexibility, uncertainty and firm performance: An empirical analysis of German manufacturing firms. *International Journal of Production Economics*, 130(1), 43–53. https://doi.org/10.1016/j.ijpe.2010.10.013

Newman, M. E. J., & Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical Review E*, 69(2 2). https://doi.org/10.1103/PhysRevE.69.026113

Nguyen, T. H., Shirai, K., & Velcin, J. (2015). Sentiment analysis on social media for stock movement prediction. *Expert Systems with Applications*, 42(24), 9603–9611. https://doi.org/10.1016/j.eswa.2015.07.052

Rechenthin, M., Street, W. N., & Srinivasan, P. (2013). Stock chatter: Using stock sentiment to predict price direction. *Algorithmic Finance*, 2(3–4), 169–196. https://doi.org/10.3233/AF-13025

Rios, J., Zhao, K., & Street, N. (2019). Predicting stock price movements via multi-relational inter-firm networks. *13th China Summer Workshop on Information Management*, 336–341.

Seiler, A., Papanagnou, C., & Scarf, P. (2020). On the relationship between financial performance and position of businesses in supply chain networks. *International Journal of Production Economics*, 227, 107690. https://doi.org/10.1016/j.ijpe.2020.107690

Sharpe, W. F. (1964). Capital asset prices: A theory of merket equilibrium under conditions of risk. *The Journal of Finance*, 19(3), 425–442. https://doi.org/10.1111/j.1540-6261.1964.tb02865.x

Vanichchinchai, A., & Igel, B. (2011). The impact of total quality management on supply chain management and firm's supply performance. *International Journal of Production Research*, 49(11), 3405–3424. https://doi.org/10.1080/00207543.2010.492805

Veličković, P., Casanova, A., Liò, P., Cucurull, G., Romero, A., & Bengio, Y. (2018). Graph attention networks. *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*.

Vrandečić, D., & Krötzsch, M. (2014). Wikidata: A free collaborative knowledgebase. *Communications of the ACM*, 57(10), 78–85. https://doi.org/10.1145/2629489

Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., & Yu, P. S. (2020). A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 1–21. https://doi.org/10.1109/tnnls.2020.2978386

Zhao, K., Ngamassi, L. M., Yen, J., Maitland, C., & Tapia, A. (2010). Assortativity patterns in multi-dimensional inter-organizational networks: A case study of the humanitarian relief sector. *In International Conference on Social Computing, Behavioral Modeling, and Prediction*, 265-272. Springer, Berlin, Heidelberg.

Zhao, K., Kumar, A., & Yen, J. (2011). Achieving high robustness in supply distribution networks by rewiring. *IEEE Transactions on Engineering Management*, 58(2), 347–362. https://doi.org/10.1109/TEM.2010.2095503

Zhou, J., Cui, G., Zhang, Z., Yang, C., Liu, Z., Wang, L., … Sun, M. (2018). Graph Neural Networks: A Review of Methods and Applications. *Retrieved from http://arxiv.org/abs/1812.08434*